



Detección de objetos en señales de radar usando redes neuronales convolucionales

Novoa Gordillo, Miguel Andrés

Departamento de Eléctrica, Electrónica y Telecomunicaciones

Carrera de Ingeniería en Electrónica y Telecomunicaciones

Trabajo de titulación, previo a la obtención del título de Ingeniero en Electrónica y
Telecomunicaciones

Ing. Carrera Erazo, Enrique Vinicio. PhD.

30 de noviembre del 2023



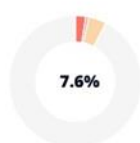
Plagiarism and AI Content Detection Report

Novoa_Miguel_Trabajo_Titulacion_V4...

Scan details

Scan time: November 20th, 2023 at 12:34 UTC Total Pages: 45 Total Words: 11227

Plagiarism Detection



| Types of plagiarism | | Words |
|---------------------|------|-------|
| Identical | 2.4% | 271 |
| Minor Changes | 0.7% | 75 |
| Paraphrased | 4.1% | 461 |
| Omitted Words | 5.8% | 653 |

AI Content Detection



| Text coverage | | Words |
|---------------|-------|-------|
| AI text | 1.7% | 190 |
| Human text | 98.3% | 11037 |

[Learn more](#)

🔍 Plagiarism Results: (35)

🌐 CD 12716.pdf

1.4%

<https://bibdigital.epn.edu.ec/bitstream/15000/23300/1/cd%2...>

Carlos Acuña

ESCUELA POLITÉCNICA NACIONAL FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA DISEÑO E IMPLEMENTACIÓN DE MINIDRONES (UAVS)...

🌐 T-ESPE-052897.pdf

0.9%

<https://repositorio.espe.edu.ec/bitstream/21000/36668/1/t-e...>

Gaby Rodríguez

1 Área de vida del grupo de Cebus albifrons por medio de herramientas bioinformáticas Rodríguez Rodríguez, Gabriela Nikole Departament...

🌐 T-ESPE-052717.pdf?sequence=1&isAllowed=y

0.9%

<http://repositorio.espe.edu.ec/xmlui/bitstream/handle/2100...>

CRISTIAN PUEBLA MONTERO

Monitoreo de la fabricación de los intercambiadores compactos usando el método de fundición por cera perdida Puebla Montero, Cristian Ja...



Certified by
Copyleaks

About this report
help.copyleaks.com

copyleaks.com
in f o t



Departamento de Eléctrica, Electrónica y Telecomunicaciones

Carrera de Ingeniería en Electrónica y Telecomunicaciones

Certificación

Certifico que el trabajo de titulación: **"Detección de objetos en señales de radar usando redes neuronales convolucionales"** fue realizado por el señor **Novoa Gordillo Miguel Andrés**; el mismo que cumple con los requisitos legales, teóricos, científicos, técnicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, además fue revisado y analizado en su totalidad por la herramienta de prevención y/o verificación de similitud de contenidos; razón por la cual me permito acreditar y autorizar para que se lo sustente públicamente.

Sangolquí, 30 de noviembre de 2023

Firma:



Carrera Erazo Enrique Vinicio

C. C. 1708792104



Departamento de Eléctrica, Electrónica y Telecomunicaciones

Carrera de Ingeniería en Electrónica y Telecomunicaciones

Responsabilidad de Autoría

Yo, **Novoa Gordillo Miguel Andrés**, con cédula de ciudadanía N° 1726919721, declaro que el contenido, ideas y criterios del trabajo de titulación: **Detección de objetos en señales de radar usando redes neuronales convolucionales** es de mi autoría y responsabilidad, cumpliendo con los requisitos legales, teóricos, científicos, técnicos, y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

Sangolquí, 30 de noviembre de 2023

Firma

Novoa Gordillo Miguel Andrés

C.C.: 1726919721



Departamento de Eléctrica, Electrónica y Telecomunicaciones

Carrera de Ingeniería en Electrónica y Telecomunicaciones

Autorización de Publicación

Yo **Novoa Gordillo Miguel Andrés**, con cédula de ciudadanía N° 1726919721, autorizo a la Universidad de las Fuerzas Armadas ESPE publicar el trabajo de titulación: **Detección de objetos en señales de radar usando redes neuronales convolucionales** en el Repositorio Institucional, cuyo contenido, ideas y criterios son de mi responsabilidad.

Sangolquí, 30 de noviembre de 2023

Firma

Novoa Gordillo Miguel Andrés

C.C.: 1726919721

Dedicatoria

El presente trabajo lo dedico a Dios quien me ha guiado siempre ante todo reto durante la carrera, a mi madre y mi padre por ser siempre mis guías y ejemplos de dedicación, a mi abuelita por estar siempre a mi lado durante todo este proceso y ser mi fuente de valores, a mi hermana mis tíos y mi familia por su constante apoyo durante este proceso, a mi novia por su motivación y ayuda ante cualquier adversidad durante el desarrollo del proyecto, a mi tutor de tesis quien me ha brindado su constante orientación y cooperación para mejorar tanto personal, como académicamente y finalmente a todo compañero, ingeniero y persona a lo largo de mi vida que ha contribuido para que este objetivo se haga realidad.

Agradecimiento

Primeramente, quiero agradecer a Dios, quien fue mi guía y fortaleza desde el comienzo de la carrera por cada oportunidad que me brindo de aprendizaje en todos desafíos que he enfrentado.

A mi madre y mi padre, con quienes no tengo palabras suficientes para expresar tanto agradecimiento por el apoyo incondicional y el amor que me han brindado; por ser mi mayor ejemplo de esfuerzo y superación y quienes son mi constante inspiración y ejemplos a seguir.

A mi abuelita que desde mi niñez fue un pilar en todo mi desarrollo y formación, quien me inculcó mediante su sabiduría y amor los mayores valores de mi vida y con quien estaré eternamente agradecido toda mi vida.

A mi hermana, mis tíos y mi familia en general, gracias por estar siempre presentes en cada momento a lo largo de estos años, quienes me han ayudado a construir cada una de mis metas.

A mi novia, por ser mi compañera de vida y con quien comparto las más grandes metas y sueños para seguir creciendo juntos como pareja.

A mi tutor de tesis, quien mediante la enseñanza y educación ha logrado construir junto a mi este proyecto a base de esfuerzo, tiempo y dedicación.

Finalmente extendiendo un agradecimiento a mi universidad, a mis compañeros e ingenieros que a lo largo de la carrera han formado parte de mi desarrollo tanto personal como académico.

Índice De Contenidos

| | |
|---|----|
| Certificado del Tutor..... | 3 |
| Autoría de Responsabilidad | 4 |
| Autorización | 5 |
| Dedicatoria..... | 6 |
| Agradecimiento..... | 7 |
| Índice de Contenidos | 8 |
| Índice de Tablas | 11 |
| Índice de Figuras | 12 |
| Resumen | 14 |
| Abstract..... | 15 |
| Capítulo I: Introducción | 16 |
| Antecedentes..... | 16 |
| Justificación | 16 |
| Alcance del proyecto | 17 |
| Objetivos | 18 |
| <i>Objetivo General</i> | 18 |
| <i>Objetivos Específicos</i> | 18 |
| Capítulo II: Marco Teórico..... | 19 |
| Sistemas de Radar | 19 |
| <i>Relación señal a ruido en señales de radar</i> | 19 |

| | |
|--|-----------|
| Aprendizaje profundo y redes neuronales artificiales | 20 |
| <i>Modelo matemático y computacional de una red neuronal artificial</i> | 21 |
| <i>Funciones de Activación</i> | 23 |
| Función de activación ReLu..... | 25 |
| Función Softmax..... | 25 |
| <i>Tipos de capas en redes neuronales artificiales</i> | 26 |
| Capa de Entrada..... | 26 |
| Capa Completamente Conectada. | 26 |
| Capa Dropout. | 27 |
| <i>Entrenamiento de Redes Neuronales Artificiales</i> | 28 |
| Algoritmos de Optimización..... | 28 |
| <i>Redes Neuronales Convolucionales</i> | 29 |
| Estructura de las Redes Neuronales Convolucionales. | 29 |
| <i>Operación de convolución y capa convolucional.....</i> | 30 |
| <i>Padding.</i> | 31 |
| <i>Capa de Pooling.</i> | 32 |
| Redes Neuronales Convolucionales Unidimensionales..... | 34 |
| Redes Neuronales Convolucionales Bidimensionales..... | 34 |
| Transfer Learning. | 34 |
| <i>Alexnet.....</i> | 36 |
| <i>Matriz de Confusión y Parámetros de Rendimiento</i> | 36 |

| | |
|---|-----------|
| Capítulo III: Metodología | 39 |
| Descripción General | 39 |
| Señales De Radar Simuladas..... | 39 |
| Preparación de Bases De Datos..... | 43 |
| <i>Bases de datos para red neuronal convolucional unidimensional.....</i> | <i>44</i> |
| <i>Bases de datos para red neuronal convolucional bidimensional.....</i> | <i>45</i> |
| Diseño de Redes Neuronales Convolucionales | 46 |
| <i>Diseño de red convolucional unidimensional</i> | <i>47</i> |
| <i>Diseño de red convolucional bidimensional.....</i> | <i>49</i> |
| Capítulo IV: Resultados | 51 |
| Resultados Red Neuronal Convolucional Unidimensional..... | 51 |
| Resultados Red Neuronal Convolucional Bidimensional..... | 55 |
| Capítulo V: Conclusiones | 60 |
| Bibliografía | 62 |

Índice de Tablas

| | |
|--|-----------|
| Tabla 1 Funciones de activación | 24 |
| Tabla 2 Resumen de los resultados de entrenamiento y validación de la red neuronal convolucional unidimensional | 53 |
| Tabla 3 Resultados de la matriz de confusión de la red neuronal convolucional unidimensional..... | 54 |
| Tabla 4 Parámetros de rendimiento de la red neuronal convolucional unidimensional ... | 55 |
| Tabla 5 Resumen de los resultados de entrenamiento y validación de la red neuronal convolucional bidimensional. | 57 |
| Tabla 6 Resultados de la matriz de confusión de la red neuronal convolucional bidimensional..... | 58 |
| Tabla 7 Parámetros de rendimiento de la red neuronal convolucional bidimensional | 59 |

Índice de Figuras

| | |
|---|----|
| Figura 1 Ramas de la inteligencia artificial..... | 21 |
| Figura 2 Estructura general de una red neuronal artificial..... | 22 |
| Figura 3 Proceso matemático de una red neuronal artificial | 23 |
| Figura 4 Grafica de función de activación ReLU | 25 |
| Figura 5 Funcionamiento de capa dropout | 28 |
| Figura 6 Operación de la Capa Convolutiva | 31 |
| Figura 7: Funcionamiento del padding | 32 |
| Figura 8 Funcionamiento de la capa pooling..... | 33 |
| Figura 9 Redes neuronales convolucionales pre entrenadas..... | 35 |
| Figura 10 Matriz de Confusión | 37 |
| Figura 11 Parámetros de Rendimiento de Matriz de Confusión | 38 |
| Figura 12 Ejemplo del funcionamiento de un radar y localización de un objetivo..... | 40 |
| Figura 13 Vector resultante del barrido del sistema simulador de señales de radar | 40 |
| Figura 14 Componentes en fase y cuadratura de una señal de radar a una SNR de 15 dBs | 41 |
| Figura 15 Componentes en fase y cuadratura de una señal de radar a una SNR de 0 dBs | 42 |
| Figura 16 Componentes en fase y cuadratura de una señal de radar a una SNR de -5 dBs | 42 |
| Figura 17 Subconjuntos de muestras tomadas de la señal de radar..... | 43 |

| | |
|---|-----------|
| Figura 18 Asignación de etiquetas a los vectores para la identificación de la presencia de objetivos | 44 |
| Figura 19 Base de datos para red neuronal convolucional unidimensional | 45 |
| Figura 20 Espectrograma de una señal de radar simulada..... | 46 |
| Figura 21 Re ajuste del área de espectrograma. | 46 |
| Figura 22 Arquitectura de la red neuronal convolucional unidimensional | 48 |
| Figura 23 Arquitectura de la red neuronal convolucional bidimensional | 50 |
| Figura 24 Ejemplo de entrenamiento de la red convolucional unidimensional..... | 51 |
| Figura 25 Ejemplo del resultado de matriz de confusión la red convolucional unidimensional..... | 52 |
| Figura 26 Ejemplo de entrenamiento de la red convolucional bidimensional | 56 |
| Figura 27 Ejemplo del resultado de matriz de confusión la red convolucional bidimensional..... | 56 |

Resumen

Uno de los principales aplicativos de los radares es la identificación de objetivos en las señales obtenidas por el barrido de este sistema; la señal resultante pasa por un proceso de análisis para determinar si existe o no la presencia de un objeto y esta labor incrementa su complejidad en bajos niveles de relación señal a ruido (SNR). Para solventar este inconveniente se propone el uso de técnicas alternativas mediante algoritmos de aprendizaje profundo (deep learning) y de redes neuronales artificiales intentando mejorar el desempeño, la capacidad y la calidad de radares actualmente existentes. Para lograr este objetivo se estudia la utilización de un tipo de red neuronal denominada red neuronal convolucional.

En el presente trabajo se realizó una implementación y análisis de las redes neuronales convolucionales unidimensionales y bidimensionales para una posterior comparativa de los resultados obtenidos entre ambas redes, las cuales se desarrollaron orientadas a la clasificación de la presencia o no de objetivos en señales de radar. Los resultados obtenidos muestran que en los niveles de SNR bajos de 0 dBs a -5 dBs las redes neuronales convolucionales lograron al menos un 99.94% de exactitud en la clasificación.

El uso de las redes neuronales convolucionales y las técnicas de aprendizaje profundo implican un mayor costo computacional y un mayor tiempo de entrenamiento de los modelos, pero tiene como ventaja óptimos resultados respecto a las prácticas tradicionales utilizadas para la detección de objetivos en señales de radar.

Palabras Clave: aprendizaje profundo, redes neuronales convolucionales, radares, detección de objetivos, relación señal a ruido

Abstract

One of the main applications of radars is the identification of targets in the signals obtained by scanning this system; The resulting signal goes through an analysis process to determine whether or not there is the presence of an object and this task increases its complexity at low levels of signal to noise ratio (SNR). To solve this problem, the use of alternative techniques is proposed through deep learning algorithms and artificial neural networks in an attempt to improve the performance, capacity and quality of currently existing radars. To achieve this objective, the use of a type of neural network called convolutional neural network is studied.

In the present work, an implementation and analysis of one-dimensional and two-dimensional convolutional neural networks was carried out for a subsequent comparison of the results obtained between both networks, which were developed oriented to the classification of the presence or absence of targets in radar signals. The results obtained show that at low SNR levels from 0 dBs to -5 dBs the convolutional neural networks achieved at least 99.94% classification accuracy.

The use of convolutional neural networks and deep learning techniques imply a higher computational cost and a longer training time for the models, but it has the advantage of optimal results compared to the traditional practices used for the detection of targets in radar signals.

Keywords: deep learning, convolutional neural networks, radars, target detection, signal to noise ratio

Capítulo I: Introducción

Antecedentes

Los sistemas de radar funcionan a través de radiaciones electromagnéticas y entre sus aplicativos permiten detectar y localizar objetos en condiciones donde el ser humano no es capaz de realizar esta tarea (López Espejo, 2011). El uso de radares se remonta desde la Segunda Guerra Mundial para alertar tempranamente aviones enemigos los cuales a simple vista eran imperceptibles (Parker Sanfuentes, 2016). Con el pasar de los años los sistemas de radar han evolucionado tecnológicamente, uno de los cambios más relevantes es la técnica de barrido electrónico, donde la antena no necesita moverse físicamente para que las ondas de radar apunten en diversas direcciones (Donalba, 2015).

Entre los retos que se deben considerar en la detección de objetos en señales de radar es que la señal de interés se ve afectada por reflexiones generadas por el ambiente, fenómeno denominado “clutter” lo cual dificulta la localización del objeto. Por este motivo se han desarrollado diversas técnicas de procesamiento de señales, de detección adaptativa o métodos más sofisticados para identificar los objetos en situaciones de clutter intenso (Pascual, 2014). En los últimos años con el avance tecnológico se ha producido un incremento en las capacidades de procesamiento de las computadoras lo que ha facilitado el desarrollo de nuevos algoritmos y técnicas de aprendizaje automático logrando optimizar procesos sin la intervención humana. Un ejemplo de ello es el aprendizaje profundo que es capaz de imitar el funcionamiento del cerebro mediante redes neuronales artificiales (Sanchez & Expósito, 2020).

Justificación

En el año 2008 se produjo un trágico acontecimiento en el Ecuador, un bombardeo aéreo en el territorio de Angostura dejó como resultado la pérdida de varias vidas; en esa época la Fuerza Aérea Ecuatoriana contaba con 4 radares que se encontraban discontinuados y sin

funcionamiento dejando en evidencia la vulnerabilidad del cielo ecuatoriano ante futuros ataques; este hecho fue un claro ejemplo de la necesidad de reforzar la vigilancia para detectar y repeler futuros atentados de esta índole (INREDH, 2021). En 2013 se llevó a cabo un proyecto que buscaba el fortalecimiento de la seguridad y el sistema de vigilancia del espacio aéreo del Ecuador; el objetivo que se buscaba alcanzar es el incremento de la defensa aérea mediante la adquisición de varios radares a diferentes países (Presidencia Ecuador, 2017).

Actualmente el territorio ecuatoriano cuenta con un conjunto de cuatro radares primarios y siete radares secundarios, cubriendo aproximadamente el 95% del espacio aéreo del Ecuador (Vaca Santacruz, 2021). Aun así, la Fuerza Aérea Ecuatoriana (FAE) en abril del 2022 informó que se tienen aproximadamente 376 áreas no vigiladas y varios casos de aeronaves no identificadas que posiblemente se dediquen a actividades ilícitas a lo largo del territorio ecuatoriano (Primicias, 2021). Es de gran importancia un sistema de monitoreo constante que analice los objetos que ingresen y salgan del territorio ecuatoriano; con el desarrollo de nuevas alternativas que ayuden a la detección de objetos en señales de radar se puede brindar un gran apoyo a la seguridad aérea, marítima o terrestre del Ecuador, además de mejorar las características de los radares actualmente existentes.

Alcance del proyecto

Para el desarrollo del sistema de detección de objetos en señales de radar se hará uso de un simulador de señales aprobado por el Centro de Investigación de Aplicaciones Militares (CIAM) con el cual se puede obtener una base de datos para el entrenamiento y validación del sistema; cada señal contiene un objeto localizado de forma aleatoria en la señal de radar, se deben simular diferentes bases de datos para diversos niveles SNR. Se utilizarán algoritmos de aprendizaje profundo centrándose en una comparativa entre las redes neuronales convolucionales unidimensionales con las redes neuronales convolucionales bidimensionales.

Ambas redes se programarán tomando en consideración su funcionamiento, estructura y aplicación orientada a la clasificación.

De la señal de radar simulada se sustraerán los datos donde existe y no existe la presencia del objetivo, ambas muestras de igual tamaño serán datos de entrada para el sistema. De la base de datos recopilada se elegirá de forma aleatoria un 70% de datos que se destinarán para el entrenamiento y el 30% restante para la validación a diferentes niveles de relación señal a ruido. De los resultados obtenidos se realizará una comparativa entre ambos sistemas para concluir cuál de los dos algoritmos presenta mejores prestaciones y mayor exactitud para la detección de objetos.

Objetivos

Objetivo General

Detectar objetos en señales de radar usando redes neuronales convolucionales para mejorar el desempeño, la capacidad y la calidad de radares actualmente existentes.

Objetivos Específicos

- Investigar los diferentes algoritmos existentes de redes neuronales convolucionales y sus aplicaciones.
- Obtener los conjuntos de datos generados por simulación de señales de radar para el entrenamiento del sistema y comprobar el funcionamiento del algoritmo con un nuevo conjunto de datos generado por el mismo software.
- Implementar los modelos basados en redes neuronales convolucionales para la detección de objetos en señales de radar.
- Evaluar los resultados obtenidos de la detección de objetos por el sistema de redes neuronales convolucionales unidimensionales contra las redes neuronales convolucionales bidimensionales y otros algoritmos de máquina tradicionales.

Capítulo II: Marco Teórico

En el presente capítulo se tratarán dos diferentes apartados: el primero explicará el funcionamiento de los sistemas de radar junto con la importancia de la relación señal a ruido para la detección de objetivos; el segundo apartado se centrará en el aprendizaje profundo y las redes neuronales artificiales, el cual a su vez se constituirá por diferentes subsecciones donde se mostrará su funcionamiento matemático y computacional, su estructura, los tipos de capas que las componen, el entrenamiento y un tipo de red neuronal artificial denominada red neuronal convolucional.

Sistemas de Radar

Los sistemas de radar son sistemas electrónicos que irradian señales electromagnéticas en forma de haz y receptan ondas que contienen la reflexión de los objetos que se encuentren en su trayecto (Alonso Cerpa, 2015).

La señal de radar recibida normalmente se ve afectada por reflexiones no deseadas debido al entorno donde se ubica el objeto, fenómeno que se denomina “clutter”. El “clutter” posee un comportamiento aleatorio producido por diversos factores provenientes de la naturaleza y evita que la relación señal a ruido sea óptima para la identificación del objeto (Pascual, 2014).

Resulta de gran importancia tener conocimiento sobre la influencia de la relación señal a ruido en una señal de radar dado que, a una mayor medida de SNR es más sencillo identificar el objetivo en la señal, ya que el ruido es menor.

Relación señal a ruido en señales de radar

El ruido puede definirse como un fenómeno natural que forma parte de todos los sistemas de telecomunicaciones, este posee un comportamiento aleatorio y se superpone a la señal de interés generando una disminución en la calidad e información de señal a analizar (Pérez Patiño,

2007). La relación señal a ruido, de su abreviatura S/N o SNR, es el resultado de la división entre las potencias de la señal y el ruido, expresado matemáticamente como:

$$\frac{S}{N} = \frac{\text{Potencia de señal}}{\text{Potencia de ruido}}$$

Habitualmente esta relación se expresa en unidades de decibelios (dB) mediante la siguiente fórmula (Pérez Patiño, 2007):

$$\left(\frac{S}{N}\right)_{\text{dB}} = 10 * \log\left(\frac{\text{Potencia de señal}}{\text{Potencia de ruido}}\right)$$

Mediante el uso de este desarrollo matemático se puede analizar que la detección de objetivos en niveles de relación señal a ruido menores de 0 dBs aumenta en gran medida la complejidad de la discriminación del objeto ya que la potencia del ruido es mayor a la potencia de la señal de radar.

Teniendo ya en consideración el desarrollo teórico de los sistemas de radar se continuará con el segundo apartado que explicará el aprendizaje profundo y las redes neuronales artificiales.

Aprendizaje profundo y redes neuronales artificiales

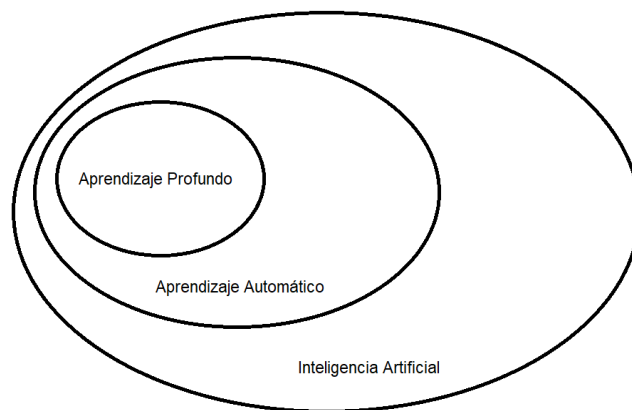
La inteligencia artificial representa uno de los mayores avances tecnológicos de los últimos años, su actual crecimiento se debe principalmente a dos factores: el alto procesamiento computacional que poseen los ordenadores y el acceso a grandes volúmenes de datos almacenados digitalmente (Muñiz González, 2018).

A partir de esto surge una rama de la inteligencia artificial denominada aprendizaje automático cuyo objetivo es que las computadoras “aprendan” mediante el desarrollo de algoritmos creando la capacidad de generalizar comportamientos e identificar patrones basados en la información proporcionada (Alba Centeno, 2019). Del aprendizaje automático surge un subconjunto denominado aprendizaje profundo el cual centra su funcionamiento en el uso de

redes neuronales artificiales, simulando la estructura cerebral del ser humano para lograr el aprendizaje por medio de diversas capas interconectadas de forma jerárquica (IAAR, 2017). En la figura 1 se muestra de forma gráfica en un diagrama de Venn las ramas de la inteligencia artificial.

Figura 1

Ramas de la inteligencia artificial



Para un mejor entendimiento teórico, resulta de gran utilidad desglosar las redes neuronales artificiales en cinco diferentes subdivisiones, se expondrá primeramente el modelo matemático y computacional de una red neuronal artificial, posteriormente se tratará la teoría de las funciones de activación, los diferentes tipos de capas en una red neuronal, el funcionamiento durante el entrenamiento de dichas redes; para finalmente exponer un tipo de red neuronal específica, denominada red neuronal convolucional.

Modelo matemático y computacional de una red neuronal artificial

Una red neuronal artificial puede definirse como un modelo matemático implementado computacionalmente que tiene por objetivo emular las funciones de las neuronas biológicas; las redes neuronales generan el aprendizaje mediante el uso de las experiencias, logran generalizar

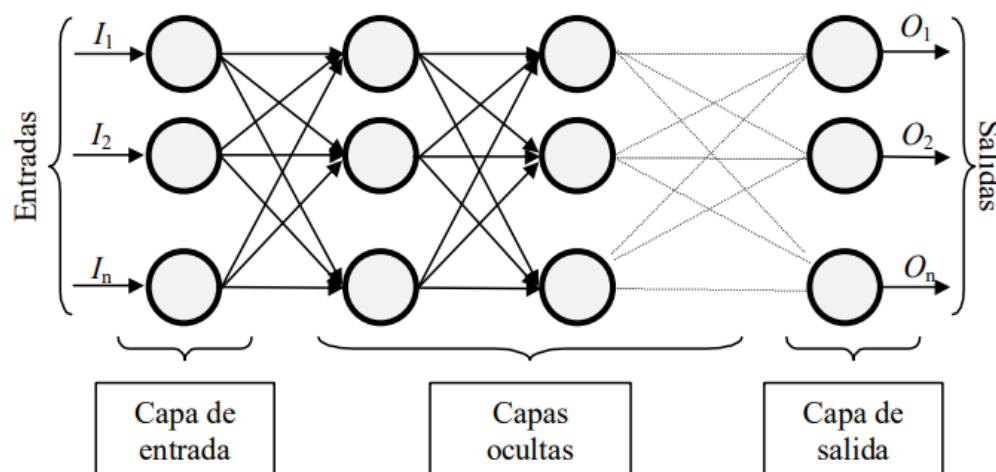
respuestas nuevas en base a respuestas previas y pueden abstraer las principales características de un conjunto de datos de entrada (Basogain Olabe, 2017).

El proceso que permite emular la funcionalidad de las neuronas biológicas es mediante el uso de una función, también denominada función de activación que permite cambiar el valor de los estados internos dependiendo de las señales que se recepten del exterior o de otras neuronas a las cuales se encuentran conectadas. Matemáticamente los estados internos se expresarían como una suma ponderada entre las entradas y los pesos (Isasi & Galván, 2004).

La estructura general de una red neuronal artificial multicapa se constituye por: una capa de entrada que se encarga del ingreso de datos, un conjunto de capas ocultas que procesan los datos de entrada o de las capas ocultas anteriores y una capa de salida que presenta el resultado del proceso de la red neuronal; es importante mencionar que el número de salidas de la última capa depende del aplicativo para la cual la red neuronal fue diseñada (Ruiz & Basualdo, 2001).

Figura 2

Estructura general de una red neuronal artificial

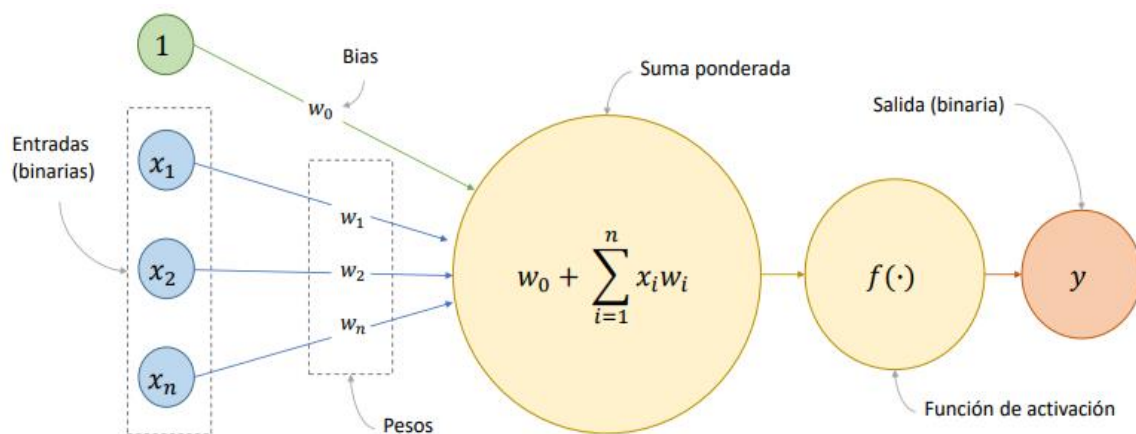


Nota. El gráfico muestra una red neuronal artificial multicapa. Tomado de (Ruiz & Basualdo, 2001).

Para explicar matemáticamente el proceso de una red neuronal artificial se tiene un vector con los datos de entrada a la red $[x_1, x_2, \dots, x_n]$ y un vector con los pesos $[w_1, w_2, \dots, w_n]$; entre ambos vectores se realiza una multiplicación y posterior sumatoria desde 1 hasta n, de forma que la señal con mayor peso tiene un mayor efecto en la red y a la vez estos mismos pesos son los que generan la imitación de cómo el cerebro puede cambiar la forma en que se asocian las neuronas. Posterior a la suma ponderada el resultado ingresa a una función de activación y finalmente se obtiene la salida de la red neuronal artificial (Cabrera Mora, 2021).

Figura 3

Proceso matemático de una red neuronal artificial



Nota. En la figura se muestra la estructura matemática de una red neuronal artificial y como mediante los datos de entrada y los pesos se obtiene la salida de la red. Tomado de (Cabrera Mora, 2021).

Funciones de Activación

En una red neuronal artificial cada neurona posee un valor asociado denominado estado de activación, una función de activación permite conocer el siguiente estado de activación en

base a la entrada total y al anterior estado. La salida de la función devuelve un valor en el rango de $\{0, 1\}$ o de $\{-1, 1\}$ dependiendo de la función escogida (González Molina, s. f.).

Las funciones de activaciones pueden ser lineales o no lineales, las no lineales simulan de mejor manera la funcionalidad de las neuronas biológicas, en la tabla 1 se realiza un resumen de la representación matemática de las funciones de activación, junto con el rango en el cual operan cada una de ellas.

Tabla 1

Funciones de activación

| Tipo | Función | Rango |
|------------|------------------------------------|----------------------|
| Identidad | $y = x$ | $[-\infty, +\infty]$ |
| Escalón | $y = \text{sign}(x)$ | $\{-1, +1\}$ |
| | $y = H(x)$ | $\{0, +1\}$ |
| Sigmoide | $y = \frac{1}{1 + e^x}$ | $[0, +1]$ |
| | $y = \text{tanh}(x)$ | $[-1, +1]$ |
| Gaussiana | $y = Ae^{-Bx^2}$ | $[0, +1]$ |
| Sinusoidal | $y = A * \sin(\omega x + \varphi)$ | $[-1, +1]$ |
| ReLU | $y = \max\{0, x\}$ | $[0, +\infty]$ |

Nota. En la tabla se muestran las diversas funciones de activación, su representación matemática y el rango en el cual se desarrollan. Tomado de (López Pacheco, 2021).

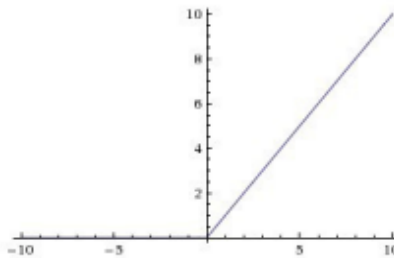
Resulta de gran importancia en una red neuronal convolucional profundizar en 2 funciones en especial: la función de activación ReLU y la función Softmax debido a que estas son las funciones más utilizadas en el diseño de la arquitectura de una red convolucional.

Función de activación ReLU. La función de activación ReLU (Rectified Linear Unit) es una de las funciones más utilizada por las redes neuronales convolucionales ya que aumenta la eficiencia gracias a la disminución del costo de cálculo y su prevención al incremento exponencial de recursos computacionales debido a que su derivada da como resultado 0 o 1 dependiendo si el valor ingresado es positivo o negativo ayudando al aprendizaje y eficiencia de la red; matemáticamente se expresa como (Casado Beinat, 2022):

$$\text{ReLU}(x) = \max \{0, x\}$$

Figura 4

Grafica de función de activación ReLU



Nota. En la gráfica se muestra que la función de activación ReLU se activa cuando el valor de la variable X pasa de cero. Para valores positivos la función ReLU toma el máximo entre cero y el valor de entrada y para valores negativos la función ReLU toma el valor de cero. Tomado de (Vilagran Solsona, 2018).

Función Softmax. La función Softmax es una función que se utiliza en las capas finales de las redes neuronales convolucionales aplicada para la clasificación de elementos, es un operador que se calcula en base a probabilidades (valores entre cero y uno), permite la clasificación mediante una distribución categórica, se define matemáticamente como (Vilagran Solsona, 2018):

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=0}^K e^{z_k}} \quad \forall j \dots K$$

Para cada elemento de entrada a la función softmax se aplica una función exponencial y la normalización mediante la sumatoria de exponenciales, esto genera el resultado probabilístico y permite la posterior clasificación (Casado Beinat, 2022).

Tipos de capas en redes neuronales artificiales

Una red neuronal artificial se compone por diversas capas, es fundamental detallar de forma teórica el uso de las más importantes para el posterior diseño de las redes neuronales artificiales, a continuación, se desarrollarán tres tipos principales de capas que son: la capa de entrada, capa completamente conectada y capa dropout.

Capa de Entrada. La capa de entrada es aquella que toma los datos del exterior que servirán para el entrenamiento o validación de la red neuronal, existen distintos tipos de datos de entrada dependiendo del modelo de red y el aplicativo a la cual fue desarrollada (Andrade Tepán, 2013).

La respuesta de la red neuronal variará dependiendo de cómo fueron presentados los datos, ocasionalmente en la capa de entrada también se genera una normalización de los datos ingresados con la finalidad de mejorar la respuesta de la red y no generar variaciones por el incremento de los pesos y mantener en un rango común sin depender de la escala inicial (Andrade Tepán, 2013).

Capa Completamente Conectada. La capa completamente conectada (full connected layer) posee la característica de interconexión entre todas las neuronales de las capas anteriores y siguientes. El objetivo principal de esta capa es generar una asociación de toda la información recopilada por la red hasta el momento (Gimenez Arnal, 2018).

Las redes neuronales artificiales poseen varias capas completamente conectadas organizadas de forma sucesiva, para aplicaciones de clasificación la última capa completamente

conectadas se configura con el número de clases para que la red realice la discriminación entre las categorías (Bonilla Carrión, 2020).

Capa Dropout. La capa dropout es utilizada por las redes neuronales artificiales para prevenir el fenómeno denominado “sobreajuste” (de su traducción “overfitting”); este fenómeno origina que la red neuronal no sea capaz de generalizar la información aprendida y como consecuencia no brinde correctas predicciones ante nuevos datos de entrada (Artola Moreno, 2019).

El dropout es una técnica de regularización cuyo funcionamiento consiste en ignorar aleatoriamente ciertas neuronas durante el entrenamiento de la red; esto con el objetivo de que en cada entrenamiento las capas ocultas tengan diferentes conexiones, nodos y se tenga una visión distinta de la configuración de la red.

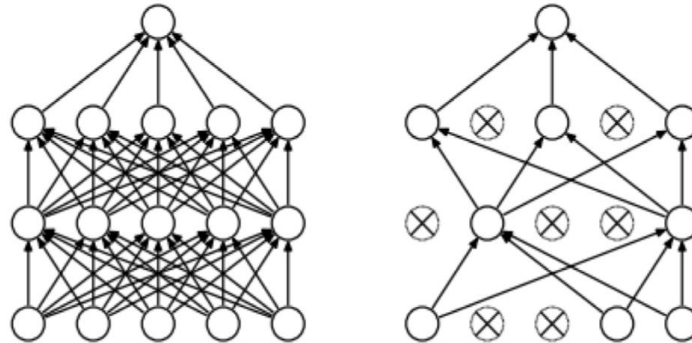
El dropout genera una mayor robustez al modelo, evita la dependencia de entre neuronas y previene el fenómeno del sobreajuste (Antona Castañares, 2020). Dado que el dropout posee un comportamiento aleatorio su valor se basa en una medida probabilística brindada por la distribución de Bernoulli, se calcula el “dropout rate” mediante la división del número de neuronas eliminadas sobre el número de neuronas totales. El valor probabilístico resultante debe estar dentro del intervalo de $[0, 1]$ (Antona Castañares, 2020).

$$\text{Dropout Rate} = \frac{\text{Neuronas Eliminadas}}{\text{Neuronas Totales}}$$

En la figura 5 se muestra de forma grafica el funcionamiento de la capa dropout, a la izquierda del grafico se tiene una red neuronal artificial en la cual todas las neuronas se encuentran interconectadas y a la derecha se evidencia el funcionamiento del dropour ignorando aleatoriamente ciertas neuronas de la red.

Figura 5

Funcionamiento de capa dropout



Nota. En la figura se muestra el funcionamiento de la capa dropout el cual ignora aleatoriamente ciertas neuronas en el entrenamiento de una red. Tomado de (Antona Castañares, 2020).

Una vez teniendo en consideración el funcionamiento de diversas capas en una red neuronal artificial resulta fundamental conocer el proceso de entrenamiento de una red.

Entrenamiento de Redes Neuronales Artificiales

Las redes neuronales artificiales tienen un proceso de entrenamiento y de validación del modelo, en el entrenamiento se genera el ajuste de los pesos con la finalidad de reducir el margen de error de los mismos. Para un correcto entrenamiento de la red neuronal este proceso se debe realizar en varios ciclos, a cada ciclo se le denomina época (García Sánchez, 2019). En el entrenamiento de la red neuronal se utilizan diferentes algoritmos de optimización detallados a continuación.

Algoritmos de Optimización. La función de un algoritmo de optimización es poder calcular el gradiente, el gradiente nos indica cómo modificar los parámetros de la red para minimizar la desviación o error a la salida por cada peso.

Existen diferentes algoritmos de optimización entre los principales están el algoritmo SGDM (Stochastic Gradient Descent with Momentum) y el algoritmo ADAM (Adaptive Moment Estimation) (Cabrera Mora, 2021).

Una vez entendido el funcionamiento de una red neuronal artificial de forma general, la siguiente subdivisión se enfoca en el desarrollo teórico de un tipo de red neuronal específica denominada red neuronal convolucional.

Redes Neuronales Convolucionales

Una red neuronal convolucional es un tipo de red neuronal que se caracteriza por el uso de diversas capas convolucionales en el procesamiento de la red; las redes neuronales convolucionales utilizan una estructura multicapa establecida de forma jerárquica, donde la salida de una capa oculta es el ingreso de otra capa (Lopez Pacheco, 2021).

Este tipo de redes habitualmente se utilizan para aplicaciones en imágenes y video como por ejemplo para la clasificación, reconocimiento de caracteres y patrones, análisis de imágenes en diferentes campos, procesamiento de series temporales, etc. Además, las redes neuronales convolucionales se clasifican según la dimensión, se tienen las redes unidimensionales y bidimensionales (Andrade Tepán, 2013).

En esta subdivisión se tratará: la estructura de las redes neuronales convolucionales, las redes convolucionales tanto unidimensionales como bidimensionales y conceptos de transfer learning utilizados para el diseño de las redes.

Estructura de las Redes Neuronales Convolucionales. Las redes neuronales convolucionales poseen en su estructura capas características como son: la capa convolucional, la capa de padding y la capa de pooling detalladas a continuación:

Operación de convolución y capa convolucional. La convolución es un operador matemático utilizado en diferentes aplicaciones y principalmente orientado a los sistemas lineales e invariantes en el tiempo (LTI) que permite conocer la respuesta de un sistema ante cualquier entrada, conociendo previamente la respuesta del sistema ante entradas conocidas como impulsos (Mejía, 2016).

El producto de la convolución genera una nueva función que se obtiene con la magnitud de la superposición de 2 funciones previas; en el dominio continuo calculado mediante la integral y en el dominio discreto mediante el uso de series, el operador de la convolución se expresa mediante el símbolo "*" (Percat Fedalto, 2015).

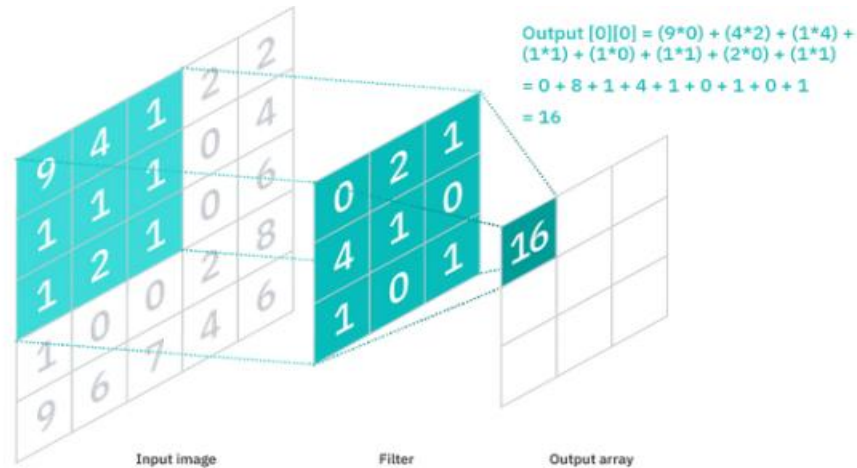
$$(f * g)_{(t)} = \int_0^t f(u)g(t - u)du; \quad \forall t > 0$$

$$x_n * y_n = \sum_{k=0}^n x_k y_{n-k}$$

La operación de la convolución se puede realizar con diversos tipos de datos como señales, imágenes, vectores, etc. En las telecomunicaciones el operador de la convolución se utiliza en aplicaciones como: el procesamiento de señales, el procesamiento de imágenes con el uso de filtros y en el aprendizaje profundo mediante las redes neuronales convolucionales (MathWorks, s. f.).

Para poder explicar el funcionamiento de la capa convolucional se tomará como ejemplo una imagen como dato de entrada a la red, sobre la imagen, que se representa como una matriz de píxeles, se aplica un filtro entrenable (también denominado kernel) que será una matriz de menor tamaño; el filtro realizará la sumatoria de la multiplicación de cada uno de los términos del filtro por cada término de la imagen, recorriendo desde la esquina superior izquierda, de izquierda a derecha y de arriba abajo en todo el ancho y alto del área de la imagen (convolución entre la imagen y el filtro) (Bonilla Carrión, 2020).

Figura 6

Operación de la Capa Convolutiva

Nota. En la figura se muestra la operación de la capa convolutiva entre una imagen de entrada y un filtro mostrando el resultado obtenido de dicho proceso. Tomado de (García Roque, 2022).

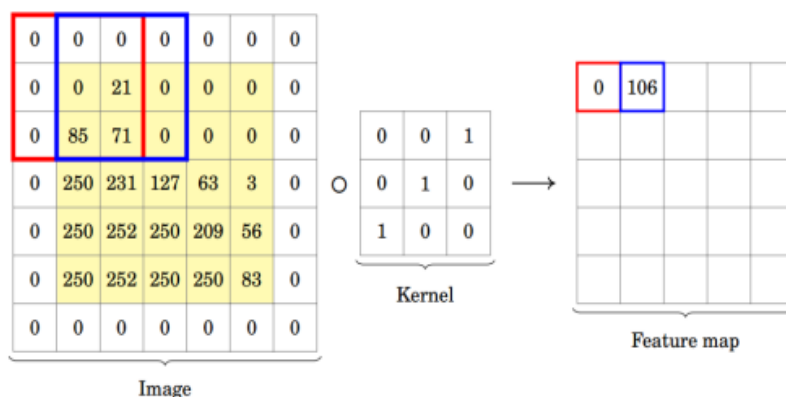
Este proceso de convolución genera el aprendizaje de la red, ya que la matriz resultante es un mapa de características que serán los datos de entrada para las siguientes capas de la red neuronal. Para aplicaciones orientadas a la clasificación, mediante la extracción de características se puede discriminar la categoría a la cual pertenece una imagen (Casado Beinat, 2022). Para una imagen de entrada de dimensión $[m \times m]$ se tiene un filtro de dimensiones $[n \times n]$, el resultado de la capa convolutiva genera una matriz característica de dimensiones $[m - n + 1] \times [m - n + 1]$ (Durán Suárez, 2017).

Padding. En la capa convolutiva la matriz resultante suele tender a perder píxeles de los bordes de la imagen, cuando el filtro aplicado es pequeño esta pérdida no es considerable pero conforme se aplica más capas convolutivas se puede acumular la pérdida de píxeles. Para solventar esto se utiliza una técnica denominada "padding" que consiste en rellenar los bordes de la imagen con ceros (Zhang et al., 2023).

El padding se utiliza con la finalidad de que el mapa de características resultante de la capa convolucional tenga la misma dimensión que la imagen de entrada y al volver a aplicar un filtro en otras capas convolucionales este mejore la extracción de características y el aprendizaje de la red (Erroz Arroyo, 2019).

Figura 7

Funcionamiento del padding



Nota. En la figura se muestra el funcionamiento del padding en una imagen rellenando de ceros los bordes de la imagen para aplicar el filtro o kernel. Tomado de (Erroz Arroyo, 2019)

Capa de Pooling. En redes neuronales convolucionales al tener una estructura jerárquica, se pueden usar varias capas convolucionales sucesivas dependiendo del diseño de la red; esto generaría un incremento exponencial en el procesamiento de la red, aumentó en el uso de recursos computacionales e incluso podría generar errores durante el aprendizaje (Artola Moreno, 2019).

Para solventar esto se hace uso de la capa pooling o también llamada capa de agrupación que se ubicaría entre las capas convolucionales cuya función es tomar la matriz de características resultante de la capa convolucional y reducir su tamaño conservando únicamente las

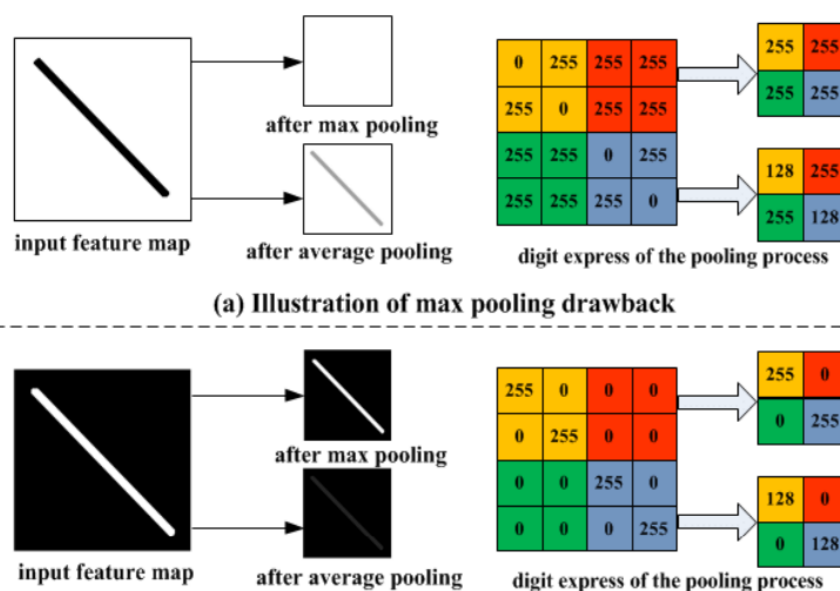
características más importantes detectadas por el filtro, disminuyendo la dimensionalidad, la carga computacional y previniendo el sobre ajuste de la red (Bonilla Carrión, 2020).

Existen dos tipos de pooling: el average-pooling y el max-pooling. Desde la esquina superior izquierda de la matriz de características se toma una matriz auxiliar cuadrada de menor dimensión y se realiza el mismo proceso que la capa convolucional, con la diferencia que en la capa pooling no existe un filtro sino únicamente se realiza un proceso de cálculo.

El cálculo en el average-pooling es obtener la media de los elementos de la matriz auxiliar y en el max-pooling se toma el elemento de mayor valor de la matriz auxiliar. Este proceso sigue el mismo recorrido como en la capa convolucional de izquierda a derecha y de arriba a abajo (Durán Suárez, 2017).

Figura 8

Funcionamiento de la capa pooling



Nota. En la figura se muestra el funcionamiento de la capa pooling tanto para el average-pooling como para el max-pooling. Tomado de (Durán Suárez, 2017).

Aparte de la estructura y las capas característica en una red neuronal convolucional es imperativo conocer que una red convolucional puede operar en diferentes dimensiones, a continuación, se van a describir las redes neuronales convolucionales unidimensionales, bidimensionales y una técnica denominada transfer learning que resulta de gran utilidad en estos tipos de redes.

Redes Neuronales Convolucionales Unidimensionales. Las redes neuronales unidimensionales son aquellas que realizan la operación de la convolución en una dimensión, sus aplicativos se orientan en procesamiento de datos seriales y en secuencias temporales. En la capa convolucional el filtro mantiene su funcionalidad realizando el recorrido de izquierda a derecha en todo el vector de entrada (Almansa Laredo, 2023).

Redes Neuronales Convolucionales Bidimensionales. Las redes neuronales bidimensionales se caracterizan por tener como datos de entrada imágenes, las cuales son matrices de píxeles de dos dimensiones. Las aplicaciones de las redes bidimensionales son muy variadas, principalmente porque se aprovecha el uso de imágenes para el procesamiento, la extracción de la información, clasificación, reconocimiento de patrones, identificación de objetos, entre otros (Bonilla Carrión, 2020).

Transfer Learning. Actualmente existen un conjunto de redes neuronales convolucionales que han sido previamente diseñadas, entrenadas y cuya estructura está disponible para orientarlas a varios aplicativos. Estas redes pre diseñadas tienen la característica de que pueden ser modificadas mediante técnicas como la transferencia de aprendizaje (transfer learning) con la finalidad de aprovechar la arquitectura de estos modelos ya desarrollados (Hernández González, 2020).

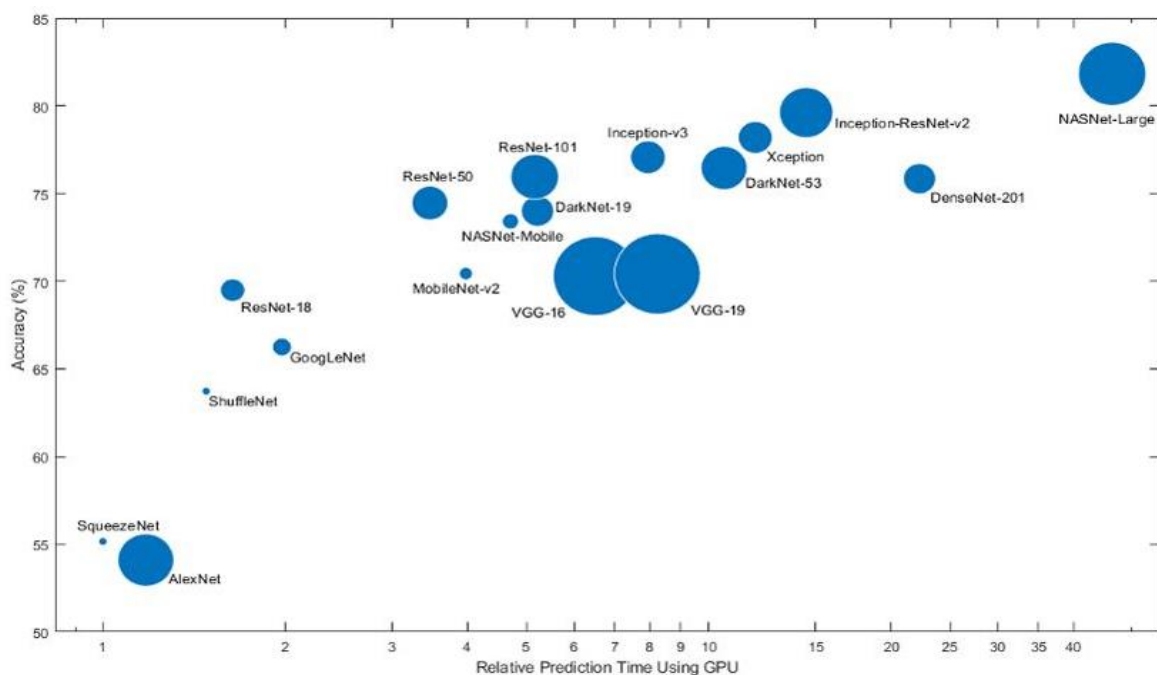
Transfer learning es una técnica que permite utilizar las redes neuronales pre entrenadas para reusar o modificar parte de su arquitectura con un nuevo conjunto de datos de entrada, esto

con la finalidad de disminuir tiempos de entrenamiento y recursos computacionales (Pérez Aguilar et al., 2021).

Existen una gran cantidad de redes pre entrenadas desarrolladas por varios investigadores en el campo del aprendizaje profundo, algunos puntos importantes para escoger el modelo adecuado se basan en: tipo de tarea a desarrollar, capacidad computacional, tamaño de la red prediseñada, precisión en la resolución, tiempo de predicción, entre otros (MATLAB - MathWorks, s. f.).

Figura 9

Redes neuronales convolucionales pre entrenadas



Nota. En la figura se muestra el conjunto de redes neuronales convolucionales pre entrenadas para su uso al público en general. Tomado de (MATLAB - MathWorks, s. f.).

De este conjunto de redes pre entrenadas se desarrollará una red convolucional bidimensional denominada Alexnet.

Alexnet. Es una red neuronal convolucional pre entrenada utilizada principalmente para la clasificación y reconocimiento de imágenes; dentro de las características de Alexnet se puede mencionar que posee ocho capas profundas, de las cuales cinco capas son convolucionales y tres capas totalmente conectadas (fully connected); además como datos de entrada de Alexnet se deben tener imágenes de dimensiones [227 x 227 x 3] (Serrano Moreno, 2018).

Alexnet posee la capacidad de poder clasificar imágenes en un conjunto de hasta mil categorías, esto por la configuración de la última capa totalmente conectada, dependiendo del aplicativo y utilizando técnicas de transferencia de aprendizaje (transfer learning) esta última capa profunda podrá ser modificada para la cantidad de clases que se necesiten (MathWorks - MATLAB, s. f.). En la figura 23 se evidencia el uso de transfer learning para el diseño de una red neuronal convolucional bidimensional.

Una vez finalizado el apartado de las redes neuronales artificiales es fundamental avanzar a la sección final que trata sobre las matrices de confusión y los parámetros de rendimiento ya que esta matriz es el resultado de la clasificación de la red neuronal convolucional a diseñar. Resulta crucial realizar una correcta interpretación de los resultados de la matriz de confusión ya que, en base a estos, se realizará un análisis de los parámetros de rendimiento en la sección de resultados.

Matriz de Confusión y Parámetros de Rendimiento

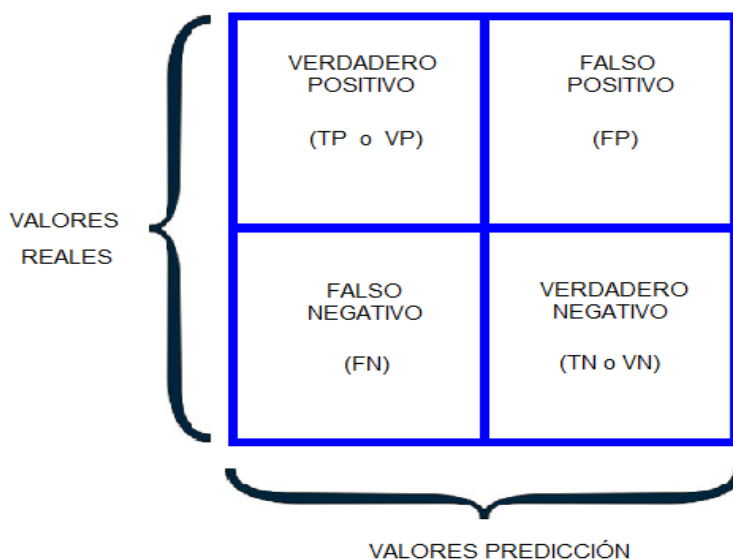
La matriz de confusión es una herramienta visual que se utiliza para resumir el comportamiento y desempeño de un sistema de clasificación (Artola Moreno, 2019). Dicha matriz mostrará el resultado obtenido del entrenamiento y validación de una red neuronal donde los valores en el eje X (columnas de la matriz) son las clases predichas y los valores del eje Y (filas de la matriz) son las clases reales. Para una matriz de clasificación binaria se tienen los siguientes elementos (Campos Mocholí, 2020):

- Verdaderos Positivos (VP o TP): Cantidad de valores positivos clasificados correctamente por la red.
- Verdaderos Negativos (VN o TN): Cantidad de valores negativos clasificados correctamente por la red.
- Falsos Positivos (FP): Cantidad de valores negativos clasificados incorrectamente como positivos por la red.
- Falsos Negativos (FN): Cantidad de valores positivos clasificados incorrectamente como negativos por la red (González Montoro, 2021).

En la figura 10 se muestra el grafico de una matriz de confusión orientada a la clasificación.

Figura 10

Matriz de Confusión



Con los resultados de la matriz de confusión del modelo se pueden calcular los parámetros de rendimiento del sistema de clasificación, detallados en la figura 11 (Telefónica Tech, 2021).

Figura 11

Parámetros de Rendimiento de Matriz de Confusión

| Matriz de confusión | | Estimado por el modelo | | | |
|---------------------|----------|--|---|---|-----------|
| | | Negativo (N) | Positivo (P) | | |
| Real | Negativo | a: (TN) | b: (FP) | Precisión ("precision") Porcentaje predicciones positivas correctas: | $d/(b+d)$ |
| | Positivo | c: (FN) | d: (TP) | | |
| | | Sensibilidad, exhaustividad ("Recall") Porcentaje casos positivos detectados | Especificidad ("Specificity") Porcentaje casos negativos detectados | Exactitud ("accuracy") Porcentaje de predicciones correctas (No sirve en datasets poco equilibrados) | |
| | | $d/(d+c)$ | $a/(a+b)$ | $(a+d)/(a+b+c+d)$ | |

Nota. En la figura se muestran los parámetros de rendimiento de una matriz de confusión, para el cálculo de exactitud, sensibilidad, especificidad y precisión. Tomado de (Telefónica Tech, 2021).

Capítulo III: Metodología

Descripción General

Para el desarrollo del presente trabajo se hace uso de un software simulador de señales de radar aprobado por el Centro de Investigación de Aplicaciones Militares (CIAM), el sistema de radar OERLIKON posee un director de tiro SKYGUARD el cual realiza un barrido de los 360 grados correspondiente a una señal de radar, este barrido da como resultado un conjunto de 6400 muestras con componente en fase y cuadratura de las cuales 43 muestras representan la presencia de un objetivo a diferentes niveles de relación señal a ruido (Lara Mina & Ortiz Moya, 2019).

Mediante el simulador de señales de radar se realizará el diseño e implementación de redes neuronales convolucionales unidimensionales y bidimensionales para su posterior comparativa; las redes neuronales se han desarrollado en el software MATLAB®2022b con el uso de herramientas como “Deep Network Designer” y comandos como “analyzeNetwork” para el diseño, entrenamiento y validación de las redes convolucionales.

El hardware utilizado para la simulación de señales de radar y el entrenamiento de la red es un ordenador y su unidad de procesamiento central (CPU) ya que no se dispone de una unidad de procesamiento gráfico (GPU) que acorta el tiempo de entrenamiento de las redes neuronales.

Señales De Radar Simuladas

En la señal de radar simulada el objetivo se encuentra ubicado de forma aleatoria, el cual a un mayor nivel de SNR es más sencillo identificarlo, pero a menor nivel de SNR se incrementa la dificultad de poder discriminar si la señal obtenida posee o no un objetivo.

La simulación de las señales de radar se las realiza desde un SNR de -5dBs hasta 0 dBs con paso de 1 dBm y desde 0 dBs hasta 15 dBs con paso de 5 dBms, en los niveles de SNR más altos se utiliza un mayor intervalo de separación dado que resulta más sencillo para el ser

humano identificar la presencia del objetivo en la señal, pero es importante de igual forma observar el comportamiento del sistema a implementar, identificando si presenta o no variaciones en los resultados obtenidos en estos niveles; los diferentes conjuntos de datos serán utilizados en el entrenamiento y validación de las redes neuronales convolucionales a diseñar. En la figura 12 se muestra de forma gráfica como un sistema de radar realizar el barrido de las muestras y detecta un objetivo y en la figura 13 se refleja el vector resultante del barrido del simulador de señales de radar, y la cantidad de muestras que representan al objetivo.

Figura 12

Ejemplo del funcionamiento de un radar y localización de un objetivo



Figura 13

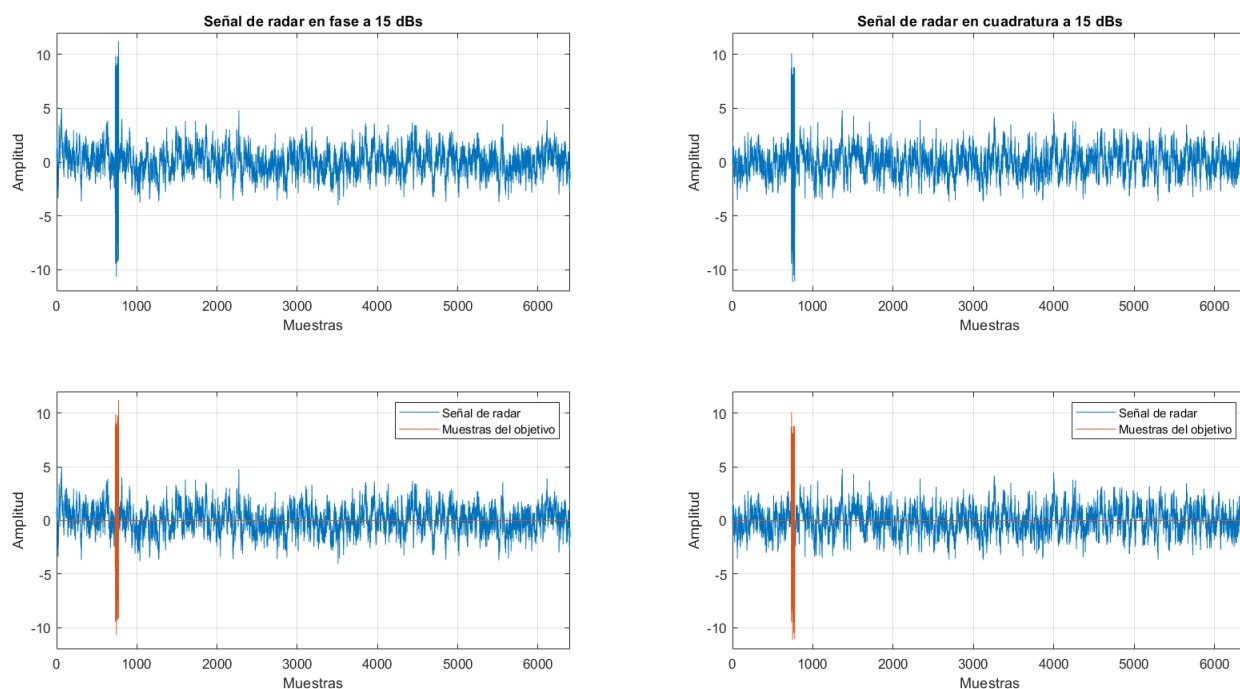
Vector resultante del barrido del sistema simulador de señales de radar



A continuación, se muestra las gráficas de los componentes en fase y cuadratura de una señal de radar simuladas a una relación señal a ruido de 15 dBs, se puede evidenciar que resulta sencillo la identificación del objetivo a un SNR alto.

Figura 14

Componentes en fase y cuadratura de una señal de radar a una SNR de 15 dBs



La complejidad para la identificación del objetivo en la señal de radar aumenta conforme disminuye el nivel de la relación señal a ruido. Por ejemplo, en la figura 15 se muestra la presencia de un objetivo a un SNR de 0 dBs, en comparación a la figura 14 con SNR de 15 dBs se incrementa la complejidad de la identificación del objetivo.

Finalmente, en la figura 16 se representa la señal de radar en el SNR más bajo de -5 dBs, en este nivel de SNR en comparación a los anteriores niveles más altos se observa la mayor complejidad para la identificación del objetivo y se evidencia que mediante el desarrollo de redes

neuronal se puede obtener grandes resultados para una correcta discriminación y con esto mejorar el desempeño de radares actualmente existentes.

Figura 15

Componentes en fase y cuadratura de una señal de radar a una SNR de 0 dBs

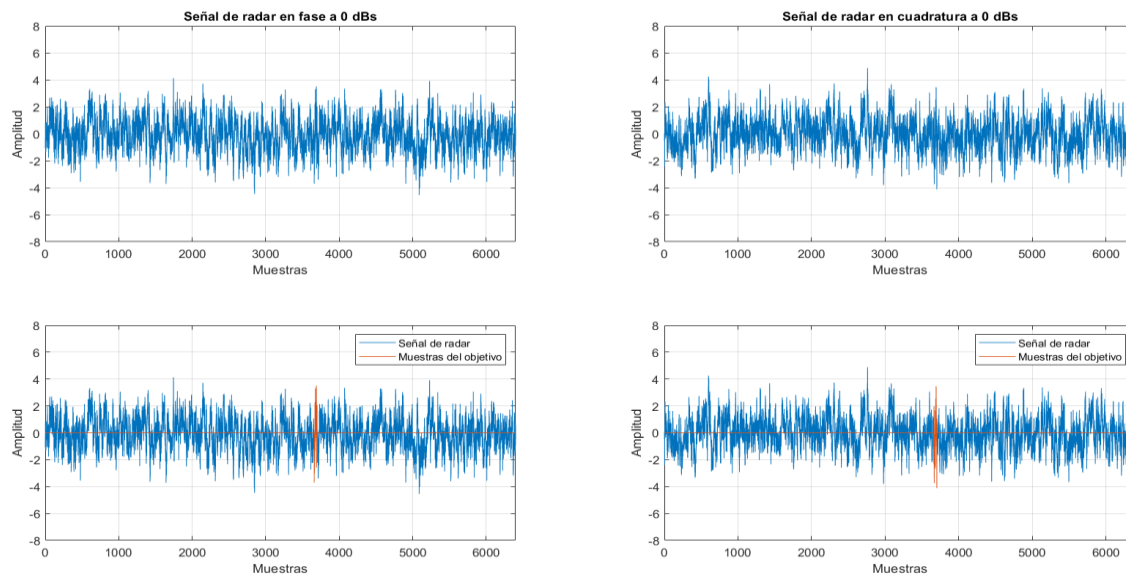
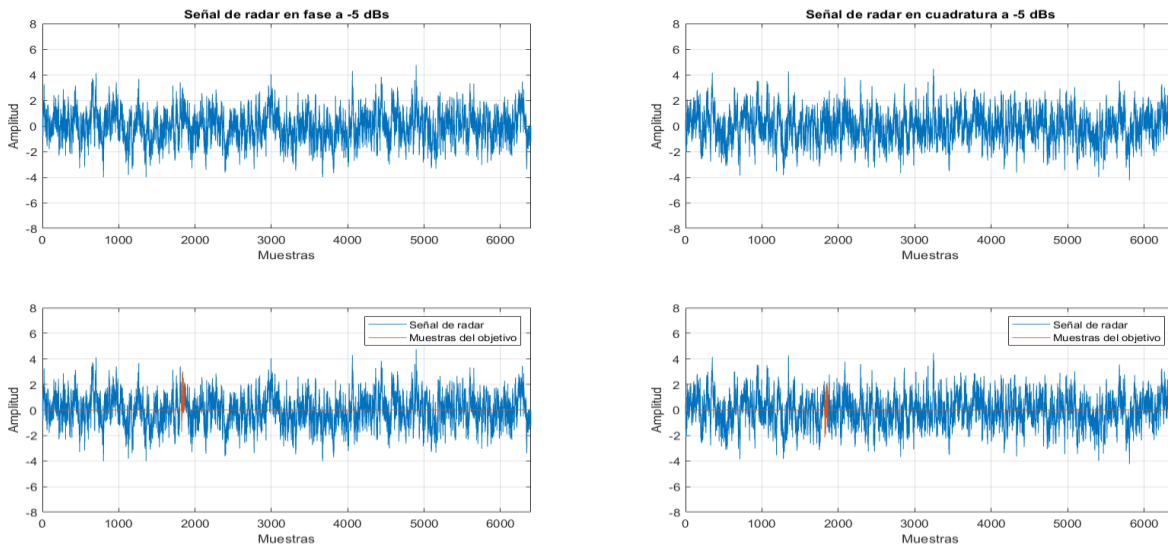


Figura 16

Componentes en fase y cuadratura de una señal de radar a una SNR de -5 dBs



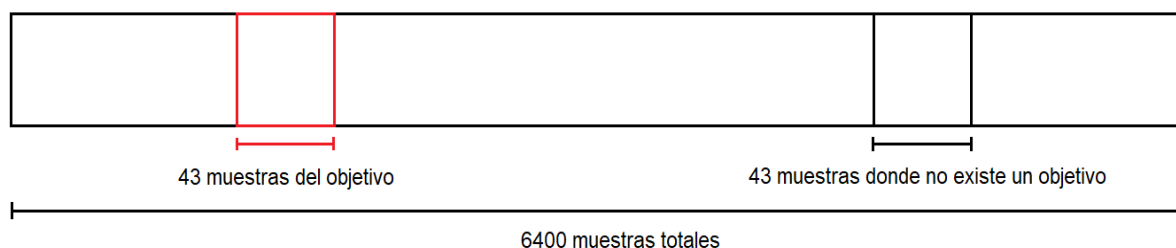
Preparación de Bases De Datos

Una vez comprendido el funcionamiento del software simulador de señales de radar se deben crear los conjuntos de datos para el entrenamiento y validación de las redes neuronales convolucionales, para las redes convolucionales unidimensionales se han utilizado un conjunto de 2 millones de señales de radar y para las redes bidimensionales un conjunto de 100 mil señales de radar, de estos datos un 70% serán destinados para el entrenamiento de las redes convolucionales y el 30% restante para la validación del funcionamiento, esto a diferentes niveles de relación señal a ruido. Ambas redes convolucionales realizan la función de clasificación de la señal de entrada para validar si existe o no la presencia de un objetivo.

Del conjunto de muestras totales de la señal de radar se tomarán 2 subconjuntos de datos, las 43 muestras donde existe la presencia de un objetivo y de forma aleatoria otras 43 muestras donde no exista la presencia del objetivo. En la figura 17 se muestra gráficamente los subconjuntos utilizados donde existe y no la presencia del objetivo.

Figura 17

Subconjuntos de muestras tomadas de la señal de radar.

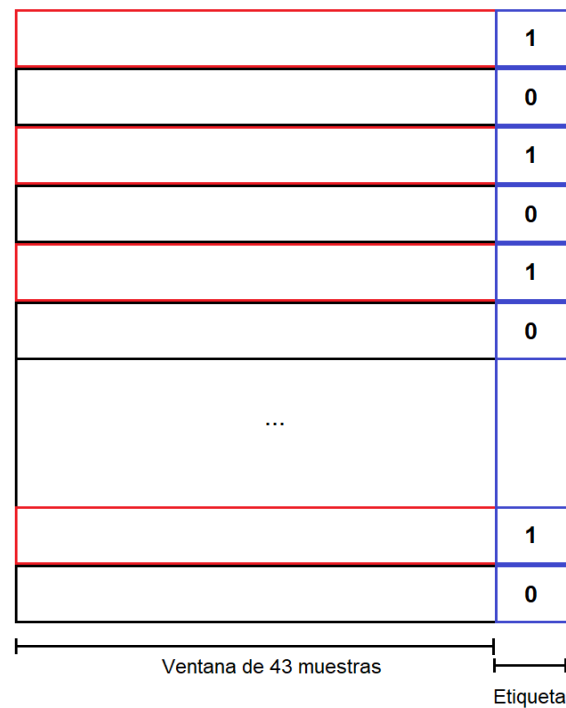


Para la identificación de los subconjuntos de datos, a las muestras que poseen la presencia de un objetivo se les asignará una etiqueta con un valor de 1 y a las muestras que no poseen el objetivo se les etiquetará con 0. Ambos vectores formarán parte de una matriz de datos y se alternarán uno seguido del otro, como se muestra en la figura 18 donde se observa la matriz

resultante la cual consta de los vectores de 43 muestras complejas y la respectiva etiqueta donde 1 representa que existe la presencia de un objetivo y 0 representa que no existe la presencia de un objetivo

Figura 18

Asignación de etiquetas a los vectores para la identificación de la presencia de objetivos

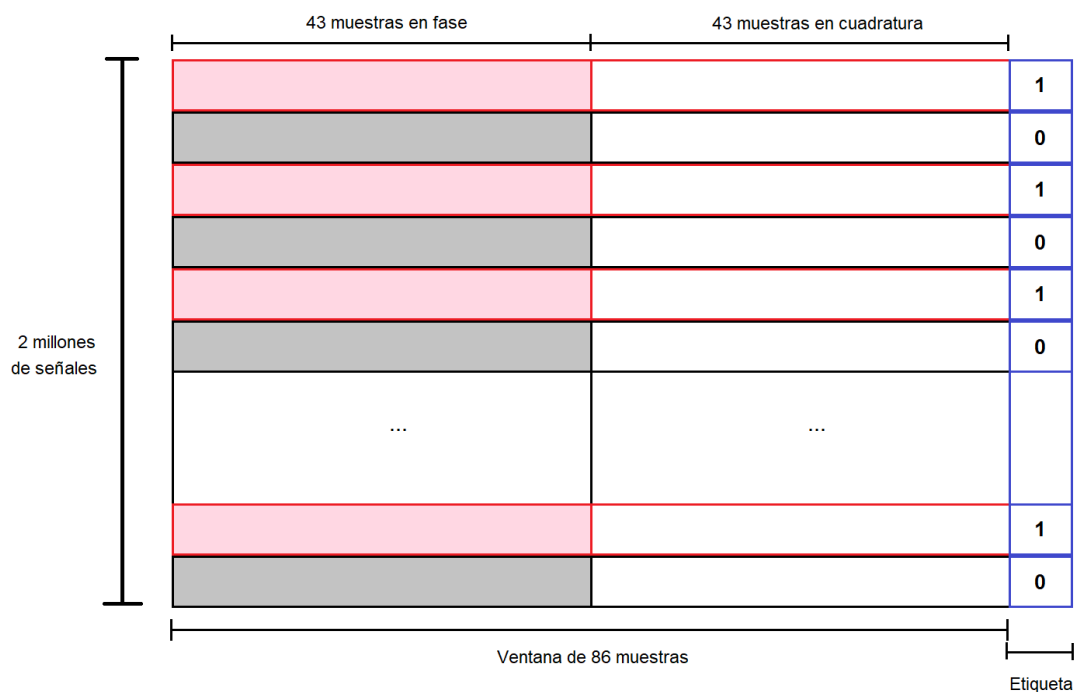


Bases de datos para red neuronal convolucional unidimensional

Para las redes convolucionales unidimensionales se tendrá como datos de entrada un vector. Dado que la ventana de 43 muestras de la señal de radar posee componentes en fase y cuadratura, se separa la parte real y parte imaginaria y se concatenaran en un solo vector resultante de 86 muestras, esto con la finalidad de utilizar la mayor cantidad de información de la señal de radar, junto con la respectiva etiqueta que muestra la presencia o no de un objetivo, como se muestra en la figura 19. Para las redes convolucionales unidimensionales se utilizarán un conjunto de 2 millones de señales de radar que serán los datos de entrada de la red.

Figura 19

Base de datos para red neuronal convolucional unidimensional



Bases de datos para red neuronal convolucional bidimensional

Las redes neuronales convolucionales bidimensionales deben tener como datos de entrada imágenes; dado que tenemos una matriz de señales de radar con componentes en fase y cuadratura y etiquetadas con la presencia de los objetivos, se crea una imagen mediante un espectrograma que hace uso de la transformada de Fourier y permite representar de forma visual los componentes en frecuencia de una señal de entrada.

Mediante el espectrograma se crean un conjunto de 100 mil imágenes que servirán para el entrenamiento y validación de la red convolucional bidimensional. En la figura 20 se muestra un ejemplo de una imagen resultante al usar la función espectrograma de una señal de radar.

Para una red neuronal convolucional cada pixel de la imagen representa una gran cantidad de información, por ese motivo se realiza un reajuste del área del espectrograma

original, eliminando información innecesaria y aprovechar de mejor manera los datos de entrada para el modelo. Este reajuste puede evidenciarse en la figura 21.

Figura 20

Espectrograma de una señal de radar simulada

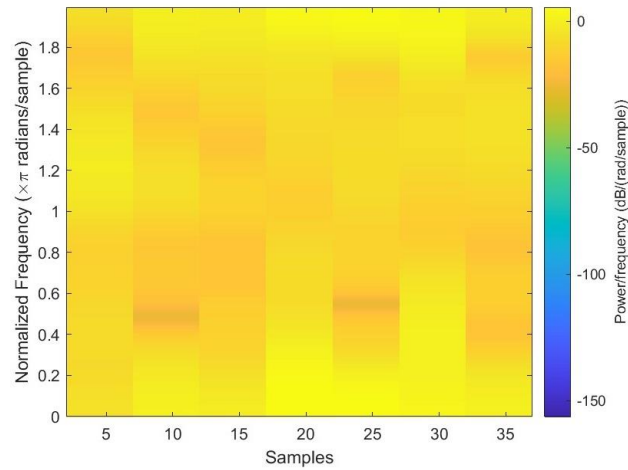
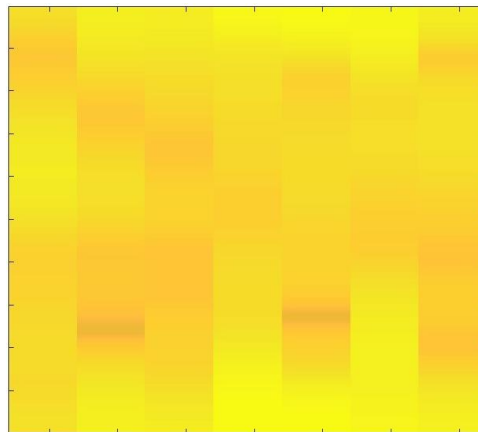


Figura 21

Re ajuste del área de espectrograma.



Diseño de Redes Neuronales Convolucionales

El apartado del diseño de las redes neuronales convolucionales se subdividirá en el diseño para las redes unidimensionales y para las redes bidimensionales

Diseño de red convolucional unidimensional

Realizar el diseño de una red neuronal convolucional conlleva un conjunto de varios factores a considerar, por ejemplo: analizar el problema a solucionar, escoger correctamente una arquitectura y el número de capas a utilizar, dimensionar adecuadamente las capas convolucionales, hacer uso de capas de pooling, dropout, fully connected, etc. Como consecuencia de un incorrecto diseño se pueden presentar fenómenos como: sobreajuste, baja exactitud y malos parámetros de rendimiento, tiempos de procesamiento computacional excesivamente altos, entre otros.

Es crucial mediante el análisis y la guía de modelos orientados a diferentes aplicativos realizar un correcto diseño de las redes neuronales convolucionales. Para la red unidimensional se probaron diversas arquitecturas y organizaciones de capas, de las cuales la que mejores reflejó fue una arquitectura compuesta por 27 capas. En el proceso de diseño se tuvo que solucionar el fenómeno del sobreajuste, y dimensionar correctamente la cantidad de filtros y el tamaño de los mismos ya que en las primeras redes diseñadas reflejaban una baja exactitud sobre todo en los niveles de relación señal a ruido más bajos.

Como datos de entrada a la red diseñada se tendrán los vectores de 86 muestras con componentes en fase y cuadratura concatenados de la señal de radar; se realizaron pruebas entrenando con los componentes solo de fase y cuadratura separados y con el cálculo de la amplitud, pero los resultados reflejaron ser muy inferiores al que se logró con el entrenamiento de los componentes concatenados; la red neuronal diseñada se compone de 4 capas convolucionales de 32 filtros cuyas dimensiones son de 10 y 5 muestras.

Se hizo uso de las capas dropout, las capas completamente conectadas (fully connected) y la función de activación ReLu. Para realizar la clasificación que identifica si existe o no la

presencia de un objetivo, la última capa fully connected se configura con el número de clases (2 clases) junto con la función softmax. En resultado del diseño se muestra en la figura 22:

Figura 22

Arquitectura de la red neuronal convolucional unidimensional



Diseño de red convolucional bidimensional

Para el diseño de la red convolucional bidimensional uno de los más grandes retos a afrontar fue la cantidad de imágenes con las cuales se generaría el entrenamiento y validación de la red, la función espectrograma crea la imagen con los componentes en frecuencia de una señal de radar mediante la transformada de Fourier, pero el proceso de la creación y recopilación de imágenes produce un alto costo computacional originando un gran incremento en el tiempo empleado únicamente en la obtención de las bases de datos.

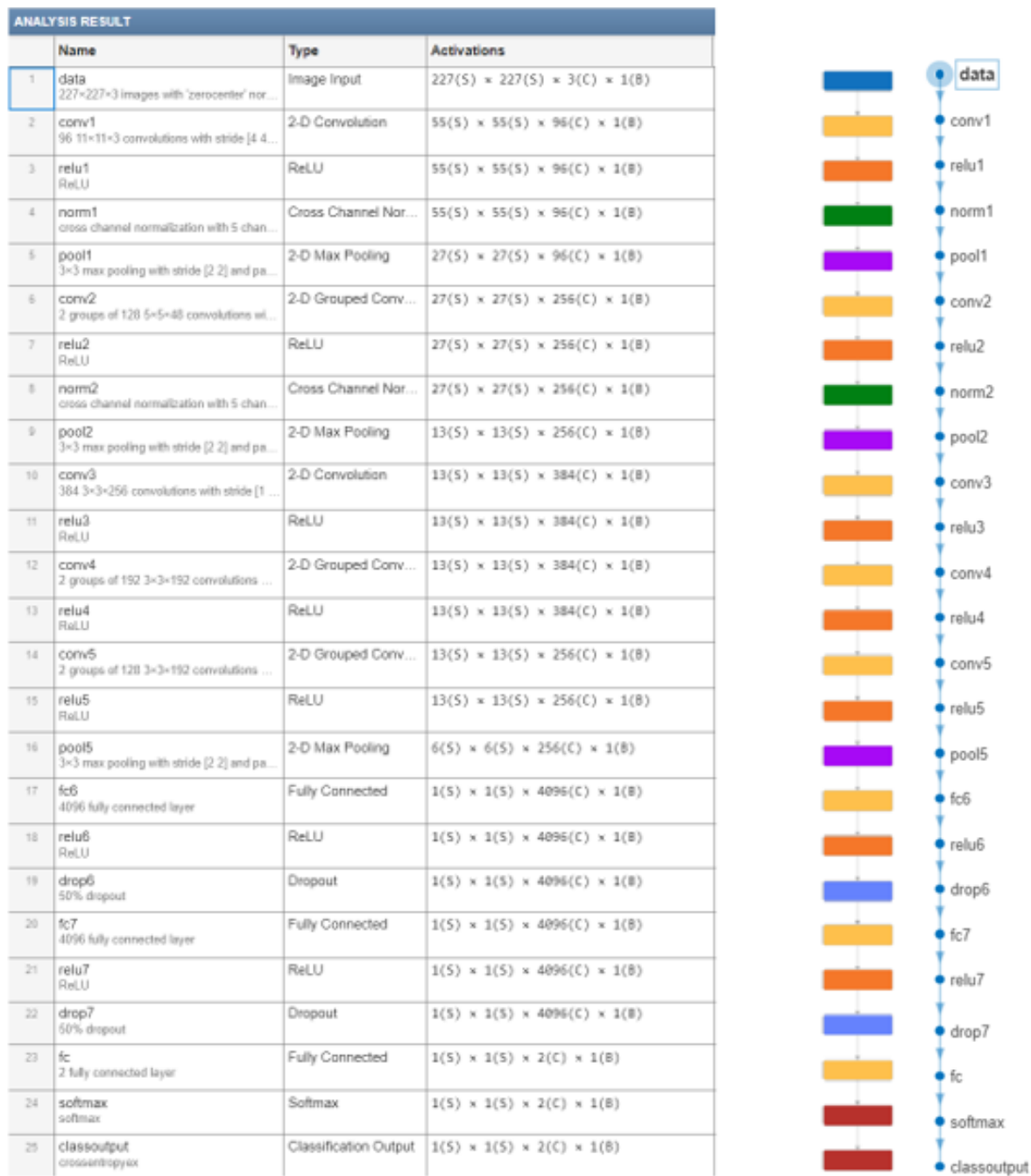
Para solucionar esto es importante optimizar los algoritmos generadores de imágenes con la finalidad de aprovechar al máximo los recursos computacionales disponibles, esto se lo puede lograr mediante: la mejora de la estructura del código programado, el uso de bibliotecas especializadas para desarrollar ciertos procesos, la eliminación de memoria caché de aplicaciones no utilizadas y en casos específicos se puede realizar el cambio de hardware a uno más actualizado.

Para la red convolucional bidimensional se utiliza la arquitectura de la red pre entrenada Alexnet, junto con la técnica de transferencia de aprendizaje (transfer learning). Alexnet está compuesta por una estructura de 25 capas, de las cuales se modificará la capa número 23, la cual es una capa completamente conectada (fully connected) y se configurará con el número de clases (2 clases) para la realizar la clasificación si existe o no la presencia de un objetivo.

Las imágenes de entrada de Alexnet deben configurarse, para que se ajusten a las dimensiones que la red pre entrenada acepta [227 x 227 x 3], el número 3 representa los canales RGB para imágenes a color. En la figura 23 se muestra la arquitectura completa de la red convolucional bidimensional implementada para la identificación de los objetivos en las señales de radar.

Figura 23

Arquitectura de la red neuronal convolucional bidimensional



Capítulo IV: Resultados

Resultados Red Neuronal Convolutacional Unidimensional

En la red neuronal unidimensional se utilizaron un conjunto de 2 millones de datos donde el 70% se destinaron para el entrenamiento (1.4 millones de señales) y el 30% restante para validación de la red (600 mil señales). Todo este proceso se desarrolló en un conjunto de 30 épocas.

Con el conjunto de datos de validación se crea una matriz de confusión cuyos resultados permitirán realizar el cálculo de los parámetros de rendimiento de la red convolutacional implementada para la identificación de los objetivos a diferentes niveles de relación señal a ruido. En la figura 24 se muestra el proceso de entrenamiento y en la figura 25 se evidencia la matriz de confusión resultante de la red neuronal convolutacional unidimensional para el mismo SNR de -5 dBs.

Figura 24

Ejemplo de entrenamiento de la red convolutacional unidimensional

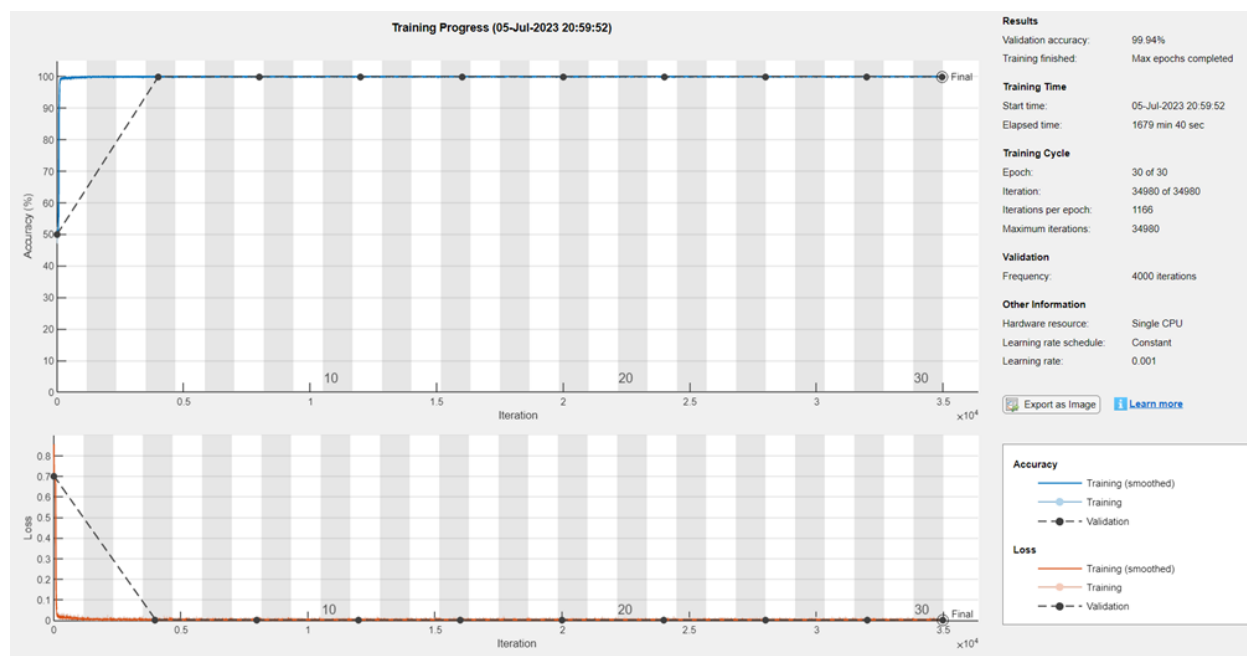


Figura 25

Ejemplo del resultado de matriz de confusión la red convolucional unidimensional

| | | Confusion Matrix | | |
|--------------|-------------|------------------|-----------------|----------------|
| | | NO OBJETIVO | SI OBJETIVO | |
| Output Class | NO OBJETIVO | 299773 50.0% | 138 0.0% | 100.0% 0.0% |
| | SI OBJETIVO | 227 0.0% | 299862 50.0% | 99.9% 0.1% |
| | | NO OBJETIVO | SI OBJETIVO | |
| | | Target Class | | |

En la tabla 2 se realiza un resumen de los resultados de entrenamiento y validación de la red neuronal convolucional unidimensional para los diferentes niveles de SNR. De esta tabla se puede identificar que la red convolucional unidimensional implementada logró el 100% de exactitud para la identificación de objetivos desde un SNR de -2 dBs en adelante; en el nivel más bajo de -5 dBs se alcanzó un 99,9391666% de exactitud lo que representa un buen resultado para la identificación de objetivos. Se realizó el cálculo promedio del tiempo que se demoró la red en el entrenamiento y validación del sistema y dio como resultado un promedio de 2076 minutos y 37 segundos (34 horas 36 min 37 segundos).

Tabla 2

Resumen de los resultados de entrenamiento y validación de la red neuronal convolucional unidimensional

| ÉPOCA | SNR (dBs) | EXACTITUD % | TIEMPO |
|--------------|------------------|--------------------|-----------------|
| 30 | -5 dB | 99,939166 % | 1679 min 40 seg |
| | -4 dB | 99,989833 % | 1758 min 18 seg |
| | -3 dB | 99,997333 % | 2134 min 43 seg |
| | -2 dB | 100,00000 % | 1958 min 33 seg |
| | -1 dB | 100,00000 % | 1975 min 41 seg |
| | 0 dB | 100,00000 % | 3539 min 36 seg |
| | 5 dB | 100,00000 % | 1945 min 29 seg |
| | 10 dB | 100,00000 % | 1815 min 21 seg |
| | 15 dB | 100,00000 % | 1885 min 55 seg |

En la tabla 3 se muestra los resultados de la matriz de confusión con los 600 mil datos destinados para la validación de la red, en los diferentes niveles de SNR.

Mediante los datos obtenidos de las matrices de confusión se realiza el cálculo de los parámetros de rendimiento mostrados en la tabla 4. De igual forma en los parámetros de rendimiento se logra como resultado un 100% desde el nivel de SNR -2 dB en adelante.

De los parámetros de rendimiento obtenidos en la tabla 4 se podría definir a la sensibilidad como el porcentaje de casos positivos detectados, aplicado a nuestro sistema de detección de objetivos este parámetro indica la cantidad de detecciones acertadas de un objetivo en el caso de existir, en el SNR más bajo se obtuvo un 99,954%.

Tabla 3

Resultados de la matriz de confusión de la red neuronal convolucional unidimensional

| SNR | TN | FP | FN | TP |
|------------|-----------|-----------|-----------|-----------|
| -5 | 299773 | 227 | 138 | 299862 |
| -4 | 299961 | 39 | 22 | 299978 |
| -3 | 299992 | 8 | 8 | 299992 |
| -2 | 300000 | 0 | 0 | 300000 |
| -1 | 300000 | 0 | 0 | 300000 |
| 0 | 300000 | 0 | 0 | 300000 |
| 5 | 300000 | 0 | 0 | 300000 |
| 10 | 300000 | 0 | 0 | 300000 |
| 15 | 300000 | 0 | 0 | 300000 |

Otro parámetro a interpretar es la especificidad del sistema ya que muestra el porcentaje de casos negativos detectados, en el sistema de detección la cantidad de detecciones acertadas de un objetivo en el caso de no existir, en el SNR más bajo se obtuvo un 99,9243%.

Mediante estos parámetros se puede calcular la probabilidad de falta alarma en la detección, definido matemáticamente como:

$$\text{Probabilidad de falsa alarma\%} = 100\% - \text{Especificidad}$$

Realizando este cálculo para el SNR más bajo de -5 dBs se obtuvo el siguiente resultado:

$$\text{Probabilidad de falsa alarma\%} = 100\% - 99,924333333\%$$

$$\text{Probabilidad de falsa alarma\%} = 0,07566667\%$$

El resultado indica que la probabilidad de recibir una falsa alarma por el sistema de detección desarrollado en la red neuronal convolucional unidimensional es del 0,07566667% que representa un valor muy bajo y dentro de los parámetros indicados para el aplicativo.

Tabla 4

Parámetros de rendimiento de la red neuronal convolucional unidimensional

| SNR | PRECISIÓN | SENSIBILIDAD | ESPECIFICIDAD | EXACTITUD |
|------------|------------------|---------------------|----------------------|------------------|
| -5 | 99,924355% | 99,954000% | 99,92433% | 99,9391666% |
| -4 | 99,987000% | 99,992666% | 99,987000% | 99,9898333% |
| -3 | 99,9973333% | 99,9973333% | 99,9973333% | 99,9973333% |
| -2 | 100,0000% | 100,0000% | 100,0000% | 100,0000% |
| -1 | 100,0000% | 100,0000% | 100,0000% | 100,0000% |
| 0 | 100,0000% | 100,0000% | 100,0000% | 100,0000% |
| 5 | 100,0000% | 100,0000% | 100,0000% | 100,0000% |
| 10 | 100,0000% | 100,0000% | 100,0000% | 100,0000% |
| 15 | 100,0000% | 100,0000% | 100,0000% | 100,0000% |

Resultados Red Neuronal Convolucional Bidimensional

Para la red neuronal bidimensional se utilizaron un conjunto de 100 mil imágenes de las cuales un 70% se destinaron para el entrenamiento (70 mil imágenes) y el 30% restante para validación de la red bidimensional (30 mil imágenes). Todo este proceso se desarrolló en un conjunto de 15 épocas. En la figura 26 se muestra el proceso de entrenamiento de la red neuronal convolucional bidimensional para el SNR más bajo de -5 dBs y en la figura 27 se evidencia la matriz de confusión resultante de la red neuronal convolucional unidimensional para el mismo nivel de SNR

Figura 26

Ejemplo de entrenamiento de la red convolucional bidimensional

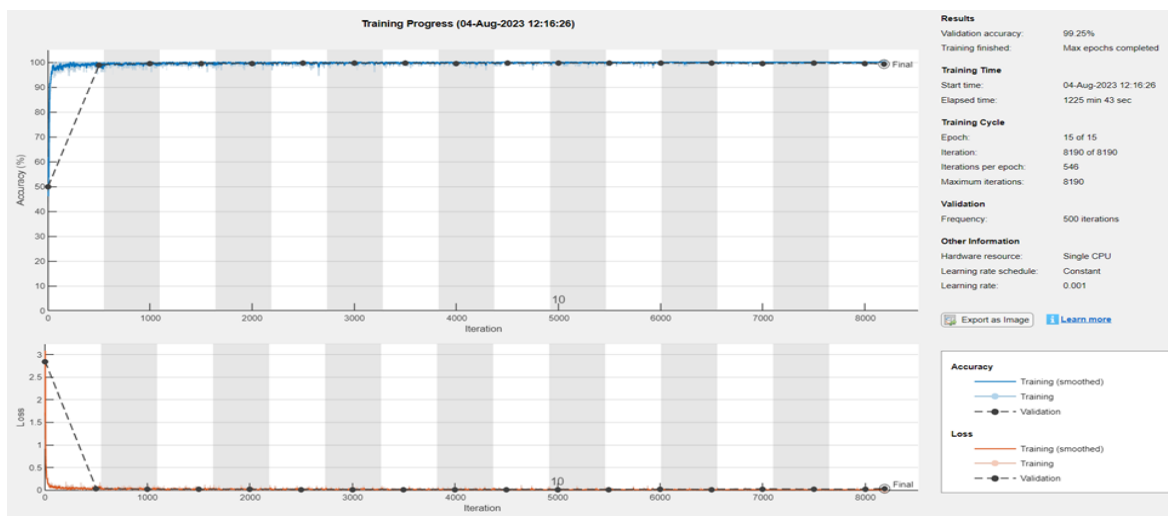


Figura 27

Ejemplo del resultado de matriz de confusión la red convolucional bidimensional.

| | | Confusion Matrix | | |
|--------------|-------------|------------------|----------------|---------------|
| | | NO OBJETIVO | SI OBJETIVO | |
| Output Class | NO OBJETIVO | 14791 49.3% | 15 0.1% | 99.9% 0.1% |
| | SI OBJETIVO | 209 0.7% | 14985 50.0% | 98.6% 1.4% |
| | | NO OBJETIVO | SI OBJETIVO | |
| | | 98.6% 1.4% | 99.9% 0.1% | 99.3% 0.7% |
| | | Target Class | | |

En la tabla 5 se realiza un resumen de los resultados de entrenamiento y validación de la red neuronal convolucional bidimensional para los diferentes niveles de SNR:

Tabla 5

Resumen de los resultados de entrenamiento y validación de la red neuronal convolucional bidimensional.

| ÉPOCAS | SNR (dBs) | EXACTITUD % | TIEMPO |
|--------|-----------|--------------|----------------|
| 15 | -5 | 99,2533333% | 1225min 43 seg |
| | -4 | 99.8600000% | 1228min 31 seg |
| | -3 | 99,9333333% | 1223min 49 seg |
| | -2 | 99,9933333% | 1224min 46 seg |
| | -1 | 100.0000000% | 1216min 36 seg |
| | 0 | 100.0000000% | 1222min 2 seg |
| | 5 | 100.0000000% | 1270min 49 seg |
| | 10 | 100.0000000% | 1272min 37 seg |
| | 15 | 100.0000000% | 1240min 46 seg |

Nota. En la tabla se realiza un resumen de los resultados obtenidos por la red neuronal convolucional bidimensional.

En la tabla 5 se muestran los resultados obtenidos por la red bidimensional, se observa que en el nivel de SNR de -1 dBs en adelante la red neuronal llegó al 100% de exactitud para la identificación de objetivos en señales de radar. El tiempo promedio en el entrenamiento y validación del sistema fue de 1235 minutos y 37 segundos (20 horas 35 minutos y 37 segundos).

En la tabla 6 se muestran los resultados de las matrices de confusión en los diferentes niveles de SNR:

Tabla 6

Resultados de la matriz de confusión de la red neuronal convolucional bidimensional

| SNR (dBs) | TN | FP | FN | TP |
|------------------|-----------|-----------|-----------|-----------|
| -5 | 14791 | 209 | 15 | 14965 |
| -4 | 14990 | 10 | 32 | 14968 |
| -3 | 14983 | 17 | 3 | 14997 |
| -2 | 14999 | 1 | 1 | 14999 |
| -1 | 15000 | 0 | 0 | 15000 |
| 0 | 15000 | 0 | 0 | 15000 |
| 5 | 15000 | 0 | 0 | 15000 |
| 10 | 15000 | 0 | 0 | 15000 |
| 15 | 15000 | 0 | 0 | 15000 |

Y mediante las matrices de confusión se puede realizar el cálculo de los parámetros de rendimiento de la red implementada, evidenciados en la tabla 7.

De los resultados objetivos por los parámetros de rendimiento es importante calcular la probabilidad de falsa alarma del sistema:

$$\text{Probabilidad de falsa alarma\%} = 100\% - \text{Especificidad}$$

Realizando este cálculo para el SNR más bajo de -5 dBs se obtuvo el siguiente resultado:

$$\text{Probabilidad de falsa alarma\%} = 100\% - 98,606666667\%$$

$$\text{Probabilidad de falsa alarma\%} = 1.3933333\%$$

El resultado indica que la probabilidad de recibir una falsa alarma por el sistema de detección desarrollado en la red neuronal convolucional bidimensional es del 1.3933333% que representa de igual forma un valor dentro de los rangos adecuados para el aplicativo de clasificación desarrollado.

Tabla 7

Parámetros de rendimiento de la red neuronal convolucional bidimensional

| SNR | PRECISIÓN | SENSIBILIDAD | ESPECIFICIDAD | EXACTITUD |
|------------|------------------|---------------------|----------------------|------------------|
| -5 | 98,62264% | 99,89986% | 98,60666% | 99,252835% |
| -4 | 99,93323% | 99,78666% | 99,93333% | 99,860000% |
| -3 | 99,88677% | 99,98000% | 99,88666% | 99,93333% |
| -2 | 99,99333% | 99,99333% | 99,99333% | 99,99333% |
| -1 | 100,0000% | 100,0000% | 100,0000% | 100,0000% |
| 0 | 100,0000% | 100,0000% | 100,0000% | 100,0000% |
| 5 | 100,0000% | 100,0000% | 100,0000% | 100,0000% |
| 10 | 100,0000% | 100,0000% | 100,0000% | 100,0000% |
| 15 | 100,0000% | 100,0000% | 100,0000% | 100,0000% |

Capítulo V: Conclusiones

Realizando una comparativa entre las redes neuronales convolucionales unidimensionales y bidimensionales se puede concluir que en los parámetros de exactitud, sensibilidad, especificidad y precisión reflejan mejores resultados en las redes convolucionales unidimensionales, en la tabla 2 se evidencia que desde el SNR de -2 dBs hasta 15 dBs las redes unidimensionales obtuvieron un 100% de exactitud y en el SNR más bajo de -5 dBs un 99,94% de exactitud junto con una baja probabilidad de falsa alarma de un 0,0756%, esto en contraposición a las redes bidimensionales las cuales en el SRN más bajo reflejaron un 99,25% y una probabilidad de falsa alarma de un 1.39%. Entre las causas por las cuales las redes convolucionales unidimensionales obtuvieron mayores porcentajes en los parámetros de rendimiento se encuentra la cantidad de datos con los cuales se trabajaron en ambos modelos: para las redes unidimensionales se usaron 2 millones de señales de radar y para las redes bidimensionales 100 mil señales, esto debido a la capacidad computacional y almacenamiento que conlleva generar y entrenar el modelo bidimensional mediante el uso de imágenes.

Una de las ventajas del uso de redes neuronales convolucionales unidimensionales en comparación a las bidimensionales es que el modelo se lo puede entrenar y validar con el uso de vectores, en contraste con el modelo bidimensional que obligatoriamente requiere el uso de imágenes; esto se puede solventar mediante el empleo de ordenadores con mayores capacidades computacionales de procesamiento y almacenamiento. Con las características necesarias en un ordenador, en las redes neuronales bidimensionales se pueden aprovechar las redes pre entrenadas junto con herramientas de transfer learning que facilitan el diseño e implementación de los modelos por su arquitectura pre diseñada.

Comparando los tiempos de entrenamiento y validación de ambos modelos se puede evidenciar que la red convolucional unidimensional empleo un mayor tiempo debido a que se utilizaron una mayor cantidad de datos para este tipo de red; ambas redes se desarrollaron con

el uso de una unidad de procesamiento central, pero se recomienda hacer uso de una unidad de procesamiento gráfico ya que el tiempo de entrenamiento y validación de los modelos se reduciría notablemente.

Tanto las redes neuronales convolucionales unidimensionales y bidimensionales aventajan a los modelos tradicionales de detección de objetivos en señales de radar sobre todo en los niveles de SNR más bajos donde se evidencia la gran complejidad de la detección del objetivo. El aprendizaje profundo representa una beneficiosa herramienta mediante el uso de las redes neuronales artificiales, permite desarrollar algoritmos capaces de solventar varias problemáticas actuales y futuras del ser humano.

Entre los trabajos futuros se espera continuar con una constante preparación en el campo del aprendizaje profundo, indagando de igual manera en el aprendizaje automático con la finalidad de poder hacer uso de estas técnicas y orientarlas a solucionar varias problemáticas en el campo de las telecomunicaciones y en la detección de objetos en señales de radar, desarrollando diferentes tipos de redes y modelos para analizar la respuesta de todos y poder compararlos.

Bibliografía

Alba Centeno, F. (2019). *Deep Lear Ning*. Universidad de Sevilla.

Almansa Laredo, Á. G. (2023). *Sistema de Identificación de Emociones Independiente del Hablante Basado en Redes Neuronales Convolucionales*. Universidad Politécnica de Madrid.

Alonso Cerpa, J. L. (2015). Introducción a los sistemas de radar y ARPA. *Universidad de la Laguna*.

Andrade Tepán, E. C. (2013). *Estudio de los principales tipos de redes neuronales y las herramientas para su aplicación*. Universidad Politécnica Salesiana Sede Cuenca.

Antona Castañares, A. (2020). *Aplicación del Dropout a la Cuantificación de la Incertidumbre en Redes Neuronales*. Universidad Politécnica de Madrid.

Artola Moreno, Á. (2019). *Clasificación de imágenes usando redes neuronales convolucionales en Python*. Universidad de Sevilla.

Basogain Olabe, X. (2017). *Redes Neuronales Artificiales y sus Aplicaciones*. Escuela Superior de Ingeniería de Bilbao.

Bonilla Carrión, C. (2020). *Redes Convolucionales*. Universidad de Sevilla.

Cabrera Mora, J. J. (2021). *Entrenamiento, optimización y validación de una CNN para la localización de un robot móvil mediante tareas de clasificación y regresión*. Universidad Miguel Hernández de Elche.

Campos Mocholí, M. (2020). *Clasificación de textos basada en redes neuronales*. Universidad Politécnica de Valencia.

- Casado Beinat, N. (2022). *Redes Neuronales Convolucionales y Aplicaciones*. Universidad Computense de Madrid.
- Donalba. (2015, junio 19). *Han cambiado los sistemas radar en los últimos años*. <https://www.donalba.com/es/blog-novedades/han-cambiado-los-sistemas-radar-en-los-ultimos-anos>.
- Durán Suárez, J. (2017). *Redes Neuronales Convolucionales en R - Reconocimiento de caracteres escritos a mano*. Universidad de Sevilla.
- Erroz Arroyo, D. (2019). *Visualizando neuronas en Redes Neuronales Convolucionales*. Universidad Politécnica de Navarra.
- García Roque, A. A. (2022). *Optimización de redes neuronales convolucionales para la clasificación de imágenes hiperespectrales*. Universidad Nacional de Educación a Distancia (UNED).
- García Sánchez, E. (2019). *Introducción a las redes neuronales de convolución. Aplicación a la visión por ordenador*. Universidad de Zaragoza.
- Gimenez Arnal, M. (2018). *Estudio y aplicación de las redes neuronales convolucionales 3D*. Universidad Politécnica de Catalunya.
- González Molina, F. (s. f.). *Aplicación de redes neuronales en el cálculo de sobretensiones y tasa de contorneamientos* [Universidad Complutense de Madrid]. Recuperado 20 de septiembre de 2023, de https://www.tdx.cat/bitstream/handle/10803/6281/Capitulo_6a.PDF?sequence=9
- González Montoro, A. (2021). *Métodos Clásicos de Clasificación: Comparación y Aplicación*. Universidad Nacional de Córdoba.

- Hernández González, E. M. (2020). *Clasificación de señales EEG basada en representaciones bidimensionales y redes neuronales convolucionales*. Instituto Nacional de Astrofísica, Óptica y Electrónica.
- IAAR. (2017). *Inteligencia Artificial*. Comunidad de Inteligencia Artificial Argentina. <https://iaarbook.github.io/deeplearning/>
- INREDH. (2021, agosto 5). *Bombardeo en Angostura*. Fundación Regional de Asesoría en Derechos Humanos. <https://inredh.org/bombardeo-en-angostura/>
- Isasi, V. P., & Galván, L. I. (2004). *Redes Neuronales Artificiales Un Enfoque Práctico* (Pearson Educación). Prentice Hall.
- Lara Mina, M. F., & Ortiz Moya, M. A. (2019). *Procesador bayesiano de señales de radar para el sistema oerlikon*. Universidad de las Fuerzas Armadas ESPE.
- López Espejo, I. (2011). Introducción al RADAR. *Universidad de Granada*.
- Lopez Pacheco, M. A. (2021). *Redes neuronales convolucionales para el modelado de sistemas no lineales con aplicación al monitoreo de daño estructural*. Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional.
- MathWorks. (s. f.). *Introducción a la convolución*. MathWorks. Recuperado 19 de septiembre de 2023, de <https://la.mathworks.com/discovery/convolution.html>
- MathWorks - MATLAB. (s. f.). *Alexnet*. MathWorks. Recuperado 21 de septiembre de 2023, de <https://la.mathworks.com/help/deeplearning/ref/alexnet.html>
- MATLAB - MathWorks. (s. f.). *Transfer learning para entrenar modelos de Deep Learning*. MathWorks. Recuperado 21 de septiembre de 2023, de <https://la.mathworks.com/discovery/transfer-learning.html>

- Mejía, E. (2016). *Función de Transferencia y la Respuesta - Impulso*. Universidad Nacional Autónoma de Honduras.
- Muñiz González, A. (2018). *Aplicaciones de técnicas de inteligencia artificial basadas en aprendizaje profundo (deep learning) al análisis y mejora de la eficiencia de procesos industriales*. Universidad de Oviedo.
- Parker Sanfuentes, J. (2016). Historia Del Radar. *Revista de Marina*.
- Pascual, J. P. 2014. (2014). *Procesamiento de señales de radar en presencia de clutter dinámico* [Universidad Nacional de la Plata]. <http://sedici.unlp.edu.ar/handle/10915/44032>
- Percat Fedalto, F. A. (2015). Producto de Convolución: Aplicaciones. . *Universidad Nacional del Sur, Argentina*. <http://lcr.uns.edu.ar/fvc/NotasDeAplicacion/FVC-PercatFernando.pdf>
- Pérez Aguilar, D., Risco Ramos, R., & Casaverde Pacherez, L. (2021). Transfer learning en la clasificación binaria de imágenes térmicas. *Ingenius - Revista de Ciencia y Tecnología*. <https://www.redalyc.org/journal/5055/505567902007/505567902007.pdf>
- Pérez Patiño, C. (2007). *Sistemas de Telecomunicaciones* (Vol. 1). Universidad de Cantabria.
- Presidencia Ecuador. (2017, marzo 30). *Sistema Integrado de Radares del Ecuador*. <https://www.presidencia.gob.ec/wp-content/uploads/downloads/2017/04/2017.03.30-PRESENTACION-DEL-SISTEMA-DE-RADARES.pdf>
- Primicias. (2021, octubre). “Pequeños radares tácticos” cubren vigilancia aérea en Manabí. *Primicias*. <https://www.primicias.ec/noticias/en-exclusiva/pequenos-radares-tacticos-para-vigilancia-antinarcoicos-manabi/>
- Ruiz, C. A., & Basualdo, M. S. (2001). *Redes Neuronales: Conceptos Básicos y Aplicaciones*. Universidad Tecnológica Nacional Facultad Regional Rosario.

- Sanchez, J., & Exposito, A. (2020). *Evaluación de algoritmos de detección de objetos basados en deep learning para detección de incidencias en carreteras*. Universidad de Valladolid.
- Serrano Moreno, C. (2018). *Sistema para la detección precoz de cáncer de piel basado en tratamiento de imágenes*. Universidad Autónoma de Madrid.
- Telefónica Tech. (2021, diciembre 13). *Cómo interpretar la matriz de confusión: ejemplo práctico*. Telefónica Tech. <https://telefonicatech.com/blog/como-interpretar-la-matriz-de-confusion-ejemplo-practico>
- Vaca Santacruz, F. (2021, noviembre 8). *La accidentada historia de los radares en Ecuador*. Plan V. <https://www.planv.com.ec/historias/politica/la-accidentada-historia-radares-ecuador>
- Vilagran Solsona, A. (2018). *Facial Expression Detection using Convolutional Neural Networks*. Universidad Politécnica de Catalunya.
- Zhang, A., Zachary, L., Li, M., & Smola, A. (2023). *Dive into Deep Learning (Vol. 1)*. <https://d2l.ai/>