

Clasificación de objetos empleando imágenes espectrales y aprendizaje de máquina

Carlos Eduardo López Joya

Trabajo de Grado para optar al título de Ingeniero Electrónico

Director

Hans Yecid Garcia Arenas

Ms. Ingeniería Electrónica

Co-Director

Henry Arguello Fuentes

Ph.D. Ingeniería Electrónica y Computación

Universidad Industrial de Santander

Facultad de Ingenierías Fisicomecánicas

Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones

Bucaramanga

2020

Dedicatoria

*A mi madre **Margot Joya** por ser mi polo a tierra, porque no hubiera podido llegar tan lejos sin tu apoyo, sin tu amor y sin tu confianza en mí, porque cuando estuve a punto de rendirme creíste en mí y me convenciste de que podía lograr todo lo que me propusiera. No tengo suficientes palabras para agradecerte todo lo que has hecho por mí.*

*A mi padre **Carlos López** por ayudarme incondicionalmente a pesar de las dificultades que hemos vivido.*

*A mi tío **Cesar Valderrama** por tus consejos que me ayudan a tomar las decisiones correctas, porque me enseñaste a superar las dificultades de la vida y a ser cada día más fuerte.*

*A mis otros **familiares** por su apoyo, contribuyendo incondicionalmente a lograr las metas y objetivos propuestos.*

A todos mis amigos y compañeros por las vivencias de estos inolvidables años de universidad.

Carlos Eduardo López Joya

Agradecimientos

A través de estas líneas expreso mi profundo agradecimiento a todos mis profesores y miembros del Comité Trabajos de Grado de Pregrado de la Escuela de Ingeniería Eléctrica, Electrónica y de Telecomunicaciones por el tiempo que invirtieron en mí para formarme académicamente.

Un agradecimiento importante a mi director de trabajo de grado, por dedicar su tiempo, experiencia y conocimiento en la guía de mi proyecto.

A la Universidad Industrial de Santander por permitirme ser parte de esta alma mater.

Tabla de Contenido

Introducción	13
1 Objetivos	15
2 Marco conceptual	16
2.1 Imágenes espectrales	16
2.2 Segmentación de imágenes	17
2.3 Aprendizaje de Máquina	18
2.4 Clasificación de objetos a partir de imágenes espectrales	21
2.5 Aprendizaje de máquina para la clasificación de imágenes espectrales	22
3 Estrategia para la clasificación de objetos	24
3.1 Diseño e implementación de la arquitectura óptica para la adquisición de una base de datos con imágenes espectrales	25
3.2 Bases de datos capturadas	29
3.3 Algoritmo de segmentación de imágenes espectrales	31
3.4 Modelos de aprendizaje de máquina	33
3.4.1 Modelo de aprendizaje de máquina unidimensional entrenado a partir de firmas espectrales de los datos segmentados	34

3.4.2	Modelo de aprendizaje de máquina empleando información espacial de la escena a partir de los datos segmentados	40
4	Simulación y Resultados	53
4.1	Rendimiento del modelo unidimensional entrenado a partir de firmas espectrales de los datos segmentados	53
4.2	Rendimiento del modelo bidimensional entrenado a partir de los datos segmentados	55
5	Conclusiones	71
	Referencias Bibliográficas	72

Lista de Figuras

Figura 1	Métodos para adquirir imágenes espectrales. (A) Adquisición por píxel. (B) Adquisición por filas. (C) Filtrado espectral. (D) Snapshot.	17
Figura 2	Ejemplo de segmentación para separar los objetos del fondo.	18
Figura 3	Modelo básico de clasificación empleando aprendizaje de máquina.	21
Figura 4	Esquema básico de una red neuronal para la clasificación de imágenes espectrales.	23
Figura 5	Arquitectura óptica para la adquisición de imágenes espectrales.	26
Figura 6	Monocromador Cornerstone 130 1/8m.	27
Figura 7	Firmas espectrales promedio de cada medicamento.	31
Figura 8	Medicamento en presentación de tableta sin preprocesamiento.	32
Figura 9	Proceso de segmentación de la base de datos para la separación del fondo y de los medicamentos.	33
Figura 10	Modelo de neuronal convolucional 1D SpectralNet1.	38
Figura 11	Modelo de red neuronal convolucional 2D CarlosNet1.	44
Figura 12	Ejemplos de salida de las capas de la red convolucional CarlosNet1.	45
Figura 13	Modelo de red neuronal convolucional 2D CarlosNet2.	50
Figura 14	Ejemplos de salida de las capas de la red convolucional CarlosNet2.	51
Figura 15	Matriz de confusión de entrenamiento para la red SpectralNet1.	54

Figura 16	Matriz de confusión de validación para la red SpectralNet1.	54
Figura 17	Matriz de confusión de entrenamiento para la red CarlosNet1.	56
Figura 18	Matriz de confusión de validación para la red CarlosNet1.	57
Figura 19	Matriz de confusión de prueba para la red CarlosNet1.	57
Figura 20	Matriz de confusión de entrenamiento para la red CarlosNet1.	58
Figura 21	Matriz de confusión de validación para la red CarlosNet1.	59
Figura 22	Matriz de confusión de prueba para la red CarlosNet1.	59
Figura 23	Matriz de confusión de entrenamiento para la red CarlosNet2.	60
Figura 24	Matriz de confusión de validación para la red CarlosNet2.	61
Figura 25	Matriz de confusión de prueba para la red CarlosNet2.	61
Figura 26	Matriz de confusión de entrenamiento para la red CarlosNet2.	62
Figura 27	Matriz de confusión de validación para la red CarlosNet2.	63
Figura 28	Matriz de confusión de prueba para la red CarlosNet2.	63
Figura 29	Aprendizaje de transferencia.	65
Figura 30	Matriz de confusión de entrenamiento para la red AlexNet.	66
Figura 31	Matriz de confusión de validación para la red AlexNet.	66
Figura 32	Matriz de confusión de prueba para la red AlexNet.	67
Figura 33	Matriz de confusión de entrenamiento para la red ResNet18.	68
Figura 34	Matriz de confusión de validación para la red ResNet18.	68
Figura 35	Matriz de confusión de prueba para la red ResNet18.	69

Lista de Tablas

Tabla 1	Especificaciones del monocromador Cornerstone 130 1/8m.	28
Tabla 2	Especificaciones de rendimiento de la cámara monocromática Stringray F-145.	29
Tabla 3	Cantidad de medicamentos adquiridos en presentación de tableta.	30
Tabla 4	Composición de capas y parámetros entrenables de la red neuronal convolucional 1D SpectralNet1.	39
Tabla 5	Comparación de las arquitecturas de las redes neuronales convolucionales 2D.	52
Tabla 6	Conjunto de firmas espectrales separadas en entrenamiento y validación.	53
Tabla 7	Conjunto de imágenes separadas en entrenamiento, validación y prueba.	55
Tabla 8	Comparación de los resultados de precisión empleando los datos de validación de las redes CarlosNet1 y CarlosNet2.	64
Tabla 9	Resultados de precisión de validación.	69

Resumen

Título: Clasificación de objetos empleando imágenes espectrales y aprendizaje de máquina. *

Autor: Carlos Eduardo López Joya **

Palabras Clave: Segmentación, imágenes espectrales, aprendizaje de máquina, aprendizaje de transferencia, clasificación.

Descripción: Una de las grandes aplicaciones de las imágenes espectrales es la clasificación de materiales, dentro de los que podemos encontrar clasificación de alimentos, cultivos y patologías en humanos. Tradicionalmente las imágenes espectrales se pueden clasificar empleando información del ángulo del vector de información espectral de cada píxel o firma espectral, específicamente empleando la métrica del mapeador de ángulo espectral (SAM) que se utiliza para comparar una firma espectral del material a clasificar conocido y el de un píxel espacial no clasificado en la imagen. Por otro lado, recientemente se han desarrollado una serie de algoritmos para la clasificación, entre ellos se encuentran los basados en redes neuronales convolucionales (CNN), las cuales han mostrado un rendimiento sobresaliente para la clasificación de imágenes espectrales de manera automática. Por lo tanto, en este proyecto se presenta un clasificador de objetos empleando imágenes espectrales adquiridas en el laboratorio de óptica del grupo HDSP, por medio de un algoritmo basado en aprendizaje de máquina con CNN, además, se propone realizar un proceso de segmentación de imágenes para extraer el área de interés dentro de la escena. Dentro de los diversos algoritmos que se presentan en este proyecto se pueden agrupar en dos bloques uno que emplea directamente la firma espectral (SpectralNet1) y otro que incluye también información espacial (CarlosNet1 y CarlosNet2). Dentro de los modelos propuestos de aprendizaje de máquina el modelo que obtiene los mejores resultados para llevar a cabo la tarea de clasificación es CarlosNet1 obteniendo un 97.3% de precisión en el conjunto de datos de validación.

* Trabajo de grado.

** Facultad de Ingeniería Fisicomecánicas. Escuela de Ingenierías Eléctrica, Electrónica y Telecomunicaciones. Director: Hans Yecid Garcia Arenas, Ms. Ingeniería Electrónica, Co-Director: Henry Arguello Fuentes, Ph.D. Ingeniería Electrónica y Computación.

Abstract

Title: Object classification using spectral imaging and machine learning. *

Author: Carlos Eduardo López Joya. **

Keywords: Segmentation, spectral image, machine learning, transfer learning, classification.

Description: One of the great applications of spectral images is the classification of materials, within which we can find classification of food, crops, and human pathologies. Traditionally, spectral images can be classified by using the angle information of the spectral information vector of each pixel or spectral signature, specifically using the metric of the Spectral Angle Mapper (SAM) that is used to compare a spectral signature of the known material to be classified and that of an unclassified spatial pixel in the image. On the other hand, a few of classification algorithms have recently been developed, among them those based on convolutional neural networks (CNN), which have shown outstanding performance for the automatic classification of spectral images. Therefore, this project presents an object classifier using spectral images acquired in the HDSP group's optics laboratory, by means of an algorithm based on automatic learning with CNN. In addition, it is proposed to perform an image segmentation process to extract the area of interest within the scene. Within the different algorithms presented in this project, they can be grouped in two blocks, one that directly uses the spectral signature (SpectralNet1) and another that also includes spatial information (CarlosNet1 and CarlosNet2). Within the proposed machine learning models, the model that obtains the best results for carrying out the classification task is CarlosNet1 obtaining 97.3% accuracy in the validation data set.

* Bachelor Thesis.

** Faculty of Engineering and Physical Sciences. School of Electronic and Electrical Engineering. Director: Hans Yecid Garcia Arenas, Ms in Electronic Engineering. Co-Director: Henry Arguello Fuentes, Ph.D. Electronic and Computer Engineering.

Introducción

Las imágenes espectrales son clasificadas tradicionalmente por medio del ángulo del vector de información espectral de cada píxel también conocido como firma espectral, específicamente, se utiliza la métrica de mapeador de ángulos espectrales (SAM) para comparar una firma espectral de un material previamente conocido y la de un píxel espacial sin clasificar en la imagen (Girouard y Bannari, 2004), esta métrica calcula el ángulo entre dos vectores en \mathbb{R}^λ para comparar su semejanza en un espacio con dimensionalidad igual al número de bandas espectrales λ , permitiendo así agrupar los píxeles con firma espectral similar a la conocida de cada material. Es importante destacar que este tipo de clasificación se hace de manera supervisada, es decir, requiere conocer previamente la firma espectral del material a clasificar y estimar los umbrales de clasificación específicos para cada clase, siendo así esta métrica no es el mejor camino para clasificar imágenes espectrales ya que puede existir firmas espectrales idénticas en diferentes objetos, ocasionando error en la clasificación de objetos.

Para clasificar imágenes espectrales el mejor método es aplicar aprendizaje de máquina ya que imitan el comportamiento del cerebro humano a partir de modelos matemáticos y probabilísticos (Haykin, 1994). Permitiendo realizar la tarea de clasificación sin supervisión humana, posibilitando almacenar una gran cantidad de datos, produciendo modelos de manera rápida y automática que puedan analizar una cantidad de datos grandes y complejos con la finalidad de producir resultados rápidos y precisos de manera automática (Hinton *et al.*, 2006).

En las últimas décadas con los avances tecnológicos en el área de inteligencia artificial

se ha venido trabajando con aprendizaje de máquina siendo el estudio de algoritmos y modelos estadísticos que permiten realizar una tarea específica sin utilizar instrucciones específicas.

En los últimos años se han venido empleando técnicas de aprendizaje de máquina para múltiples aplicaciones del mundo real como: Clasificación de objetos (Nirkin *et al.*, 2018), visión por computadora (Zheng y Xiao, 2016), clasificación de imágenes espectrales (Samat *et al.*, 2014), segmentación de imágenes a color de manera semi supervisado. Teniendo todo lo anterior en cuenta, en este proyecto se propone emplear técnicas de aprendizaje de máquina para realizar la tarea de clasificación de objetos empleando imágenes espectrales.

1. Objetivos

Objetivo general

Diseñar y entrenar un modelo de aprendizaje de máquina que permita clasificar objetos a partir de imágenes espectrales.

Objetivos específicos

Construir una base de datos de imágenes espectrales de los medicamentos en presentación de tableta seleccionados para ser clasificados.

Diseñar un modelo de aprendizaje de máquina que realice la tarea de clasificación de los medicamentos en presentación de tableta seleccionados.

Evaluar el rendimiento de la metodología de clasificación propuesta.

2. Marco conceptual

2.1. Imágenes espectrales

En las últimas décadas se ha venido trabajando con imágenes espectrales ya que tienen implicaciones en diversas aplicaciones tales como: Inspección de la calidad de alimentos (García-Esteban *et al.*, 2018), detección de impurezas en metales (Peichl *et al.*, 2015), identificación de tumores (Shahriar Sazzad *et al.*, 2019) y otras patologías en seres humanos (Miranda y Salinas, 2019). Las imágenes espectrales están compuestas por dos dimensiones espaciales (x, y) y una dimensión espectral en el dominio de la longitud de onda λ .

Las imágenes espectrales contienen múltiples bandas en el espectro electromagnético, cada una de ellas contienen un rango estrecho de longitudes de onda del espectro, las imágenes espectrales se pueden utilizar en el espectro (ultravioleta, visible e infrarrojo) (Bi *et al.*, 2011), o generando serie de combinaciones de estas, las cuales se pueden adquirir imágenes espectrales en el espectro visible y no visible o la combinación de estas.

Las imágenes espectrales se pueden modelar como un cubo de datos $\mathbf{F}(x, y, \lambda) \in \mathbb{R}^3$, donde (x, y) representan las coordenadas espaciales, y λ representa la variación espectral. Para adquirir imágenes espectrales se han venido empleando las siguientes técnicas: Adquisición por píxel a partir de un escáner (Wish-broom) se hace un barrido espacial a punto a punto el cual contiene toda la información espectral de la escena, adquisición por filas a partir de un escáner (Push-broom) permitiendo adquirir un corte espacial con información espectral de la escena, espectrómetro por filtrado espectral captura cada banda como una imagen en escala de grises con la información

espacial completa, snapshot imaging la cual contiene todos los datos espaciales (x, y) y espectrales (Bi *et al.*, 2011). En la Figura 1 se representa los diferentes métodos para adquirir imágenes espectrales.

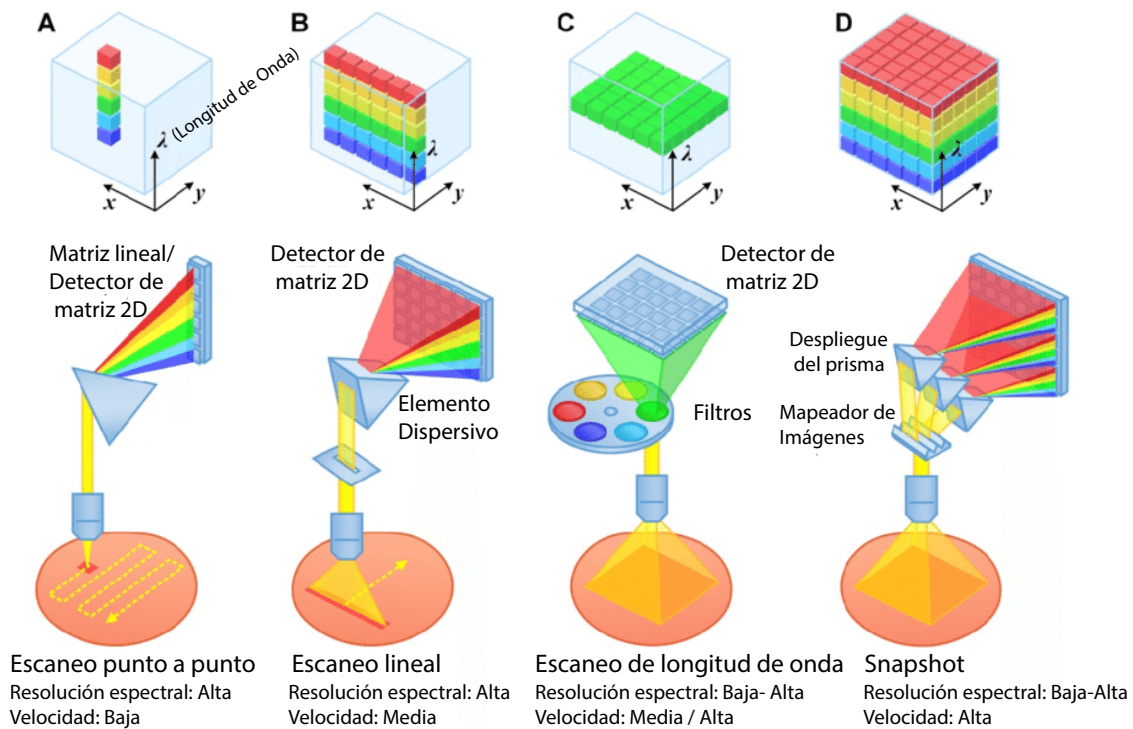


Figura 1. Métodos para adquirir imágenes espectrales. (A) Adquisición por píxel. (B) Adquisición por filas. (C) Filtrado espectral. (D) Snapshot.

2.2. Segmentación de imágenes

La segmentación de imágenes es el proceso de dividir una imagen digital en varios subconjuntos de píxeles para identificar diferentes regiones de la imagen tales como superficies, objetos, o porciones de objetos (Phothisonothai *et al.*, 2016). La segmentación de imágenes es utilizada en aplicaciones como la localización de tumores cerebrales (Singh *et al.*, 2018), reconocimiento de rostros (Nirkin *et al.*, 2018), reconocimiento de objetos (Sharma *et al.*, 2016), visión por computadora (Zheng y

Xiao, 2016), entre otros.

En las últimas décadas la segmentación de imágenes ha sido un tema de gran importancia puesto que es una técnica para delimitar y separar con precisión una región de interés, lo cual facilita la identificación de la información necesaria en las imágenes procesadas.

Las técnicas más empleadas para la segmentación de imágenes son: Formación de regiones a partir de píxeles con firma espectral similar (Ben Fadhel *et al.*, 2016a) y detección de bordes la cual se apoya en la detección de cambios locales bruscos en la intensidad de la imagen (Ben Fadhel *et al.*, 2016b). A partir de estas técnicas la segmentación de imágenes permite separar objetos de interés de una imagen de información no deseada, como el fondo u otros elementos y así dividir la imagen en subconjuntos y regiones, en la Figura 2 se observa un ejemplo de separar los objetos del fondo.

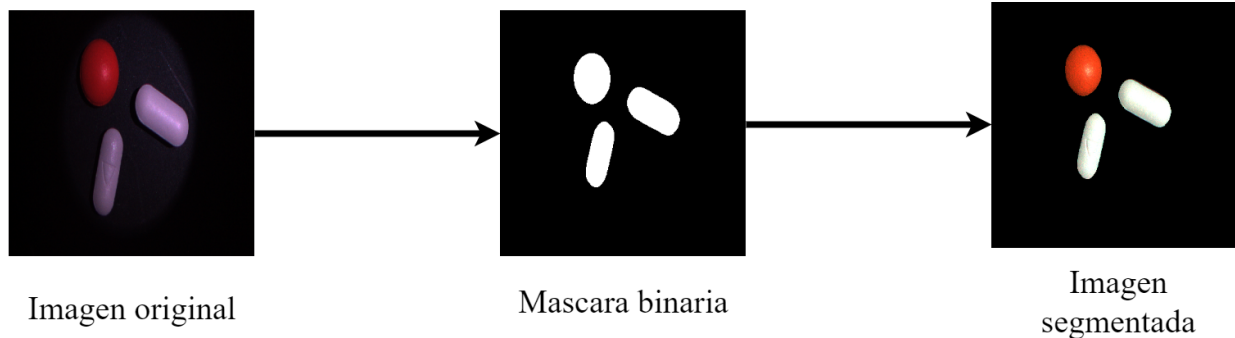


Figura 2. Ejemplo de segmentación para separar los objetos del fondo.

2.3. Aprendizaje de Máquina

El aprendizaje de máquina es el estudio de algoritmos y modelos estadísticos que permiten realizar una tarea específica, empleando modelos matemáticos, donde estos modelos son capaces de optimizar un criterio de desempeño usando datos de ejemplos o la recolección de experiencias pasadas

(Zhang *et al.*, 2017a), permitiendo hacer predicciones o decisiones de manera automática (Seixas *et al.*, 2015). Dentro de estas tareas específicas podemos encontrar: Reconocimiento de manuscritos (Makhfi *et al.*, 2011), reconocimiento de rostros (Mondal y Chatterjee, 2019), categorización de textos (Moustakis y Stankovic, 2005), entre otros.

Dentro de estas aplicaciones se han venido desarrollando varios tipos de aprendizaje de máquina, entre ellos se encuentran: Aprendizaje supervisado el cual trabaja con datos etiquetados en el conjunto de datos de entrada, donde su problema fundamental es la de encontrar una función que dada un conjunto de entradas le asigne la etiqueta adecuada a la salida (Hussein *et al.*, 2019). Otro tipo es el aprendizaje no supervisado, el cual tiene lugar cuando no se dispone datos etiquetados, tratando a los datos de entrada como un conjunto de variables aleatorias, por tal razón no existen datos de salida que correspondan a un determinado conjunto de datos de entrada, este tipo de aprendizaje se basa en la búsqueda de patrones de similitud en los datos de entrada (Radford *et al.*, 2015), entre otros.

Entre los algoritmos más usados de aprendizaje de máquina encontramos: Regresión (Zhang *et al.*, 2017b), bayesianos (Mao *et al.*, 2018), agrupación (Fesehayee *et al.*, 2017), árbol de decisión (Azad y Moshkov, 2014), soporte vectorial (Le *et al.*, 2019), redes neuronales convolucionales (Idrissi *et al.*, 2016), entre otros; (Banerjee *et al.*, 2016) (Gupta *et al.*, 2016) (Jr *et al.*, 2019). En las últimas décadas se han desarrollado una serie de algoritmos para la clasificación, entre ellos se encuentran los basados en redes neuronales convolucionales (CNN) (Mazumdar *et al.*, 2018), las cuales han mostrado un rendimiento sobresaliente para la clasificación de imágenes de manera automática (Garg *et al.*, 2019).

Las redes neuronales convoluciones es un tipo de redes neuronales artificiales con aprendizaje supervisado, las cuales están bio-inspirado en el mecanismo cognitivo, visual y biológico, permitiendo identificar distintas características para imitar la visión de los seres vivos. Las redes neuronales convolucionales permiten construir modelos de clasificación basados en datos de ejemplo permitiendo hacer predicciones de clasificación o decisiones de manera automática. Las redes neuronales se compone de una capa de entrada, capas convolucionales, capa de activación, capas de agrupación, capas totalmente conectadas y capa de salida (Mazumdar *et al.*, 2018).

Los datos utilizados para construir estos modelos generalmente provienen de múltiples conjuntos de datos. En general se usan tres conjuntos de datos en diferentes etapas para la construcción del modelo de clasificación. El modelo de clasificación se ajusta inicialmente con un conjunto de datos de entrenamiento siendo un subconjunto de datos para entrenar el modelo permitiendo ajustar los parámetros del modelo tales como los pesos de las conexiones entre neuronas. El modelo de clasificación ajustado se usa para probar el rendimiento del modelo entrenado realizando predicciones de clasificación a partir de un segundo conjunto de datos conocido como validación siendo un subconjunto de datos proporcionando una evaluación del rendimiento del modelo entrenado. Finalmente, el conjunto de datos de prueba es un conjunto de datos utilizado solo para evaluar el rendimiento del modelo diseñado, este conjunto de datos de prueba nunca se han utilizado en el entrenamiento del modelo de clasificación. En la Figura 3 se presenta el modelo básico de clasificación para los algoritmos de aprendizaje de máquina.

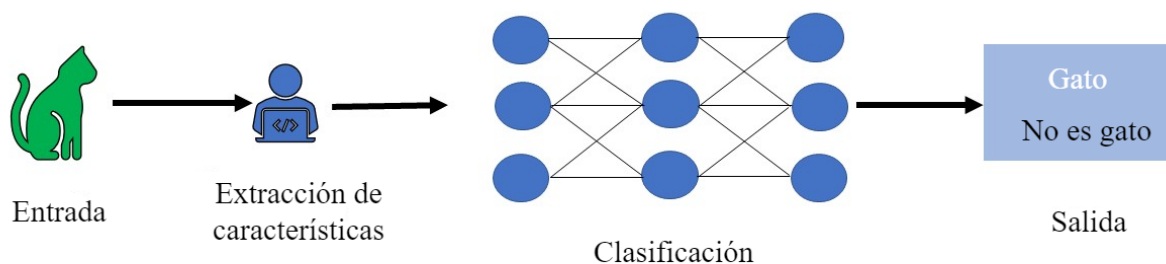


Figura 3. Modelo básico de clasificación empleando aprendizaje de máquina.

2.4. Clasificación de objetos a partir de imágenes espectrales

La clasificación de objetos se ha convertido recientemente en uno de los campos más interesantes en visión artificial e inteligencia artificial con estos se busca automatizar procesos industriales que requieren realizar la tarea de clasificación de objetos. La clasificación de objeto tiene implicaciones en diversas aplicaciones como: Análisis de deterioro de estructuras (Li y Gu, 2010), clasificación automática de organismos patógenos (Alvarez-Ramos *et al.*, 2013), detección de objetos por satélites (Veremyev *et al.*, 2019), supervisión de cultivos agrícolas (An *et al.*, 2010), identificación de vegetación (Hasan *et al.*, 2013), inspección de calidad de alimentos (Hasan *et al.*, 2013), entre otros.

En las últimas décadas se han desarrollaron diferentes métodos de clasificación empleando imágenes RGB tales como: Técnicas de extracción de características a partir de datos RGB (Ali, 2017), clasificación de objetos a partir de histogramas (Wang y Liew, 2006), clasifican de imágenes de textura (Kulkarni y Stranieri, 2018). Sin embargo al emplear imágenes RGB; se tienen grandes

limitaciones debido a que en aplicaciones específicas este tipo de datos no son suficientes para diferenciar los objetos; y no se logra obtener un buen desempeño en la etapa de clasificación como es el caso de clasificación del nivel de madurez de las manzanas (Mulyani y Susanto, 2017) o la detección temprana de cáncer (Zeng *et al.*, 2014). Para solucionar este problema de las imágenes RGB se han venido empleando imágenes espectrales con más de 3 bandas del espectro, ya que la clasificación usando imágenes espectrales puede lograr un mejor rendimiento de clasificación que su contraparte de clasificación usando imágenes RGB de imágenes debido a que se puede utilizar más bandas del espectro electromagnético, por lo cual se tiene más información de para realizar la tarea de clasificación.

Los métodos tradicionales de clasificación de imágenes espectrales basados en características espectrales no tienen en cuenta las características espaciales para solucionar este problema (Zeng *et al.*, 2015) los académicos e investigadores proponen fusionar características espacial y espectrales (Feng *et al.*, 2019) de este modo se optimiza el clasificador de imágenes.

2.5. Aprendizaje de máquina para la clasificación de imágenes espectrales

La clasificación de imágenes espectrales tiene implicaciones en diversas aplicaciones como: Segmentación automática de imágenes cerebrales espectrales de resonancia magnética (Chegini y Ghassemian, 2011), clasificación de imágenes espectrales para la detección de cáncer de piel (Dhinagar y Celenk, 2014), segmentación y clasificación de imágenes de cromosomas (Karvelis *et al.*, 2008), entre otros; (Signoroni *et al.*, 2018) (Lu y Fei, 2014) (Ma *et al.*, 2017).

En las últimas décadas el estado del arte ha propuesto novedosos métodos basados en redes neuronales convoluciones para clasificar imágenes espectrales. Entre ellos encontramos: Modelos

de redes neuronales convolucionales basados en superpíxeles para la clasificación de imágenes espectrales permitiendo reducir la cantidad de datos de entrada para la CNN (Zhao *et al.*, 2017), también han propuesto segmentación semántica a partir de CNN permitiendo etiquetar cada firma espectral en una imagen con una clase determinada empleado en imágenes de alta resolución proporcionando resultados favorables (Aytaylan y Yuksel, 2016).

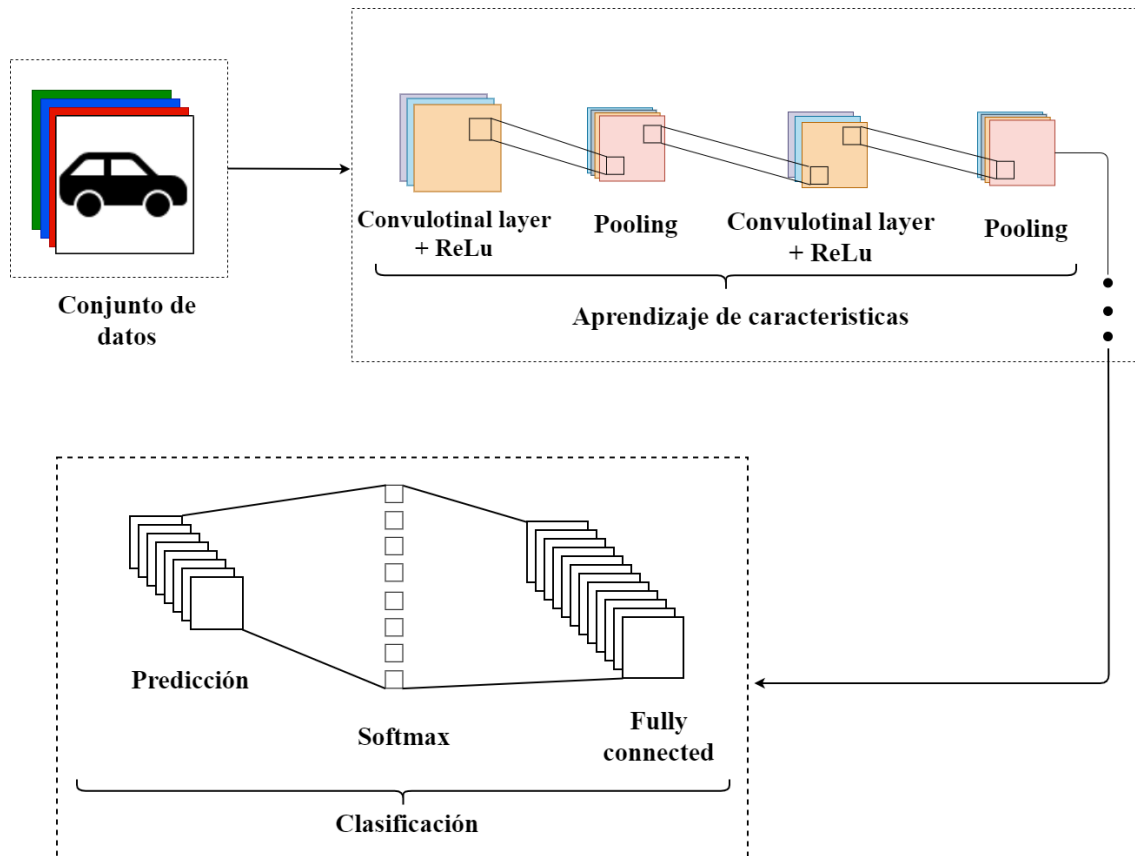


Figura 4. Esquema básico de una red neuronal para la clasificación de imágenes espectrales.

3. Estrategia para la clasificación de objetos

Este proyecto pretende clasificar medicamentos en presentación de tableta a partir de imágenes espectrales tales como: Acetaminofen, dolex, metocarbamol, mioflex, ibuprofeno, calmidol, ácido acetilsalicílico, aspirina, se escogieron estos medicamentos como caso de estudio ya que clasificarlos a partir de imágenes RGB sería una tarea compleja, puesto que sus colores en imágenes RGB serían idénticas para la mayoría de los medicamentos. Por esta razón este proyecto se centra en el diseño y entrenamiento de una arquitectura de aprendizaje de máquina para realizar la tarea de clasificación de medicamentos en presentación de tableta.

Para ello se debe construir una arquitectura óptica a partir del escaneo espectral en el laboratorio del grupo de investigación HDSP de la Universidad Industrial de Santander de Colombia ya que se cuenta con los instrumentos que permite la adquisición de la base de datos de imágenes espectrales de los objetos a clasificar, específicamente, esta arquitectura se basa en la técnica de escaneo espectral. Esta base de datos adquirida de imágenes espectrales se utiliza para el diseño de un algoritmo de segmentación permitiendo separar los objetos a clasificar, este algoritmo de segmentación se fundamenta en la separación del fondo y la sección de la imagen donde se encuentra el objeto de interés, esto con el fin de separar los objetos dentro de cada imagen espectral, lo cual facilita la etapa posterior de entrenamiento de la arquitectura de aprendizaje de máquina y clasificación de objetos.

Específicamente, en el proceso de clasificación de los medicamentos se emplea un modelo de aprendizaje de máquina, cuya implementación requiere de dos etapas, la primera de ellas se

centra en el diseño de la arquitectura (definición de la cantidad y tipo de capas que este posee) el cual inicialmente se realiza de acuerdo al estado del arte para clasificación de objetos, y la segunda etapa se centra en el entrenamiento de sus capas que se realiza a partir de la base de datos de imágenes espectrales construida previamente. Es importante destacar que el modelo de la red empleado para realizar el entrenamiento inicial se encontrará sujeto a modificaciones y adaptaciones requeridas para mejorar los resultados de clasificación.

3.1. Diseño e implementación de la arquitectura óptica para la adquisición de una base de datos con imágenes espectrales

Se construyó una arquitectura óptica en el laboratorio del grupo de investigación HDSP de la Universidad Industrial de Santander de Colombia que permite adquirir imágenes espectrales. Esta arquitectura óptica se encuentra compuesta por un monocromador Cornestone 130 1/8m cuya función es seleccionar una longitud de onda específica de la luz para iluminar la escena, también cuenta con una cámara monocromática Stringray F-145 cuya función es adquirir las imágenes en escala de grises en cada longitud de onda con una resolución espacial de 776X1030 píxeles, el método que se empleó para adquirir las imágenes espectrales es por escaneo espectral. Esto con el objetivo de construir la base datos que se empleará para el entrenamiento del algoritmo basado en aprendizaje de máquina con el fin de desarrollar la tarea de clasificación, para la adquisición de las imágenes espectrales se crea una serie scripts en el software de Matlab permitiendo la comunicación entre el monocromador y Matlab a partir del puerto serial, el cual procesara los datos, esta tarea se automatizo permitiendo reducir el tiempo de captura de la escena. Como resultado se obtiene un cubo de datos con una dimensión de 776x1030x29 conteniendo 29 bandas del espec-

tro, este script de matlab fue necesario debido a que el software del monocromador Cornestone entregado por el fabricante no permite automatizar los cambios en las longitudes de onda y sincronizarlos con una cámara Stringray, por lo cual se tomaría demasiado tiempo en adquirir cada imagen cambiando manualmente la longitud de onda. La arquitectura óptica que se diseño para la construcción de la base de datos se presenta en las Figuras 5 y 6.

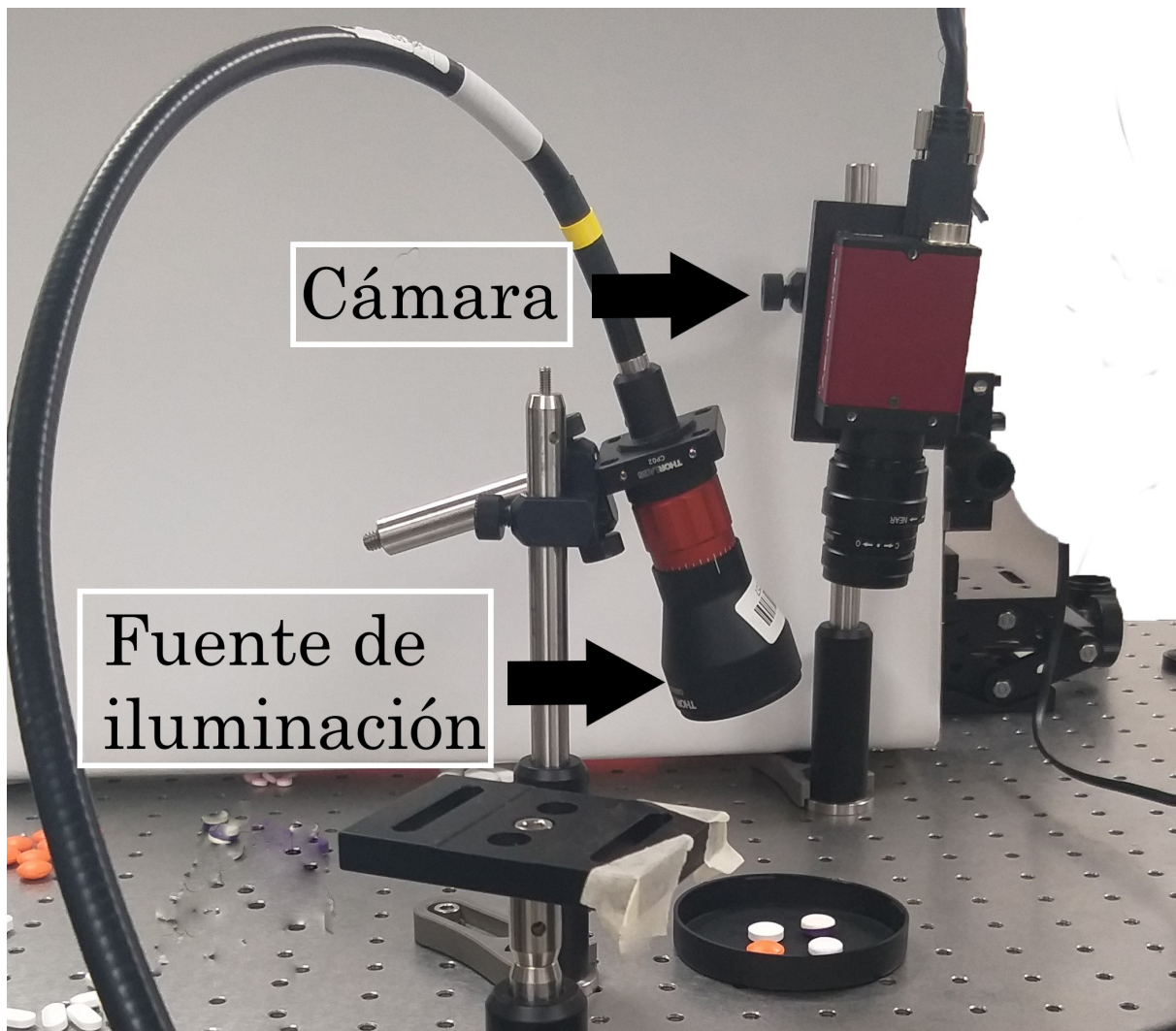


Figura 5. Arquitectura óptica para la adquisición de imágenes espectrales.

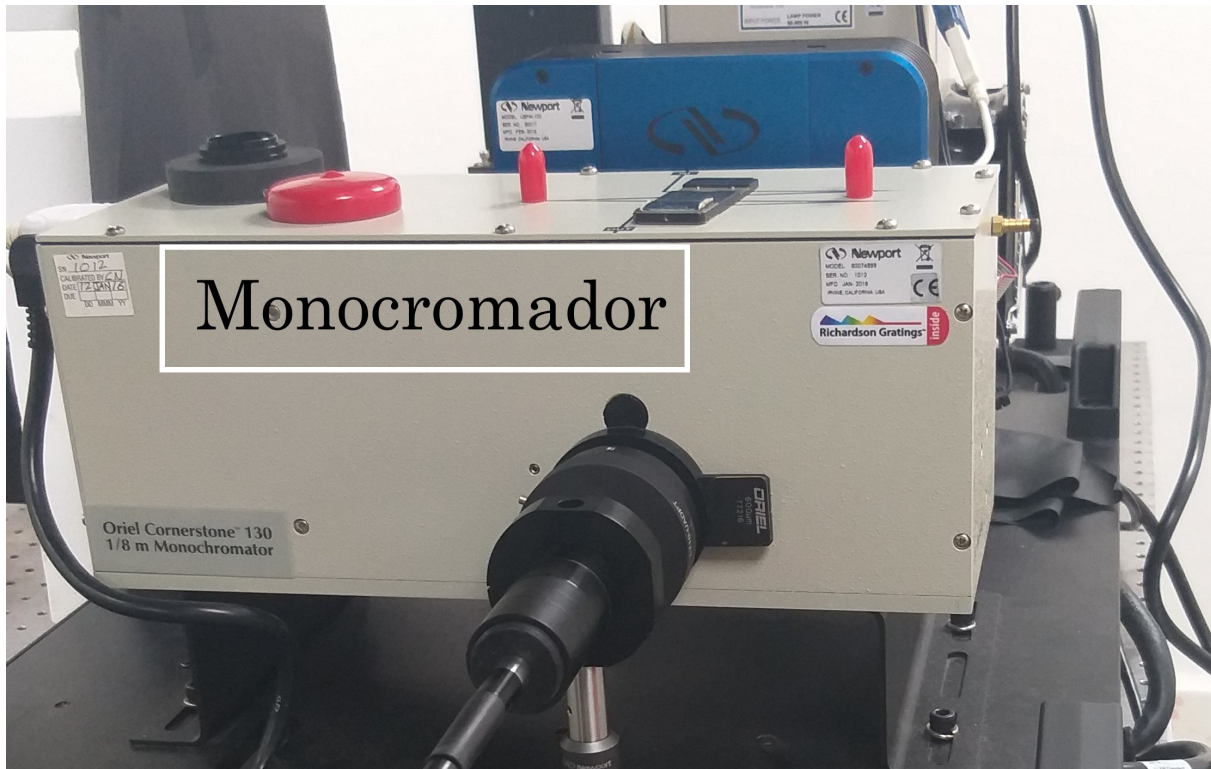


Figura 6. Monocromador Cornerstone 130 1/8m.

En la tabla 1, 2 se muestra las especificaciones del monocromador Cornestone 130 1/8 y las especificaciones de rendimiento de la cámara monocromática Stringray F-145, se debe tener en cuenta que la iluminación la entrega el monocromador Cornestone a partir de una fibra óptica.

Tabla 1
Especificaciones del monocromador Cornerstone 130 1/8m.

Longitud focal	130
F/#	F/3.7
Método de selección de longitud de onda	Motorizado
Rango de longitud de onda utilizable	180 a 2500 nm
Resolución espectral	Rejilla y ancho de corte dependiente
Exactitud de longitud de onda	0.50 nm
Precisión de longitud de onda	0.11 nm
Velocidad de respuesta máxima	350 nm / s con 1200 líneas / mm de rejilla
Luz extraviada	0.03 %
Puertos	1puerto de entrada, 1 puerto de salida
Control de obturador	Software, Hand Controller, low-level commands
Tiempo mínimo de exposición del obturador	0.2 s
Velocidad de repetición máxima del obturador	0.5 Hz
Compatibilidad motorizada de la rueda del filtro	Rueda de filtro modelo 74010, rueda de filtro Apex2
Requisitos de software de utilidad	Windows 7 o 10, sistema operativo de 32 bits o 64 bits (Software compatible con Windows XP también disponible)
Compatible con el software básico TracQ	Si
Controlador de mano 74009 compatible	Si
100-240 VAC, 47-63 Hz	100-240 VAC, 47-63 Hz
Peso	6.9 lb (3 kg)

Tabla 2

Especificaciones de rendimiento de la cámara monocromática Stringray F-145.

Interfaz	IEEE 1394b - 800 Mb / s, 2 puertos, conexión en cadena
Resolución	1388 (H) 1038 (V)
Sensor	Sony ICX285
Tipo de sensor	CCD progresivo
Tamaño del sensor	Tipo 2/3
Tamaño de píxel	6.45 μm 6.45 μm
Montura de lente (predeterminado)	C-Mount
Max. velocidad de fotogramas a resolución completa	16 fps
ADC	14 Bit
Búfer de imagen (RAM)	Hasta 128 MByte

3.2. Bases de datos capturadas

Con la arquitectura óptica que se diseñó se adquirieron 29 bandas del espectro con longitudes de onda de 360nm hasta 920nm con paso de 20nm, cada imagen que se adquirió tiene una resolución espacial de 776X1030 píxeles, teniendo en cuenta que los objetos son medicamentos en presentación de tableta tales como: Acetaminofen, dolex, metocarbamol, mioflex, ibuprofeno, calmidol,

ácido acetilsalicílico y aspirina. En la tabla 3 se indica la cantidad de datos que se adquirieron con su respectiva nombre.

Tabla 3

Cantidad de medicamentos adquiridos en presentación de tableta.

Medicamentos	
Nombre	Cantidad
Acetaminofen	26
Ácido acetilicilico	27
Aspirina	26
Calmidon	24
Dolex	21
Ibuprofeno	26
Metocarbamol	21
Mioflex	20
Total	191

Las firmas espectrales promedio que se obtuvieron en cada tipo de medicamento, se observa en la Figura 7, se puede resaltar que aunque su representación RGB aparentemente todos son blancos iguales, en el espectro se logra apreciar diferencias sustanciales que ayudan a su clasificación.

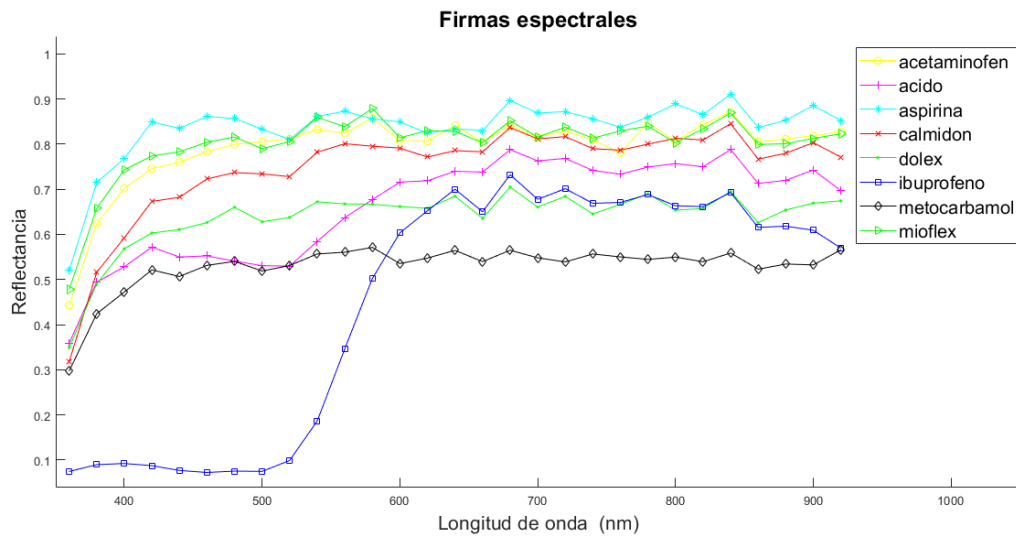


Figura 7. Firmas espectrales promedio de cada medicamento.

3.3. Algoritmo de segmentación de imágenes espectrales

Se diseña un algoritmo de segmentación el cual permite identificar las diferentes regiones de cada imagen para seleccionar un área de interés en la imagen y separar el fondo y los objetos, cada imagen de la base de datos adquirida contienen 3 o 4 medicamentos ya sean del mismo tipo o mezclados, en la Figura 8 se visualiza una imagen de la base de datos adquirida la cual contiene diferente tipos medicamentos.



Figura 8. Medicamento en presentación de tableta sin preprocesamiento.

La base de datos que se adquirió necesita de un preprocesamiento para separar el fondo y la zona de interés, en la Figura 9 se aprecia el diagrama de flujo para realizar esta tarea. A cada imagen de la base de datos adquirida se calcula la reflectancia, la cual esta dada por la ecuación 1.

$$Ref(x,y,\lambda) = \frac{Im(x,y,\lambda) - N(x,y,\lambda)}{B(x,y,\lambda) - N(x,y,\lambda)} \quad (1)$$

Con el fin de normalizar cada imagen, la misma base de datos pasa por el bloque de segmentación cuyo fin es construir las mascarar binarias la cual se asegura en eliminar todos los píxeles que no deseamos analizar, las mascarar binarias se construyen a partir de un índice de umbralización global permitiendo dividir la imagen en dos regiones (objeto y fondo), con este proceso nos permite eliminar la base donde se colocaron los medicamentos y agentes externos de cada imagen. Pos-

teriormente se aplica un producto punto a punto entre el cálculo que se realizó de la reflectancia de la base de datos y las máscaras binarias de cada imagen con esto se obtienen los datos segmentados. Luego se procede a recortar las imágenes de los medicamentos con el fin de separar cada medicamento ya que cada imagen contiene 3 o 4 medicamentos, esto se realiza con el fin de reducir el tamaño de cada imagen para no sobrecargar los recursos computacionales, de este modo se entrena la arquitectura de aprendizaje de máquina para ser clasificados estos medicamentos.

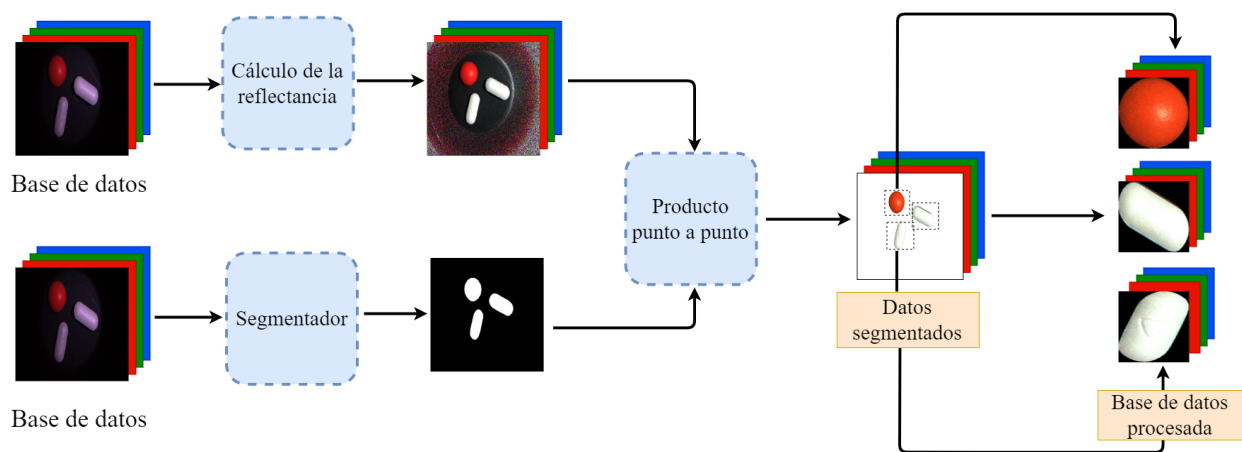


Figura 9. Proceso de segmentación de la base de datos para la separación del fondo y de los medicamentos.

3.4. Modelos de aprendizaje de máquina

Se diseñan dos enfoques de aprendizaje de máquina para clasificar el conjunto de medicamentos adquiridos, el primer enfoque consiste en una arquitectura unidimensional la cual se les dio el nombre de **SpectralNet1**, entrenada a partir de firmas espectrales de la base de datos segmentada, el segundo enfoque consiste en dos arquitecturas empleando información espacial de la escena con la base de datos segmentada las cuales se les dio el nombre de **CarlosNet1** y **CarlosNet2**. Estas arquitecturas fueron entrenadas en un computador Dell con doble procesador, procesador

Intel(R) Xeon(R) CPU ES- 2697 v3 @ 2.60Ghz, cuenta con una memoria Ram de 192 GB y una GPU NVIDIA Quadro K6000 de 12GB.

3.4.1. Modelo de aprendizaje de máquina unidimensional entrenado a partir de firmas espectrales de los datos segmentados. El diseño de la arquitectura de aprendizaje de máquina **SpectralNet1** fue inspirada en la arquitectura de la red neuronal **AlexNet**, esta arquitectura consta de cinco capas convolucionales, algunas de las cuales son seguidas de capas de max-pooling utilizada para reducir el tamaño de la imagen, también incluye tres capas totalmente conectadas. En redes neuronales convolucionales y en aprendizaje de maquina existe un termino conocido como sobre-entrenamiento se da cuando la red se ajustada a parámetros muy específicos del entrenamiento, obteniendo alto rendimiento en entrenamiento pero bajo en pruebas. Existe una técnica para reducirlo, donde la arquitectura de **AlexNet** (Gonzalez, 2007) propone aplicar en las capas completamente conectadas el método de dropout siendo esta una capa, la cual permite deshabilitar de forma aleatoria un porcentaje de las neuronas de esta red, con esto se incrementó la cantidad de capas y filtros mejorando la precisión en la clasificación de imágenes, para que el entrenamiento de la arquitectura **AlexNet** fuera de forma rápida usaron dos GPUs GTX 580 de 3GB durando al rededor de dos semanas para entrenar este modelo. El modelo de **SpectralNet1** que se propone consta de 14 capas en total, que se distribuyen de la siguiente manera:

- **Datos de entrada:** La arquitectura consta de una capa de entrada cuyas dimensiones son de 29x1 haciendo referencia a las 29 bandas espectrales de los medicamentos en presentación de tableta.

■ Sección de extracción de características

● Primera Capa.

Pasa por una capa convolucional con función de activación RELU, realizando la operación convolución entre los datos de entrenamiento y un núcleo propuesto de 2, con esto se extraen características espectrales, esta capa de convolución consta de 32 filtros en total, se realiza la normalización por lotes con los resultados de la capa anterior ajustando y escalando el método de activaciones, existiendo características de 0 a 1 y algunas de 1 a 1000, lo cual se debe normalizarlas para acelerar el proceso de aprendizaje de esta red neuronal.

● Segunda Capa

Pasa por una capa de MaxPooling con zancadas de 2 lo cual permite reducir la dimensionalidad de las firmas espectrales de esta manera se reduce los parámetros de entrenamiento de la red

● Tercera Capa

Pasa por una capa convolucional y función de activación RELU continuando con la operación convolución con respecto a la salida de la capa anterior, esta capa convolucional consta de un núcleo de 2 con 64 filtros, luego se realiza la normalización con los resultados anteriores.

● Cuarta Capa

Pasa por una capa de MaxPooling con zancadas de 2 para reducir los datos de la capa

anterior.

- **Quinta Capa**

Pasa por una capa convolucional y función de activación RELU continuando con la operación convolución con respecto a la salida de la capa anterior, esta capa convolucional consta de un núcleo de 3 con 128 filtros, luego se realiza la normalización con los resultados anteriores.

- **Sexta Capa**

Pasa por una capa de MaxPooling con zancadas de 2 para reducir los datos de la capa anterior.

- **Séptima Capa**

Pasa por una capa de Dropout del 0.2% buscando desconectar aleatoriamente neuronas para darles a otras mas pesos y de esta manera prevenir el sobre-entrenamiento de la red neuronal.

- **Clasificación:** El resultado de las capas anteriores pasa por las capas totalmente conectadas.

- **Octava Capa**

Los datos de las capas de extracción de características pasa por la capa de aplanamiento para generar un vector unidimensional.

- **Novena Capa**

Se realiza la primera capa Fully Connected entre el vector unidimensional de la capa

anterior, esta capa contiene 512 neuronas.

- **Décima Capa**

Pasa por una capa de Dropout del 0.2%.

- **Onceava Capa**

De las 512 neuronas de la capa anterior se interconectan con una capa Fully Connected de 1024 neuronas.

- **Doceava Capa**

De las 1024 neuronas de la capa anterior se interconectan con una capa Fully Connected con 8 neuronas.

- **Treceava Capa**

La capa anterior pasa por la capa Softmax la cual es una función de regresión, permitiendo clasificar las múltiples categorías ingresadas en los datos de entrenamiento.

- **Capa de salida**

- **Catorceava Capa**

Pasa por la capa de clasificación donde mostrara el porcentaje de éxito de las diferentes categorías de las firmas espectrales de los diferentes medicamentos en presentación de tableta.

- **Parámetros de la arquitectura SpectralNet1:** Esta arquitectura tiene un total de 760,488 de parámetros, constituidos por 760,040 de parámetros entrenables dados por las capas con-

volucionales y fully connected y 448 parámetros no entrenables dado por la capa Dropout.

El diseño de esta arquitectura se observa en la Figura 10.

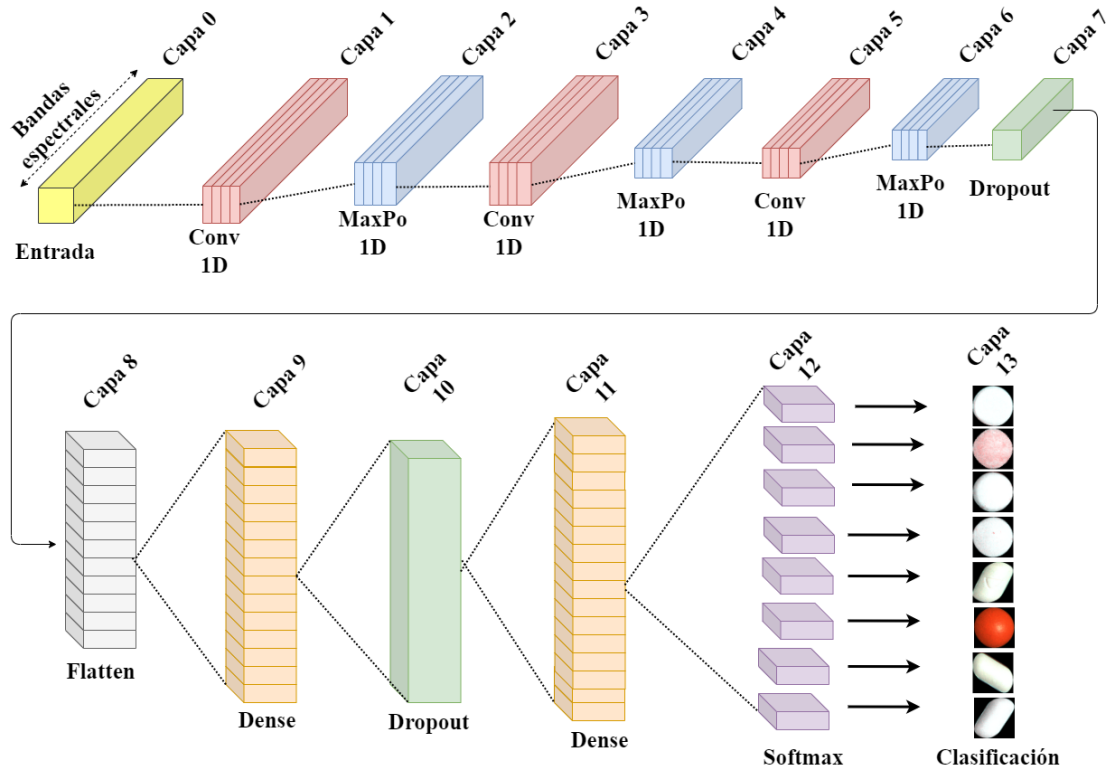


Figura 10. Modelo de neuronal convolucional 1D SpectralNet1.

En la tabla 4 se resume la composición de las capas y sus parámetros entrenables de la arquitectura **SpectralNet1**, indicando el tamaño de salida de cada capa. Es importante aclarar que las capas MaxPooling no tienen parámetros entrenables.

Tabla 4

Composición de capas y parámetros entrenables de la red neuronal convolucional 1D SpectralNet1.

Arquitectura SpectralNet1				
	Tipo de capa	Dimensión	Output shape	# de Parámetros
0	Input	(29,1)	(29,1)	-
1	Convolution 1D	32x2	(29,32)	96
	BatchNormalization			128
2	MaxPooling1D	2	(14,32)	-
3	Convolution 1D	64x2	(14,64)	4160
	BatchNormalization			256
4	MaxPooling1D	2	(7,64)	-
5	Convolution 1D	128x3	(7,128)	24704
	BatchNormalization			512
6	MaxPooling1D	2	(3,128)	-
7	Dropout (0.2 %)	-	(3,128)	-
8	Flatten	-	384	-
9	Dense	512	512	197120
10	Dropout (0.2 %)	-	512	-
11	Dense	1024	1024	525312
12	Dense	8	8	8200
13	Softmax	8	8	-
14	Clasificación	-	-	-

3.4.2. Modelo de aprendizaje de máquina empleando información espacial de la escena a partir de los datos segmentados. El diseño de las arquitecturas de aprendizaje de máquina **CarlosNet1** y **CarlosNet2** fueron inspirados en la arquitectura de la red neuronal **ResNet18** (He *et al.*, 2016). Esta arquitectura consta de tiene 18 capas de profundidad y puede clasificar imágenes en 1000 categorías de objetos la red ha aprendido representaciones de características ricas para una amplia gama de imágenes. El modelo **CarlosNet1** que se propone consta de 16 capas en total, que se distribuyen de la siguiente manera

- **Datos de entrada:** La arquitectura posee una capa de entrada la cual realiza un reescalado a la base de datos segmentada para obtener dimensiones 24x240 con una profundidad de 29 bandas espectrales.

- **Sección de extracción de características**

- **Primera Capa.**

- Pasa por una capa convolucional con función de activación RELU, realizando la operación convolución entre los datos de entrenamiento y un núcleo propuesto de 3x3, con esto se extraen características principales tales como bordes, líneas y curvas, esta capa de convolución consta de 8 filtros en total, luego Se realiza la normalización por lotes con los resultados de la capa anterior.

- **Segunda Capa**

- Pasa por una capa de MaxPooling con zancadas de 2x2 lo cual permite reducir la di-

mensionalidad espacial de la imagen de esta manera se reduce los parámetros de entrenamiento de la red.

- **Tercera Capa**

Pasa por una capa convolucional con función de activación RELU continuando con la operación convolución con respecto a la salida de la capa anterior, esta capa convolucional consta de un núcleo de 3x3 con 16 filtros, luego se realiza la normalización con los resultados anteriores.

- **Cuarta Capa**

Pasa por una capa de MaxPooling con zancadas de 2x2 para reducir los datos de la capa anterior.

- **Quinta Capa**

Pasa por una capa convolucional con función de activación RELU continuando con la operación convolución con respecto a la salida de la capa anterior, esta capa convolucional consta de un núcleo de 3x3 con 32 filtros, luego se realiza la normalización con los resultados anteriores.

- **Sexta Capa**

Pasa por una capa convolucional con función de activación RELU continuando con la operación convolución con respecto a la salida de la capa anterior, esta capa convolucional consta de un núcleo de 3x3 con 64 filtros, luego se realiza la normalización con los resultados anteriores.

- **Séptima Capa**

Pasa por una capa de MaxPooling con zancadas de 2x2 para reducir los datos de la capa anterior.

- **Octava Capa**

Pasa por una capa convolucional con función de activación RELU continuando con la operación convolución con respecto a la salida de la capa anterior, esta capa convolucional consta de un núcleo de 3x3 con 128 filtros, luego se realiza la normalización con los resultados anteriores.

- **Novena Capa**

Pasa por una capa de MaxPooling con zancadas de 2x2 para reducir los datos de la capa anterior.

- **Décima Capa**

Pasa por una capa convolucional con función de activación RELU continuando con la operación convolución con respecto a la salida de la capa anterior, esta capa convolucional consta de un núcleo de 3x3 con 256 filtros, luego se realiza la normalización con los resultados anteriores.

- **Onceava Capa**

Pasa por una capa de MaxPooling con zancadas de 2x2 para reducir los datos de la capa anterior.

- **Doceava Capa**

Pasa por una capa convolucional con función de activación RELU continuando con la operación convolución con respecto a la salida de la capa anterior, esta capa convolucional consta de un núcleo de 3x3 con 512 filtros, luego se realiza la normalización con los resultados anteriores.

- **Treceava Capa**

Pasa por una capa de MaxPooling con zancadas de 2x2 para reducir los datos de la capa anterior.

- **Clasificación:** La salida de la capa anterior pasa por un aplanamiento para generar un vector unidimensional y pasar por las capas totalmente conectadas.

- **Catorceava Capa**

El vector unidimensional de todas las características de la capa anterior pasa por una capa de Fully Connected de 8 nodos.

- **Quinceava Capa**

Pasa por la capa Softmax la cual es una función de regresión, permitiendo clasificar las múltiples categorías ingresadas en los datos de entrenamiento.

- **Capa de salida**

- **Dieciseisava Capa**

Pasa por la capa de clasificación donde mostrara el porcentaje de éxito de las diferentes categorías de medicamentos en presentación de tableta.

- **Parámetros de la arquitectura CarlosNet1:** Esta arquitectura tiene un total de 1'614.232 de parámetros en total, constituidos por 1'614.232 parámetros entrenables dados por las capas convolucionales y fully connected y 0 parámetros no entrenables.

El diseño de esta arquitectura se observa en la Figura 11.

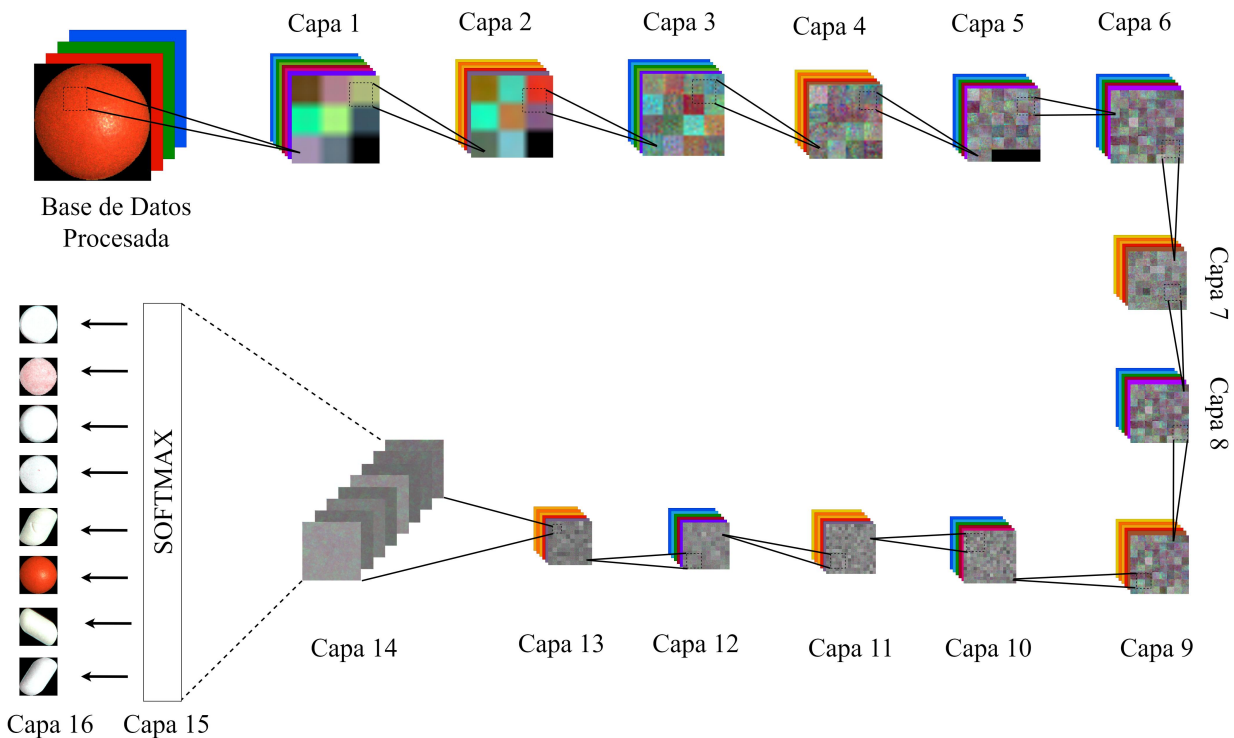


Figura 11. Modelo de red neuronal convolucional 2D CarlosNet1.

En la Figura 12 se muestra las características aprendidas del modelo **CarlosNet1**, para esto se ingreso una imagen para obtener la visualización de las características aprendidas de las diferentes capas convolucionales de esta red neuronal convolucional, para esto se han extraído los valores de los filtros que han aprendido cada capa convolucional, aplicando la convolución entre

una imagen de entrada y los valores del filtro aprendido. Solo se muestra algunos de los ejemplos debido a que en algunos casos son demasiados filtros para mostrarlos en esta imagen.

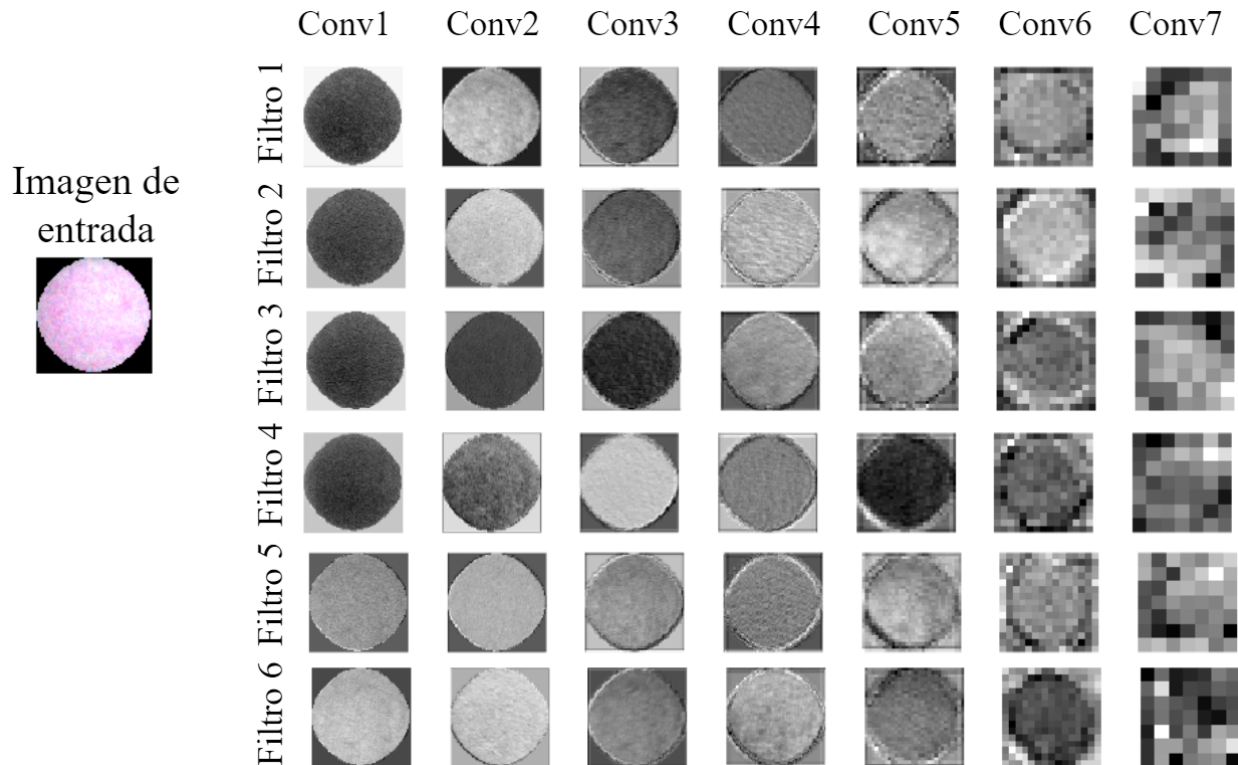


Figura 12. Ejemplos de salida de las capas de la red convolucional CarlosNet1.

La segunda arquitectura **CarlosNet2** que se diseñó consta de 16 capas en total, que se distribuyen de la siguiente manera:

- **Datos de entrada:** La arquitectura posee una capa de entrada la cual realiza un reescalado a la base de datos segmentada para obtener dimensiones 24x240 con una profundidad de 29 bandas espectrales.
- **Sección de extracción de características**

- **Primera Capa.**

Pasa por una capa convolucional con función de activación RELU, realizando la operación convolución entre los datos de entrenamiento y un núcleo propuesto de 3x3, con esto se extraen características principales tales como bordes, líneas y curvas, esta capa de convolución consta de 8 filtros en total, luego se realiza la normalización por lotes con los resultados de la capa anterior.

- **Segunda Capa**

Pasa por una capa de MaxPooling con zancadas de 2x2 lo cual permite reducir la dimensionalidad espacial de la imagen de esta manera se reduce los parámetros de entrenamiento de la red.

- **Tercera Capa**

Pasa por una capa convolucional con función de activación RELU continuando con la operación convolución con respecto a la salida de la capa anterior, esta capa convolucional consta de un núcleo de 5x5 con 16 filtros, luego se realiza la normalización con los resultados anteriores.

- **Cuarta Capa**

Pasa por una capa de MaxPooling con zancadas de 2x2 para reducir los datos de la capa anterior.

- **Quinta Capa**

Pasa por una capa convolucional con función de activación RELU continuando con la

operación convolución con respecto a la salida de la capa anterior, esta capa convolucional consta de un núcleo de 9×9 con 32 filtros, luego se realiza la normalización con los resultados anteriores.

- **Sexta Capa**

Pasa por una capa convolucional con función de activación RELU continuando con la operación convolución con respecto a la salida de la capa anterior, esta capa convolucional consta de un núcleo de 15×15 con 64 filtros, luego se realiza la normalización con los resultados anteriores.

- **Séptima Capa**

Pasa por una capa de MaxPooling con zancadas de 2×2 para reducir los datos de la capa anterior.

- **Octava Capa**

Pasa por una capa convolucional y función de activación RELU continuando con la operación convolución con respecto a la salida de la capa anterior, esta capa convolucional consta de un núcleo de 3×3 con 128 filtros, luego se realiza la normalización con los resultados anteriores.

- **Novena Capa**

Pasa por una capa de MaxPooling con zancadas 2×2 para reducir los datos de la capa anterior.

- **Décima Capa**

Pasa por una capa convolucional y función de activación RELU continuando con la operación convolución con respecto a la salida de la capa anterior, esta capa convolucional consta de un núcleo de 32×32 con 256 filtros, luego se realiza la normalización con los resultados anteriores.

- **Onceava Capa**

Pasa por una capa de MaxPooling con zancadas 2×2 para reducir los datos de la capa anterior.

- **Doceava Capa**

Pasa por una capa convolucional y función de activación RELU continuando con la operación convolución con respecto a la salida de la capa anterior, esta capa convolucional consta de un núcleo de 32×32 con 512 filtros, luego se realiza la normalización con los resultados anteriores.

- **Treceava Capa**

Pasa por una capa de MaxPooling con zancadas de 2×2 para reducir los datos de la capa anterior.

- **Clasificación:** La salida de la capa anterior pasa por un aplanamiento para generar un vector unidimensional y pasar por las capas totalmente conectadas.

- **Catorceava Capa**

El vector unidimensional de todas las características de la capa anterior pasa por una capa de Fully Connected de 8 nodos.

- **Quinceava Capa**

Pasa por la capa Softmax la cual es una función de regresión, permitiendo clasificar las múltiples categorías ingresadas en los datos de entrenamiento.

- **Capa de salida**

- **Dieciseisava Capa**

Pasa por la capa de clasificación donde mostrara el porcentaje de éxito de las diferentes categorías de medicamentos en presentación de tableta.

- **Parámetros de la arquitectura CarlosNet2:** Esta arquitectura tiene un total de 168'392.858 de parámetros en total, constituidos por 168'392.858 parámetros entrenables dados por las capas convolucionales y fully connected y 0 parámetros no entrenables.

El diseño de esta arquitectura se observa en la Figura 13.

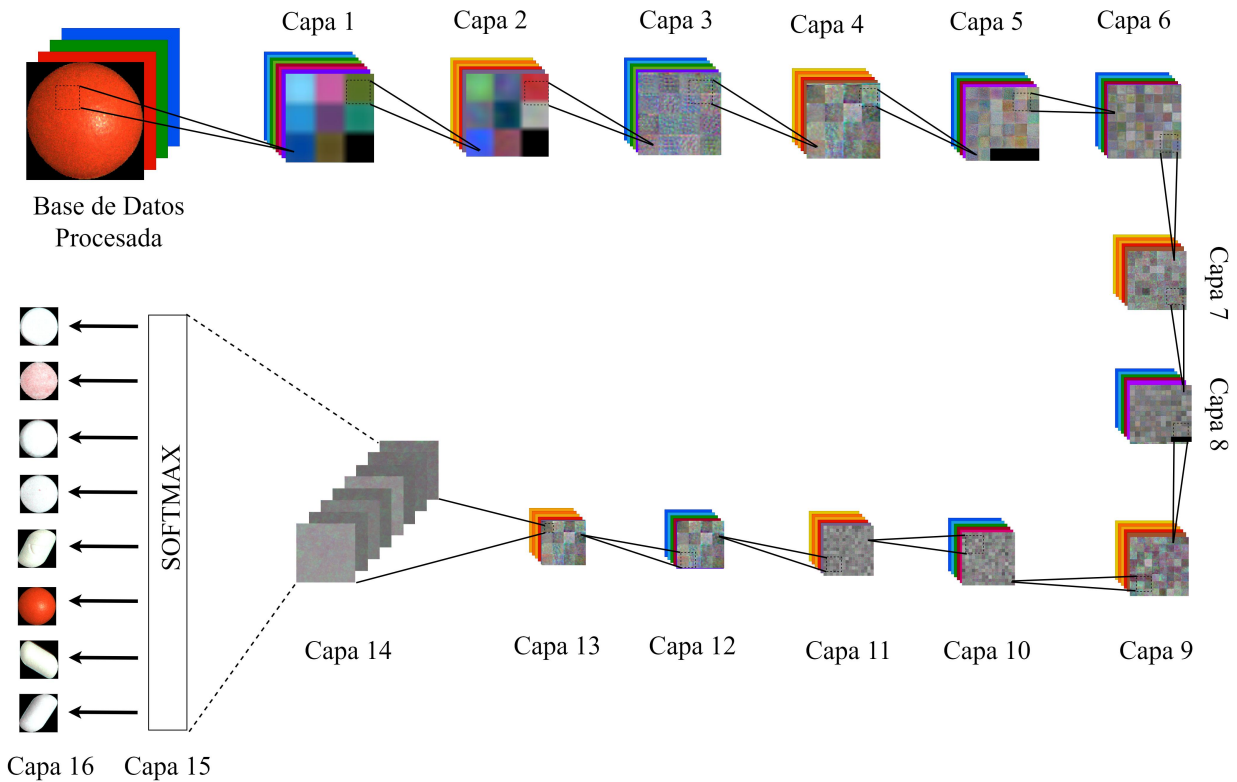


Figura 13. Modelo de red neuronal convolucional 2D CarlosNet2.

En la Figura 14 se muestra las características aprendidas del modelo **CarlosNet2**, para esto se ingreso una imagen para obtener la visualización de las características aprendidas de las diferentes capas convolucionales de esta red neuronal convolucional, para esto se han extraído los valores de los filtros que han aprendido cada capa convolucional, aplicando la convolución entre una imagen de entrada y los valores del filtro aprendido. Solo se muestran algunos de los ejemplos debido a que en algunos casos son demasiados filtros para mostrarlos en esta imagen.

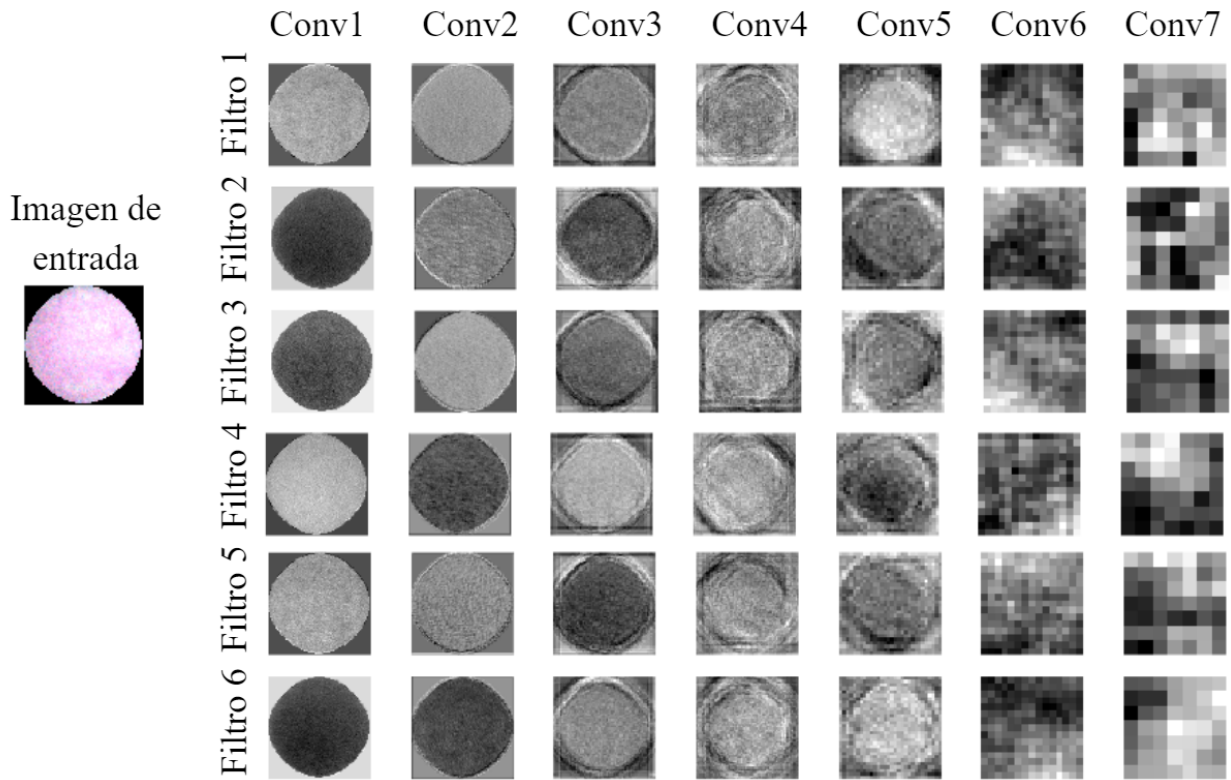


Figura 14. Ejemplos de salida de las capas de la red convolucional CarlosNet2.

En la tabla 5 se resume la composición de las capas y sus parámetros entrenables de las arquitecturas **CarlosNet1** y **CarlosNet2**, indicando el tamaño de salida de cada capa. Es importante aclarar que las capas MaxPooling no tienen parámetros entrenables.

Tabla 5

Comparación de las arquitecturas de las redes neuronales convolucionales 2D.

	Arquitectura CarlosNet1				Arquitectura CarlosNet2			
	Tipo de capa	Dimensión	Output shape	# de Parámetros	Tipo de capa	Dimensión	Output shape	# de Parámetros
0	Input	(240,240,29)	(240,240,29)	-	Input	(240,240,29)	(240,240,29)	-
1	Convocional 2D	8x3x3	(240,240,8)	2096	Convocional 2D	8x3x3	(240,240,8)	2098
	BatchNormalization			16	BatchNormalization			16
2	MaxPooling 2D	2x2	(120,120,8)	-	MaxPooling 2D	2x2	(120,120,8)	-
3	Convocional 2D	16x3x3	(120,120,16)	1168	Convocional 2D	16x5x5	(120,120,16)	3216
	BatchNormalization			32	BatchNormalization			32
4	MaxPooling 2D	2x2	(60,60,16)	-	MaxPooling 2D	2x2	(60,60,16)	-
5	Convocional 2D	32x3x3	(60,60,32)	4640	Convocional 2D	32x9x9	(60,60,32)	41504
	BatchNormalization			64	BatchNormalization			64
6	Convocional 2D	64x3x3	(60,60,64)	18496	Convocional 2D	64x15x15	(60,60,64)	460864
	BatchNormalization			128	BatchNormalization			128
7	MaxPooling 2D	2x2	(30,30,64)	-	MaxPooling 2D	2x2	(30,30,64)	-
8	Convocional 2D	18x3x3	(30,30,128)	73856	Convocional 2D	128x3x3	(30,30,128)	73856
	BatchNormalization			256	BatchNormalization			256
9	MaxPooling 2D	2x2	(15,15,128)	-	MaxPooling 2D	2x2	(15,15,128)	-
10	Convocional 2D	256x3x3	(15,15,256)	295168	Convocional 2D	256x32x32	(15,15,256)	33554688
	BatchNormalization			256	BatchNormalization			512
11	MaxPooling 2D	2x2	(7,7,256)	-	MaxPooling 2D	2x2	(7,7,256)	-
12	Convocional 2D	512x3x3	(7,7,512)	1180160	Convocional 2D	512x32x32	(7,7,512)	134217728
	BatchNormalization			1024	BatchNormalization			1024
13	MaxPooling 2D	2x2	(3,3,512)	-	MaxPooling 2D	2x2	(3,3,512)	-
14	Dense	8	8	36872	Dense	8	8	36872
15	Softmax	8	8	-	Softmax	8	8	-
16	Clasificación	-	-	-	Clasificación	-	-	-

4. Simulación y Resultados

4.1. Rendimiento del modelo unidimensional entrenado a partir de firmas espectrales de los datos segmentados

Para evaluar el rendimiento de este modelo se utilizó el conjunto de firmas espectrales adquiridas, este conjunto de datos fue dividido en entrenamiento y validación, los conjuntos de datos de entrenamiento fueron formados con un 72 % del conjunto de datos de las firmas espectrales y el 28 % restante se utilizó para formar los conjuntos de validación, este modelo se le dio el nombre de **SpectralNet1** para poderlos diferenciarlas de las otras redes. En la tabla 6 se muestra el número de elementos del conjunto de datos de entrenamiento y validación.

Tabla 6

Conjunto de firmas espectrales separadas en entrenamiento y validación.

Conjunto de firmas espectrales	
Entrenamiento	Validación
360393	145704

Para realizar el entrenamiento de la red **SpectralNet1** se hizo con 500 épocas lo cual fue suficiente para que la función de costo llegue a converger. Para la red **SpectralNet1** la matriz de confusión en entrenamiento que se obtuvo con este modelo se aprecia en la Figura 15, obteniendo un 75.4 % de aciertos, la matriz de confusión en validación se aprecia en la Figura 16, obteniendo 58.9 % de aciertos con esta red se obtiene una precisión en entrenamiento de 75.4 % y una precisión en validación de 58.9 %, se logra intuir que al entrenar esta arquitectura con las firma espectrales es difícil tener un clasificador con excelentes resultados esto se debe en la similitud entre las firmas

espectrales de cada medicamento puesto que se tiene medicamentos que tienen mismo color.

Confusion Matrix

Output Class	acetaminofen	9148 2.5%	14 0.0%	1193 0.3%	137 0.0%	2493 0.7%	0 0.0%	1921 0.5%	1967 0.5%	54.2% 45.8%
	acido	86 0.0%	10019 2.8%	5 0.0%	72 0.0%	67 0.0%	0 0.0%	94 0.0%	82 0.0%	96.1% 3.9%
	aspirina	1368 0.4%	15 0.0%	27678 7.7%	27 0.0%	2805 0.8%	0 0.0%	2128 0.6%	3198 0.9%	74.4% 25.6%
	calmidon	359 0.1%	218 0.1%	33 0.0%	37141 10.3%	877 0.2%	15 0.0%	1298 0.4%	545 0.2%	91.7% 8.3%
	dolex	3018 0.8%	191 0.1%	2089 0.6%	473 0.1%	23089 6.4%	0 0.0%	5048 1.4%	6678 1.9%	56.9% 43.1%
	ibuprofeno	0 0.0%	0 0.0%	0 0.0%	3 0.0%	0 0.0%	85846 23.8%	0 0.0%	2 0.0%	100.0% 0.0%
	metocarbamol	3486 1.0%	102 0.0%	2034 0.6%	988 0.3%	7983 2.2%	0 0.0%	38062 10.6%	7870 2.2%	62.9% 37.1%
	mioflex	3898 1.1%	117 0.0%	4272 1.2%	373 0.1%	11016 3.1%	4 0.0%	8068 2.2%	40680 11.3%	59.4% 40.6%
		42.8% 57.2%	93.8% 6.2%	74.2% 25.8%	94.7% 5.3%	47.8% 52.2%	100.0% 0.0%	67.2% 32.8%	66.7% 33.3%	75.4% 24.6%
		acetaminofen	acido	aspirina	calmidon	dolex	ibuprofeno	metocarbamol	mioflex	
	Target Class									

Figura 15. Matriz de confusión de entrenamiento para la red SpectralNet1.

Confusion Matrix

Output Class	acetaminofen	1138 0.8%	27 0.0%	287 0.2%	76 0.1%	1214 0.8%	0 0.0%	1222 0.8%	862 0.6%	23.6% 76.4%
	acido	32 0.0%	2524 1.7%	24 0.0%	152 0.1%	120 0.1%	0 0.0%	513 0.4%	21 0.0%	74.5% 25.5%
	aspirina	1494 1.0%	11 0.0%	5122 3.5%	7 0.0%	738 0.5%	0 0.0%	780 0.5%	1901 1.3%	50.9% 49.1%
	calmidon	813 0.6%	80 0.1%	427 0.3%	9062 6.2%	1203 0.8%	0 0.0%	2221 1.5%	498 0.3%	63.4% 36.6%
	dolex	2443 1.7%	74 0.1%	1930 1.3%	133 0.1%	5390 3.7%	0 0.0%	3425 2.4%	3865 2.7%	31.2% 68.8%
	ibuprofeno	0 0.0%	0 0.0%	0 0.0%	22 0.0%	0 0.0%	43464 29.8%	6 0.0%	0 0.0%	99.9% 0.1%
	metocarbamol	1722 1.2%	55 0.0%	2625 1.8%	295 0.2%	5493 3.8%	0 0.0%	11588 8.0%	4660 3.2%	43.8% 56.2%
	mioflex	2412 1.7%	96 0.1%	4004 2.7%	213 0.1%	6306 4.3%	0 0.0%	5425 3.7%	7489 5.1%	28.9% 71.1%
		11.3% 88.7%	88.0% 12.0%	35.5% 64.5%	91.0% 9.0%	26.3% 73.7%	100% 0.0%	46.0% 54.0%	38.8% 61.2%	58.9% 41.1%
		acetaminofen	acido	aspirina	calmidon	dolex	ibuprofeno	metocarbamol	mioflex	
	Target Class									

Figura 16. Matriz de confusión de validación para la red SpectralNet1.

4.2. Rendimiento del modelo bidimensional entrenado a partir de los datos segmentados

Para evaluar el rendimiento de los modelos **CarlosNet1** y **CarlosNet2** se entrena con el conjunto de datos segmentados, este conjunto de datos se divide en entrenamiento, validación y prueba, las cuales se realizaron dos simulaciones a cada arquitectura propuesta, en la primera simulación los conjuntos de datos de entrenamiento fueron formados con un 76% del conjunto de datos, el 19% se utilizó para formar los conjuntos de validación y el 5% restante para formar el conjunto de datos de prueba, se dividió el conjunto de datos de esta manera porque se obtuvieron los mejores resultados para la tarea de clasificación, para la segunda simulación el conjunto de datos de entrenamiento fueron formados con un 47% del conjunto de datos, el 12% se utilizó para formar el conjunto de validación y el 41% restante para formar el conjunto de datos de prueba, se dividió el conjunto de datos de esta manera, para observar el rendimiento de estos modelos con ausencia de datos de entrenamiento. En cuadro 7 se muestra el número total de características de cada conjunto de datos ya separados en entrenamiento, validación y prueba, estos fueron usados para realizar las simulaciones 1 y 2.

Tabla 7

Conjunto de imágenes separadas en entrenamiento, validación y prueba.

	Cantidad de imágenes				Numero de clases
	Entrenamiento	Validación	Prueba	Total	
Simulación 1	146	37	8	191	8
Simulación 2	89	22	80	191	8

Para realizar el entrenamiento de de la red **CarlosNet1** se hizo con 99 épocas ya que fueron suficientes para que la red aprendiera con satisfacción el conjunto de datos de entrenamiento. Para la primera simulación la matriz de confusión en entrenamiento que se obtuvo con este modelo se aprecia en la Figura 17, obteniendo un 100% de aciertos, queriendo decir que ha aprendido con éxito el conjunto de datos de entrenamiento, la matriz de confusión en validación se aprecia en la Figura 18, obteniendo un 97.3% de aciertos con este porcentaje se puede que la red diseñada tiene un buen rendimiento para clasificar estos medicamentos, se obtiene la matriz de confusión para los datos de prueba la cual se aprecia en la Figura 19, obteniendo un 87.9% de aciertos lo cual a fallado en uno con estos resultados se puede decir el alto rendimiento de esta red, de 8 muestras falla en una.

Confusion Matrix

Output Class	acetaminofen	20 13.7%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	acido	0 0.0%	21 14.4%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	aspirina	0 0.0%	0 0.0%	20 13.7%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	calmidon	0 0.0%	0 0.0%	0 0.0%	18 12.3%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	dolex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	16 11.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	ibuprofeno	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	20 13.7%	0 0.0%	0 0.0%	100% 0.0%
	metocarbamol	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	16 11.0%	0 0.0%	100% 0.0%
	mioflex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	15 10.3%	100% 0.0%
		100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%
	acetaminofen	acido	aspirina	calmidon	dolex	ibuprofeno	metocarbamol	mioflex		
	Target Class									

Figura 17. Matriz de confusión de entrenamiento para la red CarlosNet1.

Confusion Matrix

Output Class	acetaminofen	5 13.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	acido	0 0.0%	5 13.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	aspirina	0 0.0%	0 0.0%	5 13.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	calmidon	0 0.0%	0 0.0%	0 0.0%	4 10.8%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	dolex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	4 10.8%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	ibuprofeno	0 0.0%	0 0.0%	0 0.0%	1 2.7%	0 0.0%	5 13.5%	0 0.0%	0 0.0%	83.3% 16.7%
	metocarbamol	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	4 10.8%	0 0.0%	100% 0.0%
	mioflex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	4 10.8%	100% 0.0%
		100% 0.0%	100% 0.0%	100% 0.0%	80.0% 20.0%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	97.3% 2.7%
		acetaminofen	acido	aspirina	calmidon	dolex	ibuprofeno	metocarbamol	mioflex	
	Target Class									

Figura 18. Matriz de confusión de validación para la red CarlosNet1.

Confusion Matrix

Output Class	acetaminofen	1 12.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	acido	0 0.0%	1 12.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	aspirina	0 0.0%	0 0.0%	1 12.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	calmidon	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	NaN% NaN%
	dolex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 12.5%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	ibuprofeno	0 0.0%	0 0.0%	0 0.0%	1 12.5%	0 0.0%	1 12.5%	0 0.0%	0 0.0%	50.0% 50.0%
	metocarbamol	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 12.5%	0 0.0%	100% 0.0%
	mioflex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 12.5%	100% 0.0%
		100% 0.0%	100% 0.0%	100% 0.0%	0.0% 100%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	87.5% 12.5%
		acetaminofen	acido	aspirina	calmidon	dolex	ibuprofeno	metocarbamol	mioflex	
	Target Class									

Figura 19. Matriz de confusión de prueba para la red CarlosNet1.

Para la segunda simulación, la matriz de confusión que se obtuvo en entrenamiento se apre-

cia en la Figura 20, obteniendo un 95.5 % de aciertos, esto se debe a que se a reducido el conjunto de datos, la matriz de confusión en validación se aprecia en la Figura 21 obteniendo un 72.7 % de aciertos,la matriz de confusión de prueba se aprecia en la Figura 22, obteniendo un 75 % de aciertos lo cual ha fallado en 20, esto se debe a que se ha reducido el conjunto de entrenamiento ocasionando clasificaciones erróneas.

Confusion Matrix

Output Class	acetaminofen	12 13.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	acido	0 0.0%	13 14.6%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	aspirina	0 0.0%	0 0.0%	12 13.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	calmidon	0 0.0%	0 0.0%	0 0.0%	10 11.2%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	dolex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	7 7.9%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	ibuprofeno	0 0.0%	0 0.0%	0 0.0%	1 1.1%	0 0.0%	12 13.5%	0 0.0%	0 0.0%	92.3% 7.7%
	metocarbamol	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	10 11.2%	0 0.0%	100% 0.0%
	mioflex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	3 3.4%	0 0.0%	0 0.0%	9 10.1%	75.0% 25.0%
		100% 0.0%	100% 0.0%	100% 0.0%	90.9% 9.1%	70.0% 30.0%	100% 0.0%	100% 0.0%	100% 0.0%	95.5% 4.5%
		acetaminofen	acido	aspirina	calmidon	dolex	ibuprofeno	metocarbamol	mioflex	
	Target Class									

Figura 20. Matriz de confusión de entrenamiento para la red CarlosNet1.

Confusion Matrix

Output Class	acetaminofen	3 13.6%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	acido	0 0.0%	3 13.6%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	aspirina	0 0.0%	0 0.0%	3 13.6%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	calmidon	0 0.0%	0 0.0%	0 0.0%	1 4.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	dolex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	NaN% NaN%
	ibuprofeno	0 0.0%	0 0.0%	0 0.0%	2 9.1%	0 0.0%	3 13.6%	0 0.0%	0 0.0%	60.0% 40.0%
	metocarbamol	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 4.5%	0 0.0%	0 0.0%	0 0.0%	0.0% 100%
	mioflex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 4.5%	0 0.0%	2 9.1%	3 13.6%	50.0% 50.0%
			100% 0.0%	100% 0.0%	100% 0.0%	33.3% 66.7%	0.0% 100%	0.0% 0.0%	100% 100%	72.7% 27.3%
			acetaminofen	acido	aspirina	calmidon	dolex	ibuprofeno	metocarbamol	mioflex
		Target Class								

Figura 21. Matriz de confusión de validación para la red CarlosNet1.

Confusion Matrix

Output Class	acetaminofen	11 13.8%	0 0.0%	1 1.3%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	91.7% 8.3%
	acido	0 0.0%	11 13.8%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	aspirina	0 0.0%	0 0.0%	10 12.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	calmidon	0 0.0%	0 0.0%	0 0.0%	7 8.8%	0 0.0%	1 1.3%	0 0.0%	0 0.0%	87.5% 12.5%
	dolex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	4 5.0%	1 1.3%	2 2.5%	0 0.0%	57.1% 42.9%
	ibuprofeno	0 0.0%	0 0.0%	0 0.0%	3 3.8%	0 0.0%	9 11.3%	0 0.0%	0 0.0%	75.0% 25.0%
	metocarbamol	0 0.0%	0 0.0%	0 0.0%	0 0.0%	2 2.5%	0 0.0%	2 2.5%	2 2.5%	33.3% 66.7%
	mioflex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	3 3.8%	0 0.0%	5 6.3%	6 7.5%	42.9% 57.1%
			100% 0.0%	100% 0.0%	90.9% 9.1%	70.0% 30.0%	44.4% 55.6%	81.8% 18.2%	22.2% 77.8%	75.0% 25.0%
			acetaminofen	acido	aspirina	calmidon	dolex	ibuprofeno	metocarbamol	mioflex
		Target Class								

Figura 22. Matriz de confusión de prueba para la red CarlosNet1.

Para realizar el entrenamiento de de la red **CarlosNet2** se tuvieron en cuenta las mismas

condiciones de la red **CarlosNet1**. Para la primera simulación la matriz de confusión en entrenamiento se aprecia en la Figura 23, obteniendo un 97.3% de aciertos, la matriz de confusión en validación se aprecia en la Figura 24, obteniendo un 89.2% de aciertos, la matriz de confusión en prueba se aprecia en la Figura 25, obteniendo un 75% de aciertos lo cual a fallado en dos con estos resultados se puede decir que al cambiar los tamaños de las capas convolucionales su resultado en clasificación a fallado en 2 siendo aceptable este modelo para clasificar.

Confusion Matrix

Output Class	acetaminofen	acido	aspirina	calmidon	dolex	ibuprofeno	metocarbamol	mioflex	
acetaminofen	20 13.7%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
acido	0 0.0%	21 14.4%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
aspirina	0 0.0%	0 0.0%	20 13.7%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
calmidon	0 0.0%	0 0.0%	0 0.0%	14 9.6%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
dolex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	16 11.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
ibuprofeno	0 0.0%	0 0.0%	0 0.0%	4 2.7%	0 0.0%	20 13.7%	0 0.0%	0 0.0%	83.3% 16.7%
metocarbamol	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	16 11.0%	0 0.0%	100% 0.0%
mioflex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	15 10.3%	100% 0.0%
	100% 0.0%	100% 0.0%	100% 0.0%	77.8% 22.2%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	97.3% 2.7%
	acetaminofen	acido	aspirina	calmidon	dolex	ibuprofeno	metocarbamol	mioflex	
	Target Class								

Figura 23. Matriz de confusión de entrenamiento para la red CarlosNet2.

Confusion Matrix

Output Class	acetaminofen	5 13.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	acido	0 0.0%	5 13.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	aspirina	0 0.0%	0 0.0%	5 13.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	calmidon	0 0.0%	0 0.0%	0 0.0%	2 5.4%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	dolex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	3 8.1%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	ibuprofeno	0 0.0%	0 0.0%	0 0.0%	3 8.1%	0 0.0%	5 13.5%	0 0.0%	0 0.0%	62.5% 37.5%
	metocarbamol	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	4 10.8%	0 0.0%	100% 0.0%
	mioflex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 2.7%	0 0.0%	0 0.0%	4 10.8%	80.0% 20.0%
		100% 0.0%	100% 0.0%	100% 0.0%	40.0% 60.0%	75.0% 25.0%	100% 0.0%	100% 0.0%	100% 0.0%	89.2% 10.8%
		acetaminofen	acido	aspirina	calmidon	dolex	ibuprofeno	metocarbamol	mioflex	
Target Class										

Figura 24. Matriz de confusión de validación para la red CarlosNet2.

Confusion Matrix

Output Class	acetaminofen	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	NaN% NaN%
	acido	0 0.0%	1 12.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	aspirina	0 0.0%	0 0.0%	1 12.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	calmidon	0 0.0%	0 0.0%	0 0.0%	1 12.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	dolex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 12.5%	0 0.0%	0 0.0%	1 12.5%	50.0% 50.0%
	ibuprofeno	1 12.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 12.5%	0 0.0%	0 0.0%	50.0% 50.0%
	metocarbamol	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 12.5%	0 0.0%	100% 0.0%
	mioflex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	NaN% NaN%
		0.0% 100%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	0.0% 100%	75.0% 25.0%
		acetaminofen	acido	aspirina	calmidon	dolex	ibuprofeno	metocarbamol	mioflex	
Target Class										

Figura 25. Matriz de confusión de prueba para la red CarlosNet2.

Para la segunda simulación la matriz de confusión en entrenamiento se aprecia en la Fi-

gura 26, obteniendo un 95.5% de aciertos, la matriz de confusión en validación se aprecia en la Figura 27, obteniendo un 72.7% de aciertos, la matriz de confusión en prueba se aprecia en la Figura 28, obteniendo un 80% de aciertos lo cual ha fallado en 16 esto se debe a que la red no tiene los suficientes datos de entrenamiento para aprender ocasionando errores en el proceso de clasificación.

Confusion Matrix

Output Class	acetaminofen	12 13.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	acido	0 0.0%	13 14.6%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	aspirina	0 0.0%	0 0.0%	12 13.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	calmidon	0 0.0%	0 0.0%	0 0.0%	11 12.4%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	dolex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	10 11.2%	0 0.0%	2 2.2%	2 2.2%	71.4% 28.6%
	ibuprofeno	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	12 13.5%	0 0.0%	0 0.0%	100% 0.0%
	metocarbamol	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	8 9.0%	0 0.0%	100% 0.0%
	mioflex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	7 7.9%	100% 0.0%
		100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	80.0% 20.0%	77.8% 22.2%	95.5% 4.5%
	acetaminofen	acido	aspirina	calmidon	dolex	ibuprofeno	metocarbamol	mioflex		
	Target Class									

Figura 26. Matriz de confusión de entrenamiento para la red CarlosNet2.

Confusion Matrix

Output Class	acetaminofen	3 13.6%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%	
	acido	0 0.0%	3 13.6%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%	
	aspirina	0 0.0%	0 0.0%	3 13.6%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%	
	calmidon	0 0.0%	0 0.0%	0 0.0%	1 4.5%	0 0.0%	1 4.5%	0 0.0%	0 0.0%	50.0% 50.0%	
	dolex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	2 9.1%	0 0.0%	1 4.5%	2 9.1%	40.0% 60.0%	
	ibuprofeno	0 0.0%	0 0.0%	0 0.0%	2 9.1%	0 0.0%	2 9.1%	0 0.0%	0 0.0%	50.0% 50.0%	
	metocarbamol	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 4.5%	0 0.0%	100% 0.0%	
	mioflex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 4.5%	100% 0.0%	
			100% 0.0%	100% 0.0%	100% 0.0%	33.3% 66.7%	100% 0.0%	66.7% 33.3%	50.0% 50.0%	33.3% 66.7%	72.7% 27.3%
			acetaminofen	acido	aspirina	calmidon	dolex	ibuprofeno	metocarbamol	mioflex	
		Target Class									

Figura 27. Matriz de confusión de validación para la red CarlosNet2.

Confusion Matrix

Output Class	acetaminofen	11 13.8%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%	
	acido	0 0.0%	11 13.8%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%	
	aspirina	0 0.0%	0 0.0%	11 13.8%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%	
	calmidon	0 0.0%	0 0.0%	0 0.0%	8 10.0%	0 0.0%	3 3.8%	0 0.0%	0 0.0%	72.7% 27.3%	
	dolex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	8 10.0%	0 0.0%	6 7.5%	4 5.0%	44.4% 55.6%	
	ibuprofeno	0 0.0%	0 0.0%	0 0.0%	2 2.5%	0 0.0%	8 10.0%	0 0.0%	0 0.0%	80.0% 20.0%	
	metocarbamol	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 1.3%	0 0.0%	3 3.8%	0 0.0%	75.0% 25.0%	
	mioflex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	4 5.0%	100% 0.0%	
			100% 0.0%	100% 0.0%	100% 0.0%	80.0% 20.0%	88.9% 11.1%	72.7% 27.3%	33.3% 66.7%	50.0% 50.0%	80.0% 20.0%
			acetaminofen	acido	aspirina	calmidon	dolex	ibuprofeno	metocarbamol	mioflex	
		Target Class									

Figura 28. Matriz de confusión de prueba para la red CarlosNet2.

En la tabla 8 se aprecia la comparación de los resultados de precisión empleando el conjunto

de datos en validación para las redes **CarlosNet1** y **CarlosNet2**, se observa que en la simulación 1 se obtiene mejores resultados.

Tabla 8

Comparación de los resultados de precisión empleando los datos de validación de las redes CarlosNet1 y CarlosNet2.

Precisión en Validación	Simulación 1	Simulación 2
CarlosNet1	97,3 %	72.7 %
CarlosNet2	89.2 %	72.7 %

Se decide aplicar aprendizaje de transferencia ya que con este podemos obtener mejores porcentajes de precisión según el estado del arte, con el aprendizaje de transferencia se busca reutilizar modelos de redes neuronales de alto rendimiento tales como **ALexNet** y **ResNet18**. Para reutilizar estos modelos se debe cambiar la capa de salida para que tenga el mismo número de clases de nuestro conjunto de datos, también se debe cambiar la capa de la entrada de la imagen para que tenga un tamaño de 240X240 y una profundidad de 29 bandas espectrales y por último se cambia la primera capa convolucional para que concuerde el tamaño de la imagen con esta convolución. Aclarando que se debe conservar las características aprendidas previamente. En la Figura 29 se muestra el esquema del aprendizaje de transferencia.

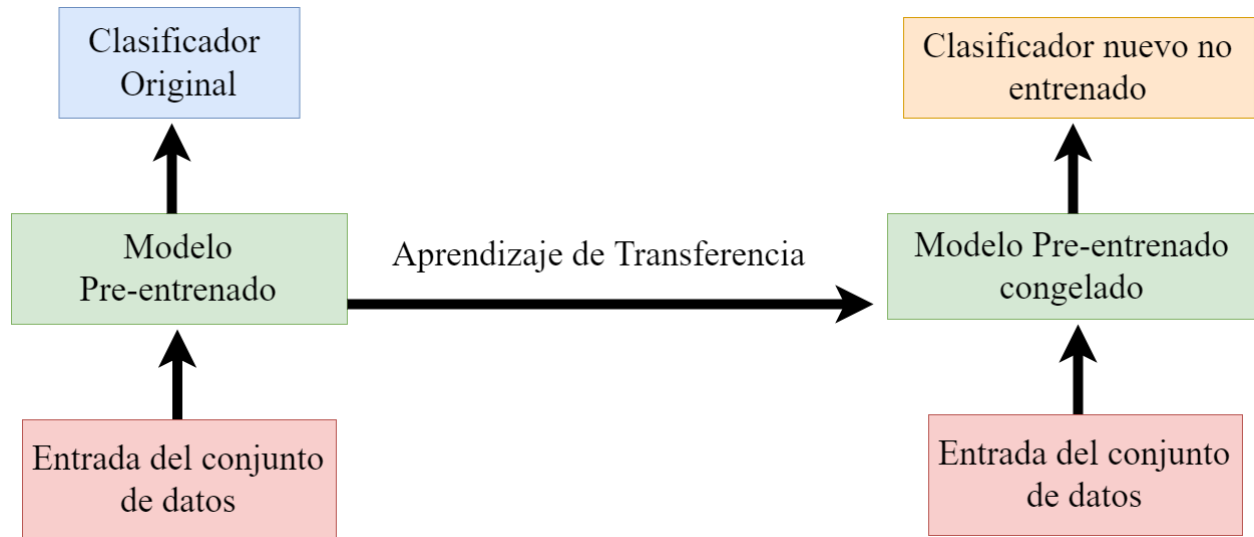


Figura 29. Aprendizaje de transferencia.

Se debe tener en cuenta que al momento de aplicar aprendizaje de transferencia se tienen las mismas condiciones que se les dieron a las redes **CarlosNet1** y **CarlosNet2**. Al aplicar aprendizaje de transferencia al modelo pre-entrenado de **Alexnet** solo se le realiza la primera simulación ya que permite obtener mejores resultados para el proceso de clasificación, los resultados de la matriz de confusión en entrenamiento se aprecian en la Figura 30, obteniendo 93.8% de aciertos, la matriz de confusión en validación se aprecia en la Figura 31, obtiene un 89.2% de aciertos, la matriz de confusión en prueba se aprecia en la Figura 32, obteniendo un 87.5% de aciertos lo cual a fallado en uno con estos resultados se puede decir que esta red tiene un buen rendimiento para clasificar.

Confusion Matrix

Output Class	acetaminofen	20 13.7%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%	
	acido	0 0.0%	21 14.4%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%	
	aspirina	0 0.0%	0 0.0%	20 13.7%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%	
	calmidon	0 0.0%	0 0.0%	0 0.0%	11 7.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%	
	dolex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	16 11.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%	
	ibuprofeno	0 0.0%	0 0.0%	0 0.0%	7 4.8%	0 0.0%	20 13.7%	0 0.0%	0 0.0%	74.1% 25.9%	
	metocarbamol	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	14 9.6%	0 0.0%	100% 0.0%	
	mioflex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	2 1.4%	15 10.3%	88.2% 11.8%	
			100% 0.0%	100% 0.0%	100% 0.0%	61.1% 38.9%	100% 0.0%	100% 0.0%	87.5% 12.5%	100% 0.0%	93.8% 6.2%
			acetaminofen	acido	aspirina	calmidon	dolex	ibuprofeno	metocarbamol	mioflex	
		Target Class									

Figura 30. Matriz de confusión de entrenamiento para la red AlexNet.

Confusion Matrix

Output Class	acetaminofen	5 13.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%	
	acido	0 0.0%	5 13.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%	
	aspirina	0 0.0%	0 0.0%	5 13.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%	
	calmidon	0 0.0%	0 0.0%	0 0.0%	2 5.4%	0 0.0%	1 2.7%	0 0.0%	0 0.0%	66.7% 33.3%	
	dolex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	4 10.8%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%	
	ibuprofeno	0 0.0%	0 0.0%	0 0.0%	3 8.1%	0 0.0%	4 10.8%	0 0.0%	0 0.0%	57.1% 42.9%	
	metocarbamol	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	4 10.8%	0 0.0%	100% 0.0%	
	mioflex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	4 10.8%	100% 0.0%	
			100% 0.0%	100% 0.0%	100% 0.0%	40.0% 60.0%	100% 0.0%	80.0% 20.0%	100% 0.0%	100% 0.0%	89.2% 10.8%
			acetaminofen	acido	aspirina	calmidon	dolex	ibuprofeno	metocarbamol	mioflex	
		Target Class									

Figura 31. Matriz de confusión de validación para la red AlexNet.

Confusion Matrix

Output Class	acetaminofen	acido	aspirina	calmidon	dolex	ibuprofeno	metocarbamol	mioflex	
acetaminofen	1 12.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
acido	0 0.0%	1 12.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
aspirina	0 0.0%	0 0.0%	1 12.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
calmidon	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	NaN% NaN%
dolex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 12.5%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
ibuprofeno	0 0.0%	0 0.0%	0 0.0%	1 12.5%	0 0.0%	1 12.5%	0 0.0%	0 0.0%	50.0% 50.0%
metocarbamol	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 12.5%	0 0.0%	100% 0.0%
mioflex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 12.5%	100% 0.0%
	100% 0.0%	100% 0.0%	100% 0.0%	0.0% 100%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	87.5% 12.5%
	acetaminofen	acido	aspirina	calmidon	dolex	ibuprofeno	metocarbamol	mioflex	

Target Class

Figura 32. Matriz de confusión de prueba para la red AlexNet.

Al aplicar aprendizaje de transferencia al modelo pre-entrenado de **ResNet18** solo se le realiza la primera simulación, los resultados de la de la matriz de confusión en entrenamiento se aprecia en la Figura 33, obteniendo un 100% de aciertos, la matriz de confusión en validación se aprecia en la Figura 34, obteniendo un 94.6% de aciertos, la matriz de confusión en prueba se aprecia en la Figura 35, obteniendo un 87.5% de aciertos lo cual a fallado en uno con estos resultados se puede decir que esta red tiene un buen rendimiento para clasificar.

Confusion Matrix

Output Class	acetaminofen	20 13.7%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	acido	0 0.0%	21 14.4%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	aspirina	0 0.0%	0 0.0%	20 13.7%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	calmidon	0 0.0%	0 0.0%	0 0.0%	18 12.3%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	dolex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	16 11.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	ibuprofeno	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	20 13.7%	0 0.0%	0 0.0%	100% 0.0%
	metocarbamol	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	16 11.0%	0 0.0%	100% 0.0%
	mioflex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	15 10.3%	100% 0.0%
			100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%
			acetaminofen	acido	aspirina	calmidon	dolex	ibuprofeno	metocarbamol	mioflex
		Target Class								

Figura 33. Matriz de confusión de entrenamiento para la red ResNet18.

Confusion Matrix

Output Class	acetaminofen	5 13.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%	
	acido	0 0.0%	5 13.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%	
	aspirina	0 0.0%	0 0.0%	5 13.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%	
	calmidon	0 0.0%	0 0.0%	0 0.0%	4 10.8%	0 0.0%	1 2.7%	0 0.0%	0 0.0%	80.0% 20.0%	
	dolex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	4 10.8%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%	
	ibuprofeno	0 0.0%	0 0.0%	0 0.0%	1 2.7%	0 0.0%	4 10.8%	0 0.0%	0 0.0%	80.0% 20.0%	
	metocarbamol	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	4 10.8%	0 0.0%	100% 0.0%	
	mioflex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	4 10.8%	100% 0.0%	
			100% 0.0%	100% 0.0%	100% 0.0%	80.0% 20.0%	100% 0.0%	80.0% 20.0%	100% 0.0%	100% 0.0%	94.6% 5.4%
			acetaminofen	acido	aspirina	calmidon	dolex	ibuprofeno	metocarbamol	mioflex	
		Target Class									

Figura 34. Matriz de confusión de validación para la red ResNet18.

Confusion Matrix

Output Class	acetaminofen	acido	aspirina	calmidon	dolex	ibuprofeno	metocarbamol	mioflex	
acetaminofen	1 12.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
acido	0 0.0%	1 12.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
aspirina	0 0.0%	0 0.0%	1 12.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
calmidon	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	NaN% NaN%
dolex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 12.5%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
ibuprofeno	0 0.0%	0 0.0%	0 0.0%	1 12.5%	0 0.0%	1 12.5%	0 0.0%	0 0.0%	50.0% 50.0%
metocarbamol	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 12.5%	0 0.0%	100% 0.0%
mioflex	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 12.5%	100% 0.0%
	100% 0.0%	100% 0.0%	100% 0.0%	0.0% 100%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	87.5% 12.5%
	acetaminofen	acido	aspirina	calmidon	dolex	ibuprofeno	metocarbamol	mioflex	
	Target Class								

Figura 35. Matriz de confusión de prueba para la red ResNet18.

En la tabla 9 se muestran los resultados se que obtuvieron con las arquitecturas propuestas y con las arquitecturas de **Alexnet** y **ResNet18**, se presentan los resultados de la simulación 1 para las redes **CarlosNet1**, **CarlosNet2** y **SpectralNet1**.

Tabla 9
Resultados de precisión de validación.

	Modelo 2D				Modelo 1D
	CarlosNet1	CarlosNet2	ResNet18	AlexNet	SpectralNet1
Precisión en Validación	97,3%	89.2%	94.6%	89.2%	58.9%

Con respecto a los resultados anteriores el que obtuvo menor rendimiento en los modelos 2D fue **AlexNet** debido a que el diseño de esta arquitectura fue pensado para un conjunto de datos

bastante numeroso porque tiene capas totalmente conectadas de 2048 neuronas aun así se obtienen un buen resultado, el modelo propuesto de **CarlosNet1** se obtiene el mayor porcentaje de precisión en validación, el éxito del modelo propuesto de la red **CarlosNet1** fue normalizar por lotes en cada capa convolucional y usar el mismo tamaño de los filtros.

5. Conclusiones

Se demostró que fue posible construir una arquitectura óptica del grupo de investigación HDSP de la Universidad Industrial de Santander de Colombia para la adquisición de la base de datos de imágenes espectrales empleando la técnica de adquisición escaneo espectral.

En este proyecto se demostró que fue posible crear un modelo de aprendizaje de máquina a partir de redes neuronales convolucionales para la clasificación de objetos empleando imágenes espectrales.

Con los modelos de aprendizaje de máquina 2D propuestos fue posible clasificar medicamentos que tenía la misma visualización en RGB entre ellos están acetaminofen, calmidon, mioflex, dolex y metocarbamol, obteniendo un porcentaje de precisión en validación hasta el 97.3% con la red **CarlosNet1**.

Referencias Bibliográficas

Ali, L. (2017). Recognition. pp. 1–5.

Alvarez-Ramos, C. M., Nino, E., & Santos, M. (2013). Automatic classification of *Nosema* pathogenic agents through machine vision techniques and kernel-based vector machines. *2013 8th Computing Colombian Conference, 8CCC 2013*.

An, Q. A. Q., Gao, W. G. W., Yu, L. Y. L., Wu, J. W. J., Yang, B. Y. B., & Liu, W. L. W. (2010). Research on crop identification method based on two-phase classification using remote sensing in large scale. *World Automation Congress (WAC), 2010*, pp. 7–11.

Aytaylan, H. & Yuksel, S. E. (2016). Semantic segmentation of hyperspectral images with the fusion of lidar data. En *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pp. 2522–2525.

Azad, M. & Moshkov, M. (2014). Minimization of decision tree depth for multi-label decision tables. *Proceedings - 2014 IEEE International Conference on Granular Computing, GrC 2014*, pp. 7–12.

Banerjee, S., Choudhary, A., & Pal, S. (2016). Empirical evaluation of K-Means, Bisecting K-Means, Fuzzy C-Means and Genetic K-Means clustering algorithms. *2015 IEEE International WIE Conference on Electrical and Computer Engineering, WIECON-ECE 2015*, pp. 168–172.

Ben Fadhel, Y., Ktata, S., & Kraiem, T. (2016a). Cardiac scintigraphic images segmentation tech-

- niques. En *2016 2nd International Conference on Advanced Technologies for Signal and Image Processing (ATSIP)*, pp. 364–369.
- Ben Fadhel, Y., Ktata, S., & Kraiem, T. (2016b). Cardiac scintigraphic images segmentation techniques. *2nd International Conference on Advanced Technologies for Signal and Image Processing, ATSIP 2016*, pp. 364–369.
- Bi, Y., Qi, F., Guo, J., & Zheng, R. (2011). The spectral image acquisition and processing system based on folded gratings spectrograph and iccd. En *Proceedings of 2011 International Conference on Electronic Mechanical Engineering and Information Technology*, volumen 3, pp. 1332–1335.
- Chegini, M. & Ghassemian, H. (2011). Spatial spectral gaussian mixture model approach for automatic segmentation of multispectral mr brain images. En *2011 19th Iranian Conference on Electrical Engineering*, pp. 1–1.
- Dhinagar, N. J. & Celenk, M. (2014). Analysis of regularity in skin pigmentation and vascularity by an optimized feature space for early cancer classification. En *2014 7th International Conference on Biomedical Engineering and Informatics*, pp. 709–713.
- Feng, J., Chen, J., Liu, L., Cao, X., Zhang, X., Jiao, L., & Yu, T. (2019). CNN-Based Multilayer Spatial-Spectral Feature Fusion and Sample Augmentation with Local and Nonlocal Constraints for Hyperspectral Image Classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(4):1299–1313.

- Fesehaye, D., Singaraveluy, L., Chen, C. C., Huang, X., Banerjee, A., Zhou, R., & Somasundaran, R. (2017). Group Clustering Using Inter-Group Dissimilarities. *Proceedings - International Conference on Distributed Computing Systems*, pp. 1011–1021.
- García-Esteban, J. A., Curto, B., Moreno, V., González-Martín, I., Revilla, I., & Vivar-Quintana, A. (2018). A digitalization strategy for quality control in food industry based on artificial intelligence techniques. En *2018 IEEE 16th International Conference on Industrial Informatics (INDIN)*, pp. 221–226.
- Garg, A., Gupta, D., Saxena, S., & Sahadev, P. P. (2019). Validation of random dataset using an efficient cnn model trained on mnist handwritten dataset. En *2019 6th International Conference on Signal Processing and Integrated Networks (SPIN)*, pp. 602–606.
- Girouard, G. & Bannari, a. (2004). Validated spectral angle mapper algorithm for geological mapping: comparative study between QuickBird and Landsat-TM. *XXth ISPRS Congress, ...*, (January 2014):599–604.
- Gonzalez, T. F. (2007). Handbook of approximation algorithms and metaheuristics. *Handbook of Approximation Algorithms and Metaheuristics*, pp. 1–1432.
- Gupta, A., Gusain, K., & Popli, B. (2016). Verifying the value and veracity of extreme gradient boosted decision trees on a variety of datasets. *11th International Conference on Industrial and Information Systems, ICIIS 2016 - Conference Proceedings*, 2018-January:457–462.
- Hasan, K., Aanstoos, J. V., & Mahrooghy, M. (2013). Stressed vegetation identification by SAR

time series as an indicator of slope instability in Mississippi River levee segments. *Proceedings - Applied Imagery Pattern Recognition Workshop*, pp. 1–4.

Haykin, S. (1994). *Neural Networks: A Comprehensive Foundation*. Prentice Hall PTR, USA, 1st edición.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016-December:770–778.

Hinton, G. E., Osindero, S., & Teh, Y.-W. (2006). A fast learning algorithm for deep belief nets. *Neural Computation*, 18(7):1527–1554. PMID: 16764513.

Hussein, S., Kandel, P., Bolan, C. W., Wallace, M. B., & Bagci, U. (2019). Lung and Pancreatic Tumor Characterization in the Deep Learning Era: Novel Supervised and Unsupervised Learning Approaches. *IEEE Transactions on Medical Imaging*, 38(8):1777–1787.

Idrissi, M. A. J., Ramchoun, H., Ghanou, Y., & Ettaouil, M. (2016). Genetic algorithm for neural network architecture optimization. *Proceedings of the 3rd IEEE International Conference on Logistics Operations Management, GOL 2016*, pp. 1–4.

Jr, R. D. M., Rosa, R. A. S., Bayer, F. M., Pettersson, M. I., & Machado, R. (2019). A CHANGE DETECTION ALGORITHM FOR SAR IMAGES BASED ON LOGISTIC REGRESSION
Aeronautics Institute of Technology – ITA , São José dos Campos - SP , 12228-900 , Brazil .

Visiona Tecnologia Espacial S . A ., São José dos Campos - SP , 12247-016 , Brazil . Feder. pp. 1514–1517.

Karvelis, P. S., Tzallas, A. T., Fotiadis, D. I., & Georgiou, I. (2008). A multichannel watershed-based segmentation method for multispectral chromosome classification. *IEEE Transactions on Medical Imaging*, 27(5):697–708.

Kulkarni, P. & Stranieri, A. (2018). Comparison of Pixel N-Grams with Histogram, Haralick's features and Bag-of-Visual-Words for Texture Image Classification. *2018 3rd International Conference for Convergence in Technology, I2CT 2018*, pp. 1–5.

Le, W., Lin, M., Jia, L., Ai, J., Fu, X., & Chen, Z. (2019). Multi-Objective Optimization of an Air-Cored Axial Flux Permanent Magnet Synchronous Machine with Segmented PMs based on Support Vector Machine and Genetic Algorithm. *2019 22nd International Conference on Electrical Machines and Systems, ICEMS 2019*, pp. 1–4.

Li, C. & Gu, X. (2010). FEM analysis of the concrete structure life under the freezing and thawing cycles conditions. *ICIC 2010 - 3rd International Conference on Information and Computing*, 2:54–57.

Lu, G. & Fei, B. (2014). Medical hyperspectral imaging: a review. *Journal of Biomedical Optics*, 19(1):1 – 24.

Ma, J., Wei, M., Xu, L., Chen, B., Liu, Y., Du, J., & Fang, Z. (2017). Ultrasonic spectral analy-

- sis for biomedical imaging. En *2017 IEEE International Conference on Imaging Systems and Techniques (IST)*, pp. 1–3.
- Makhfi, N. E., Bannay, O. E., Benslimane, R., & Rais, N. (2011). System of indexing, annotation and search in the old Arabic manuscripts. *2011 Colloquium in Information Science and Technology, CIST 2011*, p. 9.
- Mao, X., Zhao, G., & Sun, R. (2018). Naive Bayesian algorithm classification model with local attribute weighted based on KNN. *Proceedings of the 2017 IEEE 2nd Information Technology, Networking, Electronic and Automation Control Conference, ITNEC 2017*, 2018-January:904–908.
- Mazumdar, M., Sarasvathi, V., & Kumar, A. (2018). Object recognition in videos by sequential frame extraction using convolutional neural networks and fully connected neural networks. *2017 International Conference on Energy, Communication, Data Analytics and Soft Computing, ICECDS 2017*, pp. 1485–1488.
- Miranda, J. D. & Salinas, S. A. (2019). Computational measuring approach for the identification of probable intestinal system pathologies through the human iris parameters. En *2019 XXII Symposium on Image, Signal Processing and Artificial Vision (STSIVA)*, pp. 1–5.
- Mondal, I. & Chatterjee, S. (2019). Secure and Hassle-Free EVM Through Deep Learning Based Face Recognition. *Proceedings of the International Conference on Machine Learning, Big Data,*

Cloud and Parallel Computing: Trends, Perspectives and Prospects, COMITCon 2019, pp. 109–113.

Moustakis, V. & Stankovic, S. (2005). Text Informative. 00:209–212.

Mulyani, E. D. S. & Susanto, J. P. (2017). Classification of maturity level of fuji apple fruit with fuzzy logic method. *2017 5th International Conference on Cyber and IT Service Management, CITSM 2017*.

Nirkin, Y., Masi, I., Tuán, A. T., Hassner, T., & Medioni, G. (2018). On face segmentation, face swapping, and face perception. *Proceedings - 13th IEEE International Conference on Automatic Face and Gesture Recognition, FG 2018*, pp. 98–105.

Peichl, M., Albers, T., & Dill, S. (2015). Detection of small impurities in bulk material by mmw radar. En *2015 16th International Radar Symposium (IRS)*, pp. 294–299.

Phothisonothai, M., Tantisatirapong, S., & Aurasopon, A. (2016). Automated determination of watermelon ripeness based on image color segmentation and rind texture analysis. En *2016 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS)*, pp. 1–5.

Radford, A., Metz, L., & Chintala, S. (2015). Unsupervised representation learning with deep convolutional generative adversarial networks.

Samat, A., Du, P., Liu, S., Li, J., & Cheng, L. (2014). E²lms : Ensemble extreme learning ma-

- chines for hyperspectral image classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 7(4):1060–1069.
- Seixas, J. L., Barbon, S., & Mantovani, R. G. (2015). Pattern recognition of lower member skin ulcers in medical images with machine learning algorithms. En *2015 IEEE 28th International Symposium on Computer-Based Medical Systems*, pp. 50–53.
- Shahriar Sazzad, T. M., Tanzibul Ahmmed, K. M., Hoque, M. U., & Rahman, M. (2019). Development of automated brain tumor identification using mri images. En *2019 International Conference on Electrical, Computer and Communication Engineering (ECCE)*, pp. 1–4.
- Sharma, A., Singh, P. K., & Khurana, P. (2016). Analytical review on object segmentation and recognition. *Proceedings of the 2016 6th International Conference - Cloud System and Big Data Engineering, Confluence 2016*, pp. 524–530.
- Signoroni, A., Savardi, M., Pezzoni, M., Guerrini, F., Arrigoni, S., & Turra, G. (2018). Combining the use of cnn classification and strength-driven compression for the robust identification of bacterial species on hyperspectral culture plate images. *IET Computer Vision*, 12(7):941–949.
- Singh, N., Das, S., & Veeramuthu, A. (2018). An efficient combined approach for medical brain tumour segmentation. *Proceedings of the 2017 IEEE International Conference on Communication and Signal Processing, ICCSP 2017*, 2018-January:1325–1329.
- Veremyev, V. I., Vorobev, E. N., & Kokorina, Y. V. (2019). Feasibility study of air target detection by passive radar using satellite-based transmitters. *Proceedings of the 2019 IEEE Conference*

of Russian Young Researchers in Electrical and Electronic Engineering, ElConRus 2019, pp. 154–157.

Wang, S. L. & Liew, A. W. (2006). Information-based color feature representation for image classification. *Proceedings - International Conference on Image Processing, ICIP*, 6:353–356.

Zeng, H., Zhao, J., Short, M. A., McLean, D. I., Lam, S., McGregor, H. C., Kalia, S., McWilliams, A., Wang, W., & Lui, H. (2014). Real-time in Vivo tissue Raman spectroscopy for early cancer detection. *Optics InfoBase Conference Papers*, pp. 1–3.

Zeng, Z., Yao, Y., & Zhuang, Y. (2015). A Wideband Common-Mode Suppression Filter with Compact-Defected Ground Structure Pattern. *IEEE Transactions on Electromagnetic Compatibility*, 57(5):1277–1280.

Zhang, J., Li, L., Zhu, G., Meng, X., & Xie, Q. (2017a). A comparison study of semi-supervised SVM algorithms for small business credit prediction. *IEEE/ACM BESC 2016 - Proceedings of 2016 International Conference on Behavioral, Economic, Socio - Cultural Computing*, 2(1).

Zhang, X., Deng, J., & Su, R. (2017b). The EM algorithm for a linear regression model with application to a diabetes data. *PIC 2016 - Proceedings of the 2016 IEEE International Conference on Progress in Informatics and Computing*, (1):114–118.

Zhao, W., Jiao, L., Ma, W., Zhao, J., Zhao, J., Liu, H., Cao, X., & Yang, S. (2017). Superpixel-based multiple local cnn for panchromatic and multispectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 55(7):4141–4156.

Zheng, Y. & Xiao, S. (2016). Performance analysis of a moving target tracking method based on computer vision. *Proceedings - 2016 8th International Conference on Measuring Technology and Mechatronics Automation, ICMTMA 2016*, pp. 467–470.