Instituto Tecnológico de Costa Rica
Escuela de Ingeniería en Electrónica

Detección de regiones de café en imágenes aéreas de plantaciones costarricenses

para optar por el título de:
Máster en Procesamiento Digital de Señales

Juan Carlos Cruz Naranjo

San José, 24 de septiembre de 2022

Declaro que el presente informe de trabajo final de graduación ha sido realizado por mi persona, utilizando y aplicando literatura referente al tema e introduciendo conocimientos propios.

En los casos en que he utilizado bibliografía, he procedido a indicar las fuentes mediante las respectivas citas bibliográficas. En consecuencia, asumo la responsabilidad total por el trabajo realizado y por el contenido del correspondiente informe final.

Juan Carlos Cruz Naranjo
San José, Costa Rica
24 de septiembre de 2022
Céd: 304810138

Instituto Tecnológico de Costa Rica
Escuela de Ingeniería Electrónica
Maestría Académica en Electrónica
Trabajo Final de Graduación
Tribunal Evaluador
Acta de Aprobación de Tesis

Defensa del Trabajo Final de Graduación
Requisito para optar por el título de Máster en Ingeniería Electrónica
Grado Académico de Magister Scientiae

El Tribunal Evaluador aprueba la defensa del Trabajo Final de Graduación denominado **"Detección de regiones de café en imágenes aéreas de plantaciones costarricenses"**, realizado por **Juan Carlos Cruz Naranjo** Carné: **201217650**, y hace constar que cumple con las normas establecidas por la Unidad Interna de Posgrados de la Escuela de Ingeniería Electrónica del Instituto Tecnológico de Costa Rica.

Miembros del Tribunal Evaluador

Dra. Laura Cristina Cabrera Quirós
Profesora Lectora

M.Sc Michael Gruner Monzón
Profesor Lector

M.Sc. Luis Alonso Murillo Rojas
Evaluador Independiente

Dr. José Pablo Alvarado Moya
Director del Tesis

San José, 24 de septiembre de 2022

Resumen

El Instituto del Café de Costa Rica (ICAFÉ) se encuentra trabajando en un programa que tiene por objetivo proveer al sector cafetalero nacional con semilla certificada. Al ser un proceso manual, el consumo de recursos que este programa demanda actualmente representa una oportunidad de mejora mediante técnicas de aprendizaje profundo en imágenes aéreas, como parte del desarrollo de un proyecto innovador en dos etapas: la detección de cultivos de café y su respectiva clasificación varietal.

En este trabajo se propone un algoritmo de aprendizaje profundo para la identificación de las regiones correspondientes a plantaciones de café en imágenes aéreas. La metodología definida para dicha identificación corresponde a la detección de anomalías. Para esto se ha diseñado un autocodificador convolucional, como la arquitectura de red neuronal seleccionada para clasificar regiones como plantaciones de café o como anomalías. El proceso de entrenamiento del autocodificador se desarrolló utilizando exclusivamente sub-imágenes cuyo contenido corresponde a plantas de café, bajo un esquema generativo de aprendizaje no supervisado.

La experimentación se basó principalmente en modificaciones al tamaño del espacio latente del autocodificador, utilizando 128, 256 y 512 dimensiones para generar tres modelos diferentes y evaluar su capacidad de reconstrucción de las sub-imágenes, así como la separación de los datos entre regiones que corresponden a plantaciones de café y las regiones consideradas como anomalías. Para la clasificación de dichas regiones, se utilizó una Máquina de Soporte Vectorial (SVM) a la salida de los codificadores de los tres modelos.

El entrenamiento de los tres modelos convergió hacia valores de pérdida inferiores al 19 % y hacia valores de exactitud superiores al 81 %. Se generaron gráficos de dispersión de baja dimensionalidad, donde fue posible observar un traslape inferior al 50 % entre los conjuntos de datos del espacio latente correspondientes a café y anomalías, que en su mayoría representan secciones de otras plantas. También se realizó un análisis estadístico del error de reconstrucción con los tres modelos de autocodificador, donde se determinó que el menor traslape se obtiene cuando dicho error es calculado sobre la sub-imagen RGB original concatenada horizontalmente con su versión filtrada utilizando el algoritmo LBP.

Como resultados cualitativos, se reconstruyeron cuatro mosaicos completos que representan escenarios agrícolas a partir de las sub-imágenes clasificadas, donde se identificó que la precisión de los resultados depende de factores como la iluminación, las sombras y la altura a la que se capturaron dichos mosaicos. Los resultados sugieren la necesidad de mejorar la precisión de la clasificación de regiones, donde se deberán explorar métodos supervisados y utilizar imágenes con más información que permitan mejorar la separación entre las sub-imágenes de café y las sub-imágenes de otras plantas visualmente similares.

Palabras Clave: imágenes aéreas, aprendizaje profundo, autocodificador, espacio latente, detección de anomalías, error de reconstrucción.

Abstract

The Costa Rican Coffee Institute (ICAFÉ) is currently working on a program that aims to provide the national coffee sector with certified seed. Being a manual process, the consumption of resources that this program currently demands represents an opportunity for improvement, through deep learning techniques in aerial images as part of the development of an innovative project in two stages: the detection of coffee crops and their respective varietal classification.

In this work, a deep learning algorithm is proposed for the identification of regions corresponding to coffee plantations in aerial images. The methodology defined for such identification corresponds to the detection of anomalies. For this purpose, a convolutional autoencoder has been designed as the neural network architecture selected to classify regions as coffee plantations or as anomalies. The training process of the autoencoder was developed using exclusively sub-images whose content corresponds to coffee plants, under a generative unsupervised learning scheme.

The experimentation was mainly based on modifications to the size of the latent space of the autoencoder, using 128, 256 and 512 dimensions to generate three different models and evaluate its capacity to reconstruct the sub-images, as well as the separation of the data between regions corresponding to coffee plantations and regions considered as anomalies. For the classification of these regions, a Support Vector Machine (SVM) was used at the output of the encoders of the three models.

The training processes of the three models converged to loss values below 19 % and accuracy values above 81 %. Low dimensionality scatter plots were generated, where it was possible to observe less than 50 % overlap between the latent space data sets corresponding to coffee and anomalies, most of which represent sections of other plants. A statistical analysis of the reconstruction error was also performed with the three autoencoder models, where it was determined that the lowest overlap is obtained when such error is calculated on the original RGB sub-image horizontally concatenated with its filtered version using the LBP algorithm.

As qualitative results, four complete mosaics representing agricultural scenarios were reconstructed from the classified sub-images, where it was identified that the accuracy of the results depends on factors such as illumination, shadows and the height at which the mosaics were captured. The results suggest the need to improve the accuracy of region classification, where supervised methods should be explored and images with more information should be used to improve the separation between sub-images of coffee and sub-images of other visually similar plants.

Keywords: aerial imagery, deep learning, autoencoder, latent space, anomaly detection, reconstruction error.

A mis amados padres

Agradecimientos

Este trabajo ha sido posible gracias al esfuerzo de mis padres, pilares fundamentales en mi vida, quienes me brindaron su apoyo incondicional durante todo el proceso de mi formación como profesional.

Agradezco a mi abuela, quien partió de este mundo en 2015 y quien me impulsó desde niño con amor a luchar por mis sueños con valentía, fe y esperanza a pesar de los momentos difíciles y situaciones complicadas.

Un agradecimiento especial a la empresa *RidgeRun Embedded Solutions* por financiar gran parte del aprendizaje adquirido para llevar a cabo el desarrollo de este proyecto.

Índice General

Índice de Tablas

Índice de Figuras

1 | Introducción

1.1. Reseña histórica del café en Costa Rica

De acuerdo con la reseña histórica descrita en [1], el cultivo del café se introdujo a Costa Rica a finales del siglo XVIII, siendo las primeras semillas de la especie *Coffea Arábica* y la variedad *Typica*. En 1808, el cultivo del café empezó a crecer en Costa Rica, donde los primeros cafetales fueron cultivados en los suelos fértiles de origen volcánico del Valle Central. El clima costarricense presenta una temporada lluviosa y una temporada seca, con temperaturas uniformes durante todo el año que favorecen el desarrollo correcto del cultivo del café. Después de la independencia de España en 1821, los gobiernos municipales del país crearon políticas de entrega de plantas y concesión de tierras a los interesados en cultivar café, de manera que su siembra se volvió intensiva en los años siguientes.

Para dicho año, Costa Rica contaba con 17 mil cafetales en producción. La primera exportación estuvo dada por 2 quintales de café a Panamá en 1820. Para la década de 1940, mientras se incentivaba aún más la siembra del café, se inició con la construcción de un camino al Atlántico costarricense que representaría una ruta directa hacia los puertos del Reino Unido. En 1846 se finalizó la construcción del camino a Puntarenas, incrementando el comercio del café al sustituir las mulas por carretas para su transporte. En estos años, el café se convirtió en el único producto de exportación hasta 1890, considerándose hasta la actualidad como un pivote en la economía nacional de Costa Rica.

1.2. Retos para la agricultura en Costa Rica

El café es un cultivo procedente de Etiopía, cuya producción y consumo han incrementado desde que se popularizó como bebida durante el siglo XV. A partir de este punto, el café se convirtió en un pilar del comercio internacional, donde los productores más importantes están en América Latina y Asia, mientras que sus principales consumidores están en Estados Unidos y la Unión Europea. El 80 % de la producción de café está en manos de pequeños productores, los cuales se enfrentan a una dramática tendencia a la baja de los precios del mercado. Además, la variabilidad climática propició la aparición de enfermedades como la

roya causada por el hongo *Hemileia Vastatrix*, que ataca a la planta de café y produce una disminución en la calidad y cantidad de la producción.

Estos son factores que afectan a más de 120 millones de personas, impactando negativamente a la economía de países enteros. Estos factores se vuelven críticos cuando en dichos países el café es uno de los principales productos de exportación, como sucede con Costa Rica. La alta demanda de café está dada por los aumentos anuales en el consumo de la bebida, donde se estima que cada segundo se consumen 255 kilogramos de café en el mundo. En los mercados donde se manejan los mayores precios, los consumidores solicitan más información sobre el origen de los granos y las condiciones de sostenibilidad de su producción. Para cumplir con ese objetivo, se han establecido más de 69 estándares distintos, algunos generales y otros específicos para el café. Estos estándares pretenden respaldar la sostenibilidad de la producción del café, siendo actualmente un requisito de acceso al mercado [2].

En la década de los 80 del siglo XX, se promovió en Costa Rica una diversificación agrícola con el objetivo de ampliar la oferta exportable y de reducir la dependencia económica de los productos tradicionales, dentro de los cuales destacan el café y el banano. Con el pasar de los años, los retos de la economía global, el cambio climático y el establecimiento de nuevos tratados comerciales con más países, han provocado la necesidad de adaptarse e integrar nuevas prácticas que permitan promover en el sector agrícola costarricense la exportación de productos con mayor valor agregado. Ante este escenario, es clave impulsar el desarrollo de programas de modernización de empresas y reconversión productiva, incluyendo inteligencia comercial, investigación de mercados, gestión empresarial, así como la investigación y transferencia de tecnología, considerados como factores fundamentales para que el sector agropecuario de Costa Rica sea competitivo [3].

Como componentes adicionales, se considera necesario promover el acceso a la información y su análisis a los productores, buscando la mejora en sus estándares de calidad, su planificación y el uso efectivo de los recursos con mejores fundamentos para la toma de decisiones.

1.3. Vehículos aéreos no tripulados en agricultura

Para abordar los desafíos del sector agrícola costarricense, ha sido necesario mejorar la comprensión de los ecosistemas productivos complejos, multivariados e impredecibles mediante el monitoreo, la medición y el análisis continuo de fenómenos físicos. El entendimiento de los ecosistemas agrícolas implica el análisis de datos y el uso de nuevas tecnologías de información y comunicación, incluyendo la observación de cultivos a pequeña y gran escala para buscar mejoras en las tareas existentes de gestión y toma de decisiones. La observación de ecosistemas a gran escala se ve facilitada por la detección remota, la cual permite obtener imágenes de los entornos agrícolas mediante satélites, aviones y vehículos aéreos no tripulados (UAV, por sus siglas en inglés). El uso de UAV representa un método no destructivo para la recopilación de información sobre las características del terreno y los cultivos, permitiendo

obtener los datos de forma sistemática en grandes áreas geográficas [4].

Las imágenes aéreas capturadas a través de los UAV se han convertido en una parte integral de la agricultura moderna. La resolución espacial proporcionada por los sensores de los UAV ha revolucionado el flujo de trabajo en el campo, ya que aportan la capacidad de representar información gráfica sobre las características del ecosistema agrícola. Esto permite utilizar dichas imágenes en tareas como la medición del estado de los cultivos y su rendimiento durante los distintos períodos de crecimiento, la identificación y monitoreo de malezas y enfermedades [5], el estudio de la cobertura de los cultivos en determinadas regiones geográficas, el estudio y análisis de suelo [6], entre otras.

Como potencial uso, la extensión del cultivo del café en Costa Rica superaba las 94 mil hectáreas en 2021 y se estima que ha incrementado durante el 2022 [7]. Esto trae consigo una serie de retos para el país en términos económicos, comerciales, logísticos y ambientales, donde las imágenes aéreas capturadas de forma remota se utilizan como herramienta para el análisis de los ecosistemas agrícolas cafetaleros y para la toma de decisiones acorde con los resultados de dicho análisis, contribuyendo con satisfacer las necesidades del sector cafetalero en el país [8].

1.4. Aprendizaje profundo en agricultura de precisión

Como herramienta para la comprensión de los ecosistemas agrícolas ha surgido la agricultura inteligente, un concepto que aborda los desafíos de la producción agrícola en términos de eficiencia, impacto ambiental, seguridad alimentaria y sostenibilidad. El uso de agricultura inteligente involucra el análisis de imágenes y videos para extraer información de los ambientes agrícolas, permitiendo tomar decisiones que contribuyen a la reducción del uso innecesario de extensiones de tierra para el cultivo, la producción de aguas azules y el uso excesivo de fertilizantes [9].

El enfoque predominante en la agricultura inteligente ha sido el aprendizaje profundo (DL), definido por un conjunto de técnicas del aprendizaje automático [10]. El aprendizaje profundo consiste en la utilización de redes neuronales artificiales (ANN, por sus siglas en inglés) más profundas que las redes neuronales artificiales tradicionales y que proporcionan una representación jerárquica de los datos a través de diferentes niveles de abstracción, permitendo mayor capacidad de aprendizaje para mejorar el rendimiento y la precisión en los resultados [4]. Sin embargo, el aprendizaje profundo demanda mayor cantidad de datos a partir de los cuales aprender, mayor capacidad de procesamiento y arquitecturas más complejas que las requeridas por otros enfoques de aprendizaje automático.

El desarrollo de las técnicas de aprendizaje profundo utilizadas como parte de la agricultura inteligente ha potenciado el valor económico y ecológico de la agricultura de precisión, ya que proporciona un enfoque eficaz para facilitar el manejo inteligente y la toma de decisiones en

tareas como la categorización visual de cultivos [11], la detección de enfermedades en plantas en tiempo real [12, 13], el reconocimiento de plagas, el conteo y la recolección automática de frutos [14], el monitoreo de la calidad y salud de los cultivos [15].

1.4.1. Técnicas de aprendizaje profundo en imágenes agrícolas

En la tabla 1.1 se presenta un conjunto general de modelos de aprendizaje profundo empleados en la agricultura para tareas asociadas al reconocimiento de rasgos fenotípicos de los cultivos en imágenes. Estas tareas se clasifican en dos categorías dentro del dominio del aprendizaje automático: aprendizaje supervisado utilizando etiquetas o valores objetivo para cada muestra, y aprendizaje no supervisado extrayendo representaciones significativas para obtener características clave de los datos sin utilizar etiquetas [16].

Tabla 1.1: Modelos basados en aprendizaje profundo para el análisis de imágenes agrícolas [17]

Aprendizaje	Modelo	Tarea	Referencias
Supervisado	Redes convolucionales	Reconocimiento de objetos	[18, 19]
		Clasificación de escenarios	[20, 21]
		Geolocalización	[22]
No supervisado	Autocodificador	Extracción de características	[20]
		Detección de anomalías	[23]

Una tarea del aprendizaje profundo corresponde a la detección de anomalías en imágenes y videos. En este ámbito, la detección de anomalías consiste en la identificación de patrones en los datos que no se ajustan al comportamiento esperado. Estos patrones son denominados anomalías, valores atípicos, observaciones discordantes, excepciones, peculiaridades o contaminantes, de acuerdo al dominio de aplicación [24]. La detección de anomalías en imágenes mediante técnicas de aprendizaje profundo representa una oportunidad para evitar el proceso de generar datos etiquetados como normales y anómalos. La generación de un set de imágenes etiquetadas de forma manual es un proceso exhaustivo que demanda más recursos e incluso la necesidad de contratar criterio experto, aumentando el costo de la generación del set a tal punto de ser inviable.

La detección de anomalías en imágenes agrícolas requiere del reconocimiento de características físicas observables para que el algoritmo utilizado pueda aprender de ellas. Las técnicas de aprendizaje automático y aprendizaje profundo para la detección de anomalías, se han utilizado en imágenes aéreas para identificar el cultivo o los cultivos con algún tipo de pato-

logía presente en regiones determinadas, donde se definen como anomalías todas las regiones que no corresponden al cultivo de interés o todas las regiones que presentan cultivos enfermos [25–27]. La ubicación de estas regiones en las imágenes también brinda la posibilidad de obtener una clasificación varietal de los cultivos. Esta información representa una potencial herramienta para mejorar la toma de decisiones sobre las plantaciones con miras a incrementar su calidad y eficiencia [4].

1.4.2. Clasificación varietal en plantaciones de café

El Instituto del Café de Costa Rica (ICAFÉ) se encuentra trabajando en un programa de semilla, el cual consiste en proveer al sector cafetalero nacional con semilla certificada. El desarrollo de este programa implica un trabajo intensivo de campo que permite la selección de los lotes con la mayor homogeneidad varietal y con la menor cantidad de plantas consideradas como fuera de tipo. Este proceso de selección permite la obtención de semilla de café con alta pureza genética. Actualmente dicho trabajo se realiza de forma manual, recorriendo cada hilera y observando cada planta para determinar si corresponde o no a un individuo de interés dentro de un lote específico.

Debido a la complejidad de este proceso y al consumo de recursos que éste demanda, surge la necesidad de contar con un sistema para la identificación automática de variedades de café presentes en imágenes aéreas de plantaciones costarricenses. Se pretende que este sistema contribuya con la aceleración del trabajo de descarte de lotes que posean alto porcentaje de plantas fuera de tipo, evitando el consumo innecesario de recursos en lotes no aptos para una semilla determinada. Este esfuerzo del ICAFÉ permitirá proveer al sector cafetalero con una tecnología para colocar a su disposición semilla de alta pureza genética, de una forma más eficiente que en la actualidad. Para satisfacer los requerimientos del programa, el sistema debe contar con dos capacidades principales: el reconocimiento de regiones correspondientes a cafetales en las imágenes aéreas y la clasificación de las variedades de café presentes en las regiones reconocidas.

La pregunta de investigación de este trabajo corresponde a ¿qué tan preciso es un sistema de aprendizaje profundo para el reconocimiento de regiones correspondientes a cafetales en imágenes aéreas, cuando el entrenamiento de la red neuronal se desarrolla utilizando únicamente sub-imágenes que contienen secciones de plantas de café, sin necesidad de etiquetarlas a nivel de pixel?

1.5. Objetivos y estructura del trabajo

Este trabajo tiene como objetivo proponer un algoritmo para la identificación de regiones correspondientes a plantas de café en imágenes aéreas de plantaciones costarricenses, mediante

técnicas de aprendizaje profundo para la detección de anomalías. Con la implementación de estas técnicas se pretende evitar el etiquetado de las imágenes a nivel de pixeles, de manera que el proceso de aprendizaje se desarrolle sobre imágenes cuyo contenido está dado únicamente por partes de plantas de café. Dentro de este contexto, las secciones de las imágenes que no corresponden a plantas de café, son consideradas como anomalías.

Para lograrlo, se propone la construcción de un set de datos compuesto por sub-imágenes obtenidas a partir de un conjunto de imágenes aéreas, las cuales representan mosaicos que contienen información sobre los ecosistemas cafetaleros de interés.

También se plantea la exploración de un conjunto de propuestas del estado de arte, que utilizan técnicas de aprendizaje automático y aprendizaje profundo en imágenes para aplicaciones agrícolas. Con esta exploración, se pretenden definir los componentes de las arquitecturas base de red neuronal. El objetivo de contar con arquitecturas base, corresponde a establecer un conjunto de redes neuronales a partir de las cuales se puedan generar resultados experimentales, aplicar cambios y evaluarlos cualitativa y cuantitativamente.

Por último, se pretende generar una versión etiquetada de las imágenes aéreas a partir los mosaicos originales, donde sea posible visualizar las regiones clasificadas como plantaciones de café y las regiones consideradas como anomalías.

Las principales contribuciones de este trabajo, están dadas por un set de datos depurado que contiene sub-imágenes de secciones de plantas de café a ser utilizadas en procesos de entrenamiento y pruebas, además de un conjunto de sub-imágenes consideradas como anomalías para utilizarse en pruebas. Otra contribución principal, consiste en una primera versión de un sistema capaz de clasificar regiones como café o como anomalías, representando una línea base para identificar oportunidades de mejora en la precisión de los resultados.

En el capítulo 2 de este documento, se presenta un conjunto de propuestas asociadas al uso de técnicas de aprendizaje profundo en imágenes para diferentes aplicaciones agrícolas. También se presenta una serie de conceptos relacionados con arquitecturas de redes neuronales que representan técnicas de aprendizaje no supervisado en imágenes. En el capítulo 3, se presenta una descripción del sistema propuesto para identificar regiones correspondientes a plantaciones de café en imágenes aéreas. En el capítulo 4, se describen los experimentos realizados con el sistema propuesto, junto con sus respectivos resultados y análisis. Por último, en el capítulo 5 se presentan las conclusiones extraídas a partir de los resultados obtenidos.

2 | Marco Teórico

Los sistemas de aprendizaje automático y aprendizaje profundo tienen como base un conjunto de algoritmos de reconocimiento de patrones, que son seleccionados como posibles soluciones a problemas dentro de una aplicación determinada. Un sistema típico de reconocimiento de patrones se muestra en la figura 2.1, el cual presenta un sensor con el que se capturan representaciones de elementos del mundo real en forma de datos, un mecanismo de preprocesamiento de datos, un mecanismo de extracción de características manual o automático, un algoritmo de clasificación o descripción, y un conjunto de datos de entrenamiento ya clasificados o descritos [28].

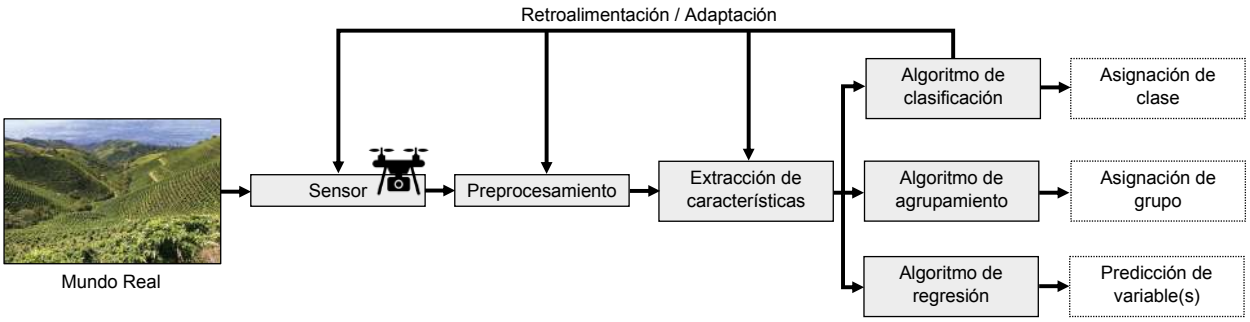


Figura 2.1: Sistema típico de reconocimiento de patrones (adaptado de [28]).

El flujo de operación representado en la figura 2.1, también es utilizado como referencia en el diseño de aplicaciones que emplean imágenes aéreas para el análisis de ecosistemas agrícolas y la toma de decisiones a partir de los resultados. En este capítulo, se presenta un conjunto de propuestas del estado del arte que proveen posibles soluciones a problemas en ecosistemas agrícolas cafetaleros, utilizando técnicas de aprendizaje automático y aprendizaje profundo. También se presenta una serie de conceptos asociados al flujo representado en la figura 2.1, que fue tomado como base para definir la solución propuesta en este trabajo.

2.1. Trabajos Relacionados

Las propuestas que se encuentran actualmente en la literatura, relacionadas al trabajo con imágenes aéreas para el análisis de ecosistemas agrícolas cafetaleros, son en su mayoría de la última década. Si se considera la utilización de técnicas de aprendizaje profundo para el análisis de dichas imágenes, es posible determinar que ésta es una área de investigación reciente y en crecimiento.

Parte de los esfuerzos en cuanto al cultivo del café están dados por el análisis de enfermedades. En [29], Marcos Alexandre et al. proponen el uso de una CNN para aprender a identificar la enfermedad de la roya en el café. La roya corresponde a una enfermedad causada por el hongo patógeno *Hemileia vastatrix* que ataca la parte inferior de las hojas de café y se caracteriza por la presencia de puntos amarillos con polvo. De no ser tratada a tiempo, esta enfermedad puede causar la caída en la producción de café de hasta el 45 %. La red propuesta fue evaluada mediante la comparación entre un conjunto de 159 mosaicos segmentados por un experto y la identificación obtenida a través del sistema de aprendizaje, el cual fue entrenado con 2859 sub-imágenes extraídas a partir de los mosaicos originales. Los resultados de dicha comparación muestran que el enfoque propuesto es capaz de detectar la infección con alta precisión, evidenciado por un índice de *Dice* de 0.82 que se obtuvo a partir de imágenes erosionadas.

Otro ejemplo de análisis de enfermedades en café a partir de imágenes se presenta en [30], donde los autores proponen una metodología para detectar la presencia de nematodos en cultivos de café. En este estudio se utilizaron UAV para obtener imágenes RGB de alta resolución a partir de una plantación de café comercial. La metodología permite la extracción de características visuales de regiones en las imágenes, y utiliza técnicas de aprendizaje automático supervisado para clasificar las áreas en dos clases: con plaga y sin plaga. Se emplearon técnicas de aprendizaje utilizando enfoques con y sin segmentación. Los resultados demostraron el potencial de la metodología propuesta, con un promedio para la medida $F1$ del 63 % que se obtuvo mediante el uso de una red neuronal convolucional U-Net empleando segmentación manual.

Otros esfuerzos están dirigidos al análisis geográfico en extensiones del cultivo de café. En [31], Rafael Baeta et al. proponen una estrategia basada en pixeles para el mapeo geográfico de los cultivos de café mediante la combinación de dos tendencias del reconocimiento de patrones para aplicaciones de teledetección: aprendizaje profundo y fusión/selección de características de múltiples escalas. La propuesta consiste en el entrenamiento y la combinación de redes neuronales convolucionales (CNN, por sus siglas en inglés), diseñadas para recibir como entrada ventanas de contexto alrededor de pixeles etiquetados. Los mapas finales se crean combinando la salida de esas redes para un conjunto de pixeles no etiquetados. Los resultados experimentales demostraron que las escalas múltiples producen mejores mapas de cultivos de café que el uso de escalas individuales.

La obtención de información para el análisis geográfico de plantaciones y las características físicas de los cultivos a partir de imágenes, requiere de tareas como la detección y localización de objetos y estructuras, la segmentación semántica, la clasificación de imágenes o de regiones de interés, entre otras. Los resultados de cada una de estas tareas dependen del método de captura de imágenes utilizado, el cual debe ser seleccionado de acuerdo al tipo de problema a solventar. Es por esta razón que en la actualidad se cuenta con sensores que capturan imágenes en diferentes regiones del espectro, como es el caso de imágenes RGB [32], térmicas [33], fluorescentes [34], tridimensionales [35], estéreo [36] y multiespectrales [37]. Para el caso de la captura de imágenes aéreas con UAV, la altura de vuelo se convierte en otro factor determinante para la cantidad de información que puede extrarse de los ecosistemas agrícolas de interés.

Una propuesta relacionada al análisis geográfico del café se presenta en [38], donde el autor propone una estrategia de seguimiento para plantaciones de café a través de ortomosaicos RGB y multiespectrales capturados mediante UAV. Se realiza la detección, conteo y estudio de características vitales de los cafetos en las plantaciones analizadas. Para la detección se utilizaron redes neuronales artificiales con el modelo YOLO [39]. Mientras que el estudio de características vitales de los cafetos fue establecida a través de ecuaciones matemáticas, estimadas en base a datos de campo brindados por el Instituto del Café de Costa Rica (ICAFÉ), y ortomosaicos con índices de vegetación de diferencia normalizada (NDVI). Como resultado, se obtuvo que la mejor detección de cafetos en imágenes ocurrió con una resolución espacial de 1cm/pixel y 4cm/pixel. Además, se obtuvo una detección de verdaderos positivos equivalente al 90.2 % de los cafetos, a un índice de intersección sobre unión (IoU) de 0.35 en imágenes con una resolución espacial de 4cm/pixel, obteniendo un mAP50 de 75.5 % mediante una arquitectura YOLOv3. Sin embargo, los resultados de los modelos matemáticos que relacionan el índice NDVI en los ortomosaicos y las características vitales de las plantas de café no presentaron resultados concluyentes.

El costo de adquisición de UAV se ha reducido en los últimos años, provocando que su uso para fines investigativos supere al uso de imágenes satelitales. Sin embargo, también se han realizado estudios para el análisis geográfico de plantaciones de café empleando este tipo de imágenes. Por ejemplo, en [40] los autores presentan un estudio donde utilizaron un método de clasificación basado en pixeles y un método de clasificación de análisis de imágenes basado en objetos (OBIA, por sus siglas en inglés), con el objetivo de detectar café a partir de imágenes satelitales *WorldView 2* en la región cafetera de Kona en Hawai. El enfoque OBIA produjo mejores resultados en la detección de campos de café, con una precisión general del 81 % en comparación con el 68 % para el clasificador de pixeles. Además, la clasificación con OBIA no requiere de procesamiento adicional para generar polígonos de campos de café esperados, como lo requiere el método basado en pixeles.

Otro ejemplo de análisis geográfico se presenta en [41], donde Edemir Ferreira et al. desarrollaron una investigación sobre la aplicación de los enfoques actuales de adaptación de dominio no supervisado (UDA, por sus siglas en inglés) [42], en la tarea de transferir conocimiento entre regiones de cultivo que tienen diferentes patrones de café. Dada una región geográfica

con cafetales totalmente mapeados, se observó que este conocimiento puede ser empleado para entrenar un clasificador y mapear una nueva región sin necesidad de muestras provenientes de la región objetivo. Los resultados experimentales mostraron que la transferencia de conocimiento a través de UDA funciona mejor que la aplicación directa de un clasificador entrenado en una región determinada para predecir los cultivos de café presentes en una nueva región. Sin embargo, los autores advirtieron que los métodos UDA pueden conducir a una transferencia negativa, indicando que los dominios son sumamente diferentes. Con esto se concluyó que las estrategias de transferencia pueden ser ineficaces dependiendo del escenario de exploración.

En [43], los autores evaluaron la capacidad de generalización de las CNN como descriptores en dos escenarios distintos: detección aérea y detección remota en la clasificación de imágenes. Se utilizaron dos conjuntos de datos, de los cuales uno corresponde a una composición de escenas capturadas por el sensor SPOT en 2005 durante vuelos realizados en cuatro condados del estado de Minas Gerais, Brasil: Arceburgo, Guaranésia, Guaxupé y Monte Santo, donde las imágenes son de áreas con y sin cultivos de café en alta resolución. Este conjunto de datos presenta ciertos desafíos dados por las variaciones intraclase causadas por diferentes técnicas de manejo de los cultivos. Además, el sur de Minas Gerais es una región montañosa con plantas de diferentes edades y con distorsiones y sombras espectrales. Se utilizaron imágenes de 64×64 pixeles, considerando únicamente las bandas verde, roja e infrarroja, tomadas como las más representativas en la discriminación del área de vegetación. Las CNN obtuvieron los mejores resultados para imágenes aéreas, mientras que en la detección remota fueron superadas por descriptores de color de bajo nivel como el BIC [44].

Parte de las propuestas para el análisis geográfico también incluye el uso de autocodificadores en imágenes aéreas. En [45] los autores presentan un método denominado autocodificador siamés totalmente convolucional (FCNN, por sus siglas en inglés) para la detección de cambios en imágenes aéreas, en particular para las imágenes obtenidas mediante UAV. Este tipo de autocodificadores aprende una función de distancia para comparar los mapas de características e indicar la presencia de cambios. En este trabajo se demostró que es posible reducir la cantidad de muestras etiquetadas necesarias para lograr resultados competitivos mediante el uso de un autocodificador. Se realizaron evaluaciones del rendimiento con dos conjuntos de datos diferentes, demostrando que la metodología propuesta supera a dos métodos más del estado del arte, requiriendo a su vez menos datos de entrenamiento.

También se han presentado propuestas para el análisis de otros rasgos físicos en las plantas de café, como es el caso de los granos. En [46], los autores presentan una comparación de cuatro características establecidas para detectar los frutos rojos (maduros) en las plantas de café mediante imágenes RGB de alta resolución, capturadas con un UAV que viajó a través de las hileras de la plantación. La metodología propuesta permite la extracción de características visuales de las regiones de la imagen y utiliza técnicas supervisadas de aprendizaje automático para clasificar áreas como frutos de café y no frutos (como ramas y hojas). Se compararon varios métodos de aprendizaje utilizando los datos de prueba obtenidos en una plantación de café: kNN [47], bosques aleatorios de decisión [48] y redes neuronales. Los resultados

experimentales demostraron que el modelo de red neuronal es más confiable que los demás métodos utilizados para identificar con precisión los frutos de café.

Los trabajos explorados proveen información sobre arquitecturas de aprendizaje profundo y resultados a ser considerados para extraer características de las imágenes aéreas de plantaciones de café, las cuales permiten identificar cuáles regiones pertenecen a cultivos de café y cuales no. Esto permite establecer una base para la identificación y clasificación de variedades de café a partir de imágenes aéreas.

A continuación, se presenta un conjunto de conceptos relacionados con arquitecturas de autocodificadores, empleados para la detección de anomalías en imágenes. Además, se desarrollan conceptos asociados a métricas utilizadas en procesos de entrenamiento, pruebas y clasificación, así como conceptos relacionados con algoritmos clasificadores de datos. Por último, se presentan conceptos sobre técnicas de preprocesamiento de imágenes y algoritmos de reducción de dimensiones en los datos para facilitar su visualización y análisis.

2.2. Arquitecturas de red neuronal

En el ámbito de detección de anomalías en imágenes, las redes neuronales convolucionales (CNN) y las redes generativas adversarias (GAN) representan enfoques de referencia para extraer información a partir de los datos, formando parte de arquitecturas más complejas para lograr dicha detección de anomalías, como es el caso de los autocodificadores.

2.2.1. Redes neuronales convolucionales (CNN)

Las redes neuronales convolucionales (CNN o ConvNet) corresponden a un tipo de red neuronal profunda para el procesamiento de datos, las cuales son empleadas principalmente en tareas de reconocimiento y clasificación en imágenes. Bajo este escenario, la red recibe como entrada una imagen y produce como salida la clase que mejor describe a la imagen de entrada o la probabilidad de que dicha imagen pertenezca a una clase determinada. Las CNN poseen una topología similar a un retículo, con estructuras 2D para el caso en que los datos de entrada sean imágenes [49]. El nombre de red neuronal convolucional indica que la red emplea la operación matemática lineal denominada convolución en lugar de la multiplicación matricial general en al menos una de sus capas. Esto permite que las capas de la red tengan la capacidad de extraer patrones característicos a partir de filtros convolutivos [50].

La convolución es una operación lineal sobre dos funciones de dominio real o imaginario [50]. Si la convolución se realiza entre dos señales de naturaleza bidimensional, definidas a lo largo de dos dimensiones perpendiculares entre sí, la operación se denomina convolución 2D. Las imágenes digitales representan un tipo de señales bidimensionales discretas, donde la

convolución se realiza sobre la matriz de una imagen mediante una matriz 2D de dimensión menor conocida como el *kernel*. La convolución 2D se define de la siguiente manera:

$$(f * g)(x, y) = \sum_m \sum_n f(m, n) g(x - m, y - n), \tag{2.1}$$

donde $f(m, n)$ corresponde a la imagen de entrada y $g(x, y)$ al *kernel*.

En la figura 2.2 se puede identificar un bloque convolucional compuesto por una primera capa básica, donde se aplica la convolución para extraer características de la imagen. Dicho bloque también posee una segunda capa de agrupación de valores promedio (*pooling*), empleada para la reducción de dimensiones. De forma general, el bloque convolucional se compone de un conjunto apilado de ambas capas: convolucional y de agrupación.

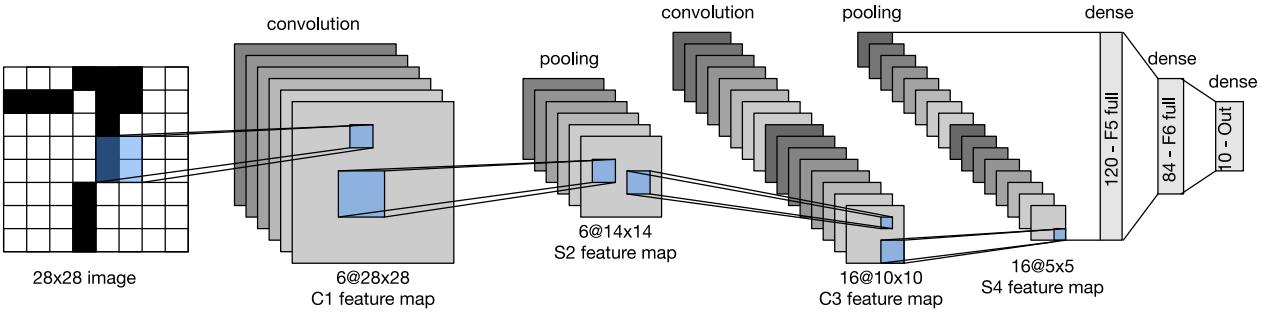


Figura 2.2: Ejemplo de red neuronal convolucional (adoptado de [51]).

En este ejemplo de aplicación, cada capa convolucional utiliza un núcleo de 5×5 y procesa cada salida con una función de activación sigmoidal. En este caso, la primera capa convolucional presenta 6 canales de salida y la segunda capa convolucional aumenta aún más la profundidad del canal a 16. Posteriormente, se emplea la función de activación *ReLU* para anular los valores negativos y preservar los positivos tal y como ingresan a la red [52].

2.2.2. Redes generativas adversarias (GAN)

Las redes generativas adversarias (GAN) corresponden a otro tipo de redes profundas, utilizadas en la resolución de problemas no supervisados y semi-supervisados. Esta estructura propone la estimación de modelos generativos a través de un proceso de confrontación, donde se lleva a cabo el entrenamiento de dos modelos de forma simultánea. El primero de ellos corresponde a un modelo generativo G, que captura la distribución de los datos tomando ruido aleatorio como entrada y el segundo está dado por un modelo discriminativo D, cuyo propósito es la estimación de la probabilidad de que una muestra provenga de los datos de entrenamiento o de la distribución de G [53].

El procedimiento de entrenamiento para G consiste en maximizar la probabilidad de que D cometa un error. En el espacio de funciones arbitrarias G y D existe una solución única,

con G recuperando la distribución de datos de entrenamiento y D igual a una constante con valor de 1/2. Si G y D están definidas por perceptrones multicapa, todo el sistema se puede entrenar mediante retropropagación [54].

Durante el proceso de entrenamiento el generador aprende a producir muestras cada vez más realistas, mientras que el discriminador aprende a mejorar y distinguir de forma más precisa entre los datos generados y los datos reales [55]. El funcionamiento general de una GAN se presenta en la figura 2.3.

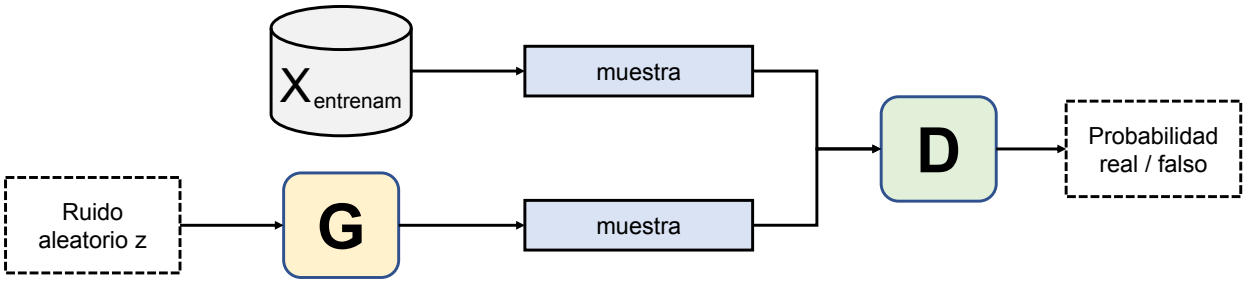


Figura 2.3: Esquema general de una GAN (adaptado de [55]).

Para conocer la distribución de salida del generador sobre los puntos de datos $\mathbf{x}$, se define una variable de ruido de entrada anterior $p_z(\mathbf{z})$. Cada asignación al espacio de datos se puede representar como $G(\mathbf{z};\theta_g)$, donde G corresponde a una red neuronal profunda generativa con parámetros θ_g. Se define el discriminador $D(\mathbf{x};\theta_d)$ que genera $D(\mathbf{x}) \in [0,1]$, lo cual representa la probabilidad de que $\mathbf{x}$ se haya tomado del conjunto de entrenamiento en lugar del generador G.

El discriminador D es entrenado para maximizar la probabilidad de asignar etiquetas correctamente para datos de entrenamiento reales y para muestras falsas de G. El entrenamiento simultáneo permite que G minimice $1 - D(G(\mathbf{z}))$ en un problema de optimización descrito por la siguiente relación:

$$\min_G \left\{ \max_G \left\{ \mathbb{E}_{\mathbf{x}\sim p_{data}(\mathbf{x})}[\log D(\mathbf{x})] + \mathbb{E}_{\mathbf{z}\sim p_{\mathbf{z}}(\mathbf{z})}[\log(1 - D(G(\mathbf{z})))] \right\} \right\} \tag{2.2}$$

Inicialmente, los gradientes de D se calculan para discriminar muestras falsas en los datos de entrenamiento. Posteriormente, G se actualiza para generar muestras con mayor probabilidad de clasificarse como datos. Después de varios pasos de entrenamiento, si G y D tienen suficiente capacidad y se logra un equilibrio de *Nash* [56], alcanzarán un punto en el que ambos no podrán mejorar más.

2.2.3. El autocodificador

Un autocodificador corresponde a una red neuronal entrenada mediante aprendizaje no supervisado, el cual se emplea para aprender sobre reconstrucciones cercanas a su entrada original. Un autocodificador está estructurado de manera tal que, al recibir una entrada, la transforma en una representación simplificada para intentar reconstruir la entrada original a partir de dicha representación [57]. Un autocodificador está compuesto por dos subredes: un codificador h que genera la representación simplificada a partir de la entrada, y un decodificador z que intenta reconstruir la entrada original a partir de la representación generada por el codificador. En su forma más simple, un autocodificador de una capa se describe como

$$h(x) = \omega(W_{xh}x + b_{xh}) \tag{2.3}$$

$$z(h) = \omega(W_{hx}h + b_{hx}), \tag{2.4}$$

donde W representa los pesos y b el sesgo (*bias*) de la red neuronal. En este caso, ω representa una función de transformación no lineal. En la figura 2.4 se presenta el esquema general de un autocodificador.

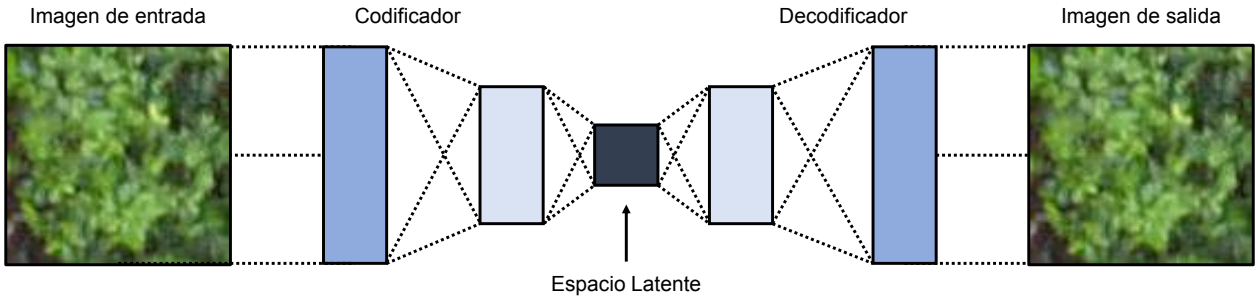


Figura 2.4: Arquitectura general de un autocodificador.

El codificador denotado en (2.3) asigna un vector de entrada x a una representación oculta h mediante un mapeo no lineal. A esta representación se le conoce como representación de espacio latente. El decodificador denotado por (2.4), mapea la representación oculta h de regreso al espacio de entrada original como una reconstrucción mediante la transformación inversa del codificador. La diferencia entre el vector de entrada original x y la reconstrucción z se denomina error de reconstrucción e:

$$e = \|x - z\| . \tag{2.5}$$

Un autocodificador aprende a minimizar e. La detección de anomalías basada en autocodificador es un método basado en desviaciones que emplean aprendizaje semi-supervisado, donde el error e es considerado como el puntaje de anomalía. Al entrenarse con los datos

normales, se espera que el autocodificador produzca un error de reconstrucción mayor para las entradas anormales que para las normales, lo cual se adopta como criterio para identificar dichas anomalías [58]. Al trabajar con imágenes, se considera conveniente implementar las arquitecturas del codificador y el decodificador mediante redes neuronales convolucionales.

2.2.3.1. El autocodificador variacional

Un autocodificador variacional (VAE) corresponde a un modelo generativo, donde la salida del codificador está dada por dos vectores: un vector de valores medios μ y un vector de desviaciones estándar σ, obtenidos a partir de un conjunto de datos aleatorios x_i. Estos datos son muestreados para obtener un vector que representa la entrada del decodificador. El vector μ se encarga de determinar el espacio en el cual se centra la codificación para una entrada, mientras que el vector σ delimita el área de variación de la codificación, garantizando que los datos codificados puedan generarse a partir de una distribución determinada. El decodificador aprende que los puntos cercanos a un punto en específico del espacio latente pertenecen a la misma clase, permitiendo que los datos con poca variación entre sí puedan ser identificados como pertenecientes a dicha clase [59]. En la figura 2.5 se muestra el esquema general de un VAE, donde se observan los vectores de medias μ y desviaciones estándar σ que componen el espacio latente z.

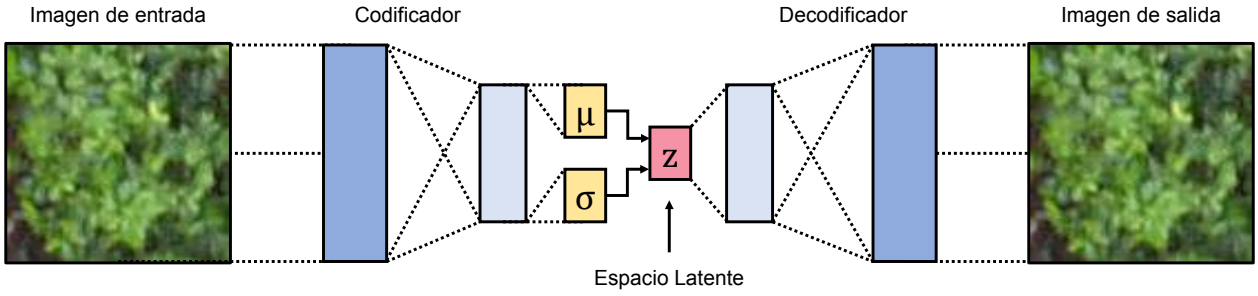


Figura 2.5: Arquitectura general de un autocodificador variacional.

El principio de funcionamiento de un VAE parte de la probabilidad condicional del Teorema de Bayes:

$$p(z|x) = \frac{p(x|z)p(z)}{p(x)}, \tag{2.6}$$

donde $p(z)$ representa la probabilidad a-priori y $p(z|x)$ la probabilidad a-posteriori, siendo z la variable aleatoria que genera a la observación x. Debido a la complejidad para obtener $p(x)$, el VAE aplica inferencia variacional para estimar su valor. Este proceso consiste en aproximar $p(z|x)$ mediante una distribución $q(z|x)$. Para lograrlo, se debe minimizar la distancia entre ambas distribuciones de probabilidad:

$$\text{mín}\left\{D_{KL}\left[q(z|x)||p(z|x)\right]\right\}, \tag{2.7}$$

donde D_{KL} corresponde a la divergencia de *Kullback-Leibler*, empleada para determinar dicha distancia. Para minimizar (2.7), se requiere obtener:

$$\text{máx}\left\{E_{q(z|x)} \log p(x|z) - D_{KL}\left(q(z|x)||p(z)\right)\right\}, \tag{2.8}$$

donde el primer término de la diferencia representa la probabilidad de reconstrucción y el segundo término la distribución $q(z|x)$, cercana a la verdadera distribución a priori $p(z)$ [60].

Un autocodificador variacional se construye como una red neuronal donde el codificador aprende a obtener una representación del espacio latente z a partir de x, y el decodificador aprende a obtener una reconstrucción $\hat{x}$ a partir de la información de z.

En un VAE, la función de pérdida está dada por la suma de dos términos:

$$\mathcal{L}(x, \hat{x}) + \sum_j D_{KL}\left[q_j(z|x)||p(z)\right], \tag{2.9}$$

donde el primer término penaliza el error de reconstrucción y el segundo término sugiere la similitud requerida entre las distribuciones $q(z|x)$ y $p(z)$, para cada dimensión j del espacio latente.

2.2.3.2. El autocodificador adversario

Un autocodificador adversario (AAE) corresponde a un modelo generativo que opera sobre un espacio latente continuo, al igual que un VAE. Sin embargo, en este caso se utiliza una distribución a priori para controlar la salida del codificador y se incluye una red adversaria (GAN). La salida del codificador sigue estando compuesto por valores medios y desviaciones estándar, pero ahora se emplea la distribución a priori para modelar el espacio latente. Para lograrlo, la salida del codificador se distribuye uniformemente sobre una distribución conocida, la cual puede estar dada por una distribución normal, Gamma, F, exponencial u otra distribución probabilística de variable continua.

Bajo este esquema, el codificador aprende a convertir la distribución de datos a la distribución a-priori, mientras que el decodificador aprende un modelo generativo profundo que mapea de la distribución a-priori a la distribución de los datos.

La función de codificación del autocodificador, denotada como $q(z|x)$, define una distribución agregada a-posteriori $q(z)$ sobre el espacio latente que se marginaliza como

$$q(z) = \int_x q(z|x)p_d(x)dx, \tag{2.10}$$

donde x es la entrada, z el espacio latente, $p(z)$ la distribución a-priori, $q(z|x)$ la distribución del codificador y $p(x|z)$ corresponde a la distribución del decodificador. La distribución de los datos se denota como $p_d(x)$ y la distribución del modelo como $p(x)$.

Un AAE es un autocodificador regularizado por el mapeo de $q(z)$ a $p(z)$ mediante una red adversaria, mientras que se procura reducir el error de reconstrucción a través de un autocodificador básico. El esquema general de un autocodificador adversario se presenta en la figura 2.6, donde el generador de la red adversaria está dado por el codificador $q(z|x)$ del autocodificador. La red adversaria y el autocodificador son entrenados en conjunto con el descenso del gradiente estocástico (SGD) en dos fases: la reconstrucción y la regularización [61].

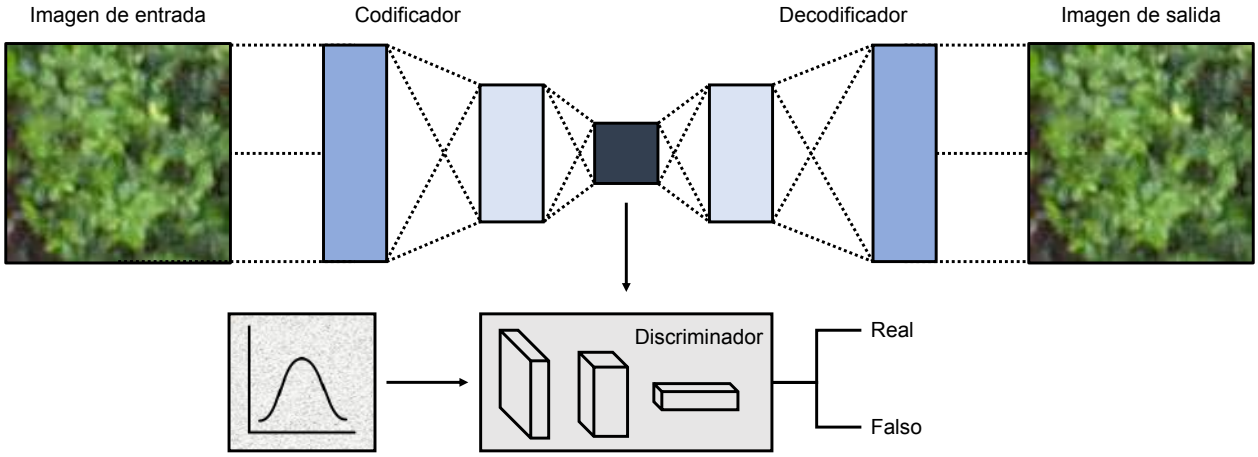


Figura 2.6: Arquitectura general de un autocodificador adversario.

En la fase de reconstrucción, el autocodificador actualiza el codificador y el decodificador para minimizar el error de reconstrucción. En la fase de regularización, la red adversaria primero actualiza la red del discriminador para diferenciar entre muestras verdaderas generadas con la distribución a-priori y muestras generadas con la información del espacio latente. Posteriormente, la red adversaria actualiza su generador (el codificador) para confundir al discriminador. Una vez finalizado el proceso de entrenamiento, el decodificador define un modelo generativo capaz de mapear $p(z)$ a $q(z)$.

2.3. Clasificadores

Los algoritmos de clasificación representan un proceso conformado por tres fases principales: la fase de entrenamiento, la fase de validación y la fase de prueba. Para cada fase, se define un conjunto de datos determinado. En la primera fase, el modelo del clasificador se entrena utilizando patrones de entrada denominados datos de entrenamiento, donde se ajustan los parámetros de dicho modelo. El error de entrenamiento es utilizado como una medida de qué tan bien se ajusta el modelo entrenado a los datos de entrenamiento [62]. En la fase de validación, el conjunto de datos de validación proporciona una evaluación no sesgada del modelo entrenado mientras se ajustan los hiperparámetros de dicho modelo. En la fase de prueba, el conjunto de datos se usa para evaluar el modelo con los hiperparámetros ajustados.

Existen dos tipos de problemas de clasificación de acuerdo al número de clases de los datos: la clasificación binaria en la que solamente se consideran dos clases, y la clasificación multiclase cuando se deben tomar en cuenta más de dos clases [62].

2.3.1. Máquina de Soporte Vectorial (SVM)

SVM (*Support Vector Machine*) corresponde a un algoritmo de aprendizaje automático supervisado, que se emplea como posible solución a problemas de clasificación o regresión. En tareas de clasificación, un SVM recibe como entrada dos o más clases de datos etiquetados y actúa como un clasificador discriminativo. Bajo este escenario, el objetivo de un SVM corresponde a maximizar una función matemática particular con respecto a un determinado conjunto de datos [63]. El proceso de clasificación utilizando SVM se basa en cuatro conceptos: el hiperplano de separación, el hiperplano de margen máximo, el margen suave y la función del kernel.

Un hiperplano $\mathbb{H}$ es un subespacio lineal de un espacio vectorial $\mathbb{V}$, tal que la base de $\mathbb{H}$ tiene cardinalidad uno menos que la cardinalidad de la base de $\mathbb{V}$. Esto implica que si $\mathbb{V} \in \mathfrak{R}^n$, entonces $\mathbb{H} \in \mathfrak{R}^{n-1}$. Por lo tanto, el hiperplano de separación puede considerarse como un plano que divide los datos en un espacio multidimensional [64]. Un hiperplano de separación representa un conjunto de elementos $v \in \mathbb{V}$ que satisfacen la relación

$$w^T v = k, \tag{2.11}$$

donde w es un vector de $\mathbb{V}$ perpendicular al hiperplano y que determina su orientación, y donde k es un término de intersección utilizado para seleccionar el hiperplano. La distancia entre el hiperplano y el origen de $\mathbb{V}$ se define como

$$d = k/\|w\| \,. \tag{2.12}$$

Para establecer el hiperplano de margen máximo, se define la distancia entre el hiperplano de separación y los puntos del conjunto de entrenamiento más cercanos como el margen del hiperplano. El algoritmo SVM procura maximizar dicha distancia, donde los puntos más cercanos corresponden a los vectores de soporte. Durante el proceso de maximización de las distancias, se establecen los vectores de soporte de forma dinámica a partir de los puntos del conjunto de entrenamiento, como parte del proceso de ajuste de hiperparámetros [63].

En la práctica, la gran mayoría de casos de clasificación demandan un método de separación de clases más flexible que un hiperplano lineal. Ante estos escenarios, el SVM permite que algunos datos sean ubicados del lado equivocado del hiperplano de separación. Esta capacidad se alcanza mediante modificaciones al SVM, donde se agrega un margen suave que permite ubicar algunos puntos de datos dentro del margen del hiperplano de separación, sin afectar significativamente el resultado final de la clasificación. La introducción del margen suave implica el uso de un parámetro especificado por el usuario, el cual ayuda a determinar

cuántos puntos de datos pueden irrespetar al hiperplano de separación y hasta dónde se les permite atravesar los límites. El ajuste de este parámetro representa un reto para el usuario porque aún se necesita maximizar el margen con respecto a los puntos de datos clasificados correctamente [63].

Existen dos formas de especificar la flexibilidad que provee el margen suave por parte del usuario. La primera forma es utilizando un modelo C-SVM, donde $C \in [0, \infty[$, lo que dificulta su especificación. El valor óptimo del parámetro C depende de los requerimientos de la aplicación y es necesaria una evaluación exploratoria empírica. La segunda forma corresponde a emplear un modelo ν-SVM, donde $\nu \in]0, 1[$, simplificando la búsqueda del parámetro ν óptimo y aportando un mayor grado de independencia de la aplicación de interés.

Por otro lado, la función del kernel corresponde a una función que tiene por objetivo proyectar los datos desde un espacio de baja dimensión a un espacio de alta dimensión. La selección de la función del kernel adecuada, facilita la separación de los datos en el nuevo espacio de alta dimensión [63]. Una de las funciones del kernel más utilizadas corresponde a la función de base radial (RBF). Esta función toma dos puntos x_1 y x_2, para calcular su semejanza K mediante la relación:

$$\mathrm{K}(x_1, x_2) = \exp\left(\gamma \|x_1 - x_2\|^2\right), \tag{2.13}$$

donde $\|x_1 - x_2\|$ representa la distancia euclidiana entre ambos puntos y γ corresponde a un parámetro que determina qué tan rápido decrece la métrica de similitud entre los dos puntos.

2.4. Reducción de dimensionalidad en datos

En ocasiones, el análisis de los datos empleados en las diferentes etapas de los autocodificadores y otras arquitecturas complejas, demanda la visualización de la distribución de los datos en espacios de máximo tres dimensiones. Sin embargo, generalmente dichos datos presentan una dimensionalidad muy superior, siendo necesario utilizar algoritmos como t-SNE o UMAP que son capaces de reducir la cantidad de dimensiones de los datos para una visualización apropiada.

2.4.1. UMAP

UMAP (*Uniform Manifold Approximation and Projection*) corresponde a otro algoritmo utilizado para la reducción de dimensiones en un conjunto de datos, facilitando su visualización y análisis. Representa una herramienta similar a t-SNE [65], que también se puede usar para la reducción general de dimensiones no lineales. UMAP utiliza un marco teórico basado en la geometría de Riemann y la topología algebraica [66], considerando tres supuestos sobre los datos:

- Los datos están uniformemente distribuidos en el espacio de Riemann.
- La métrica de Riemann es un valor constante localmente o puede aproximarse como tal.
- El conjunto de datos se ubica sobre una variedad topológica que está localmente conectada.

Desde un punto de vista computacional práctico, UMAP puede describirse en términos de operaciones sobre grafos ponderados. Esto implica que UMAP pertenece a la clase de algoritmos de aprendizaje de grafos basados en los k-vecinos más cercanos, al igual que t-SNE. Debido a lo anterior, UMAP puede describirse en dos fases principales. En la primera fase se construye un grafo ponderado y en la segunda fase se calcula una representación de baja dimensión de dicho grafo [67].

Este algoritmo requiere de un conjunto de parámetros que tienen un impacto significativo a la hora de definir la distribución resultante de los datos de baja dimensionalidad: el número de vecinos, la distancia mínima, el número de componentes y la métrica utilizada.

El número de vecinos corresponde a un parámetro que controla la manera en que UMAP equilibra la estructura local con la estructura global de los datos. Para lograrlo, el parámetro es usado para restringir el tamaño de la vecindad local que el algoritmo considera cuando intenta aprender la estructura del conjunto de datos. Si el número de vecinos es bajo, UMAP estará forzado a concentrarse en la estructura local, mientras que los valores altos impulsarán el algoritmo a considerar vecindades más grandes de cada punto cuando se estima la estructura del conjunto de datos.

La distancia mínima es un parámetro que controla la capacidad de UMAP para agrupar los puntos. Este parámetro indica la distancia mínima permitida entre los puntos dentro de la representación de baja dimensión. Esto implica que valores bajos provocarán estructuras más burdas, útil para casos de agrupamiento (*clustering*). Para valores grandes de distancia mínima, se reducirá la capacidad del algoritmo para juntar puntos y éste se enfocará en la preservación de la estructura topológica amplia de los datos.

El número de componentes es un parámetro que permite al usuario determinar la dimensionalidad del espacio de baja dimensión en el que serán ubicados los datos. Por lo general, se busca obtener un espacio bidimensional o tridimensional para permitir la visualización de los datos.

Por último, la métrica representa un parámetro que controla la manera en que se calcula la distancia en el espacio de los datos de entrada. La métrica a utilizar puede estar dada por la distancia euclidiana, la distancia de mahalanobis, la distancia cosenoidal, entre otras.

2.5. Métricas

Para evaluar la capacidad de reconstrucción de los autocodificadores, se utilizan métricas que permiten definir un valor de error al comparar los datos de entrada con los de salida. Tal es el caso del error absoluto medio, que representa una opción a utilizar cuando los datos corresponden a imágenes. Otras métricas como la precisión y la exhaustividad, son empleadas para evaluar los resultados de algoritmos clasificadores como las máquinas de soporte vectorial (SVM).

2.5.1. Error absoluto medio (MAE)

El error absoluto medio corresponde a una medida de la magnitud promedio de los errores en un conjunto de predicciones, sin considerar su dirección. En el ámbito de la reconstruccion de imágenes, MAE representa el promedio sobre la imagen de prueba de las diferencias absolutas entre la imagen reconstruida y la imagen original, donde todas las diferencias individuales tienen el mismo peso. El error absoluto medio se calcula como

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i|\,, \tag{2.14}$$

donde y representa la imagen original y $\hat{y}$ la imagen reconstruida, ambas conformadas por N pixeles.

El error absoluto medio se utiliza como métrica para evaluar el error de reconstrucción de un autocodificador cuyas entradas son imágenes. En este caso, interesa obtener el error de reconstrucción de un autocodificador cuando sus entradas corresponden a imágenes aéreas de plantas de café.

2.5.2. Precisión y Exhaustividad

En el ámbito de clasificación de datos, la precisión representa la proporción de muestras positivas que se clasificaron correctamente con respecto al número total de muestras positivas previstas [62], tal y como se presenta en la relación

$$P = \frac{T_p}{T_p + F_p}, \tag{2.15}$$

donde T_p es la cantidad de verdaderos positivos y F_p la cantidad de falsos positivos. La exhaustividad representa las muestras positivas correctamente clasificadas con respecto al

número total de muestras positivas [62]. Esta métrica se representa como

$$R = \frac{T_p}{T_p + F_n}, \tag{2.16}$$

donde F_n representa la cantidad de falsos negativos.

2.6. Preprocesamiento de imágenes

El preprocesamiento de imágenes representa una etapa en la que se modifica su contenido para potenciar información sobre alguna característica de interés como el color, la textura, o información sobre algún canal determinado, de manera que dicha información pueda ser incluida como parte del entrenamiento de una red neuronal. Para incluir información sobre la textura de las imágenes, se utilizan algoritmos descriptores de textura como el patrón local binario (LBP).

2.6.1. Patrón local binario (LBP)

El operador LBP (*Local Binary Pattern*) corresponde a un algoritmo utilizado para el análisis de texturas invariables en imágenes a escala de grises. Este análisis considera una definición general de textura en un vecindario local.

Para cada pixel en una imagen, se establece un código binario producto de umbralizar el valor del pixel con el valor del pixel central en un vecindario determinado. Posteriormente, se crea un histograma que permite obtener las ocurrencias de diferentes patrones binarios, donde el vecindario puede estar dado por los ocho vecinos más cercanos a un pixel, o incluso por los P pixeles que conformen un vecindario circular [68].

El código binario LBP se define de la siguiente manera:

$$\mathrm{LBP}_{P,R} = \sum_{p=0}^{P-1} s(g_p - g_c)2^p, \tag{2.17}$$

donde g_c representa el valor de intensidad del nivel de gris del pixel central y g_p representa el valor del pixel vecino respectivo del pixel central. P corresponde al número de vecinos, R es el radio de la vecindad circular y $s(x)$ es la función escalón unitario definida por la siguiente relación:

$$s(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases}. \tag{2.18}$$

Además, se han desarrollado distintas versiones del método LBP para mejorar la eficacia ante diferentes condiciones de textura: LBP uniforme, LBP invariante a la rotación y LBP uniforme e invariante a la rotación [69].

Algunas propiedadades del operador LBP son su tolerancia a los cambios de iluminación y su simplicidad computacional, lo que permite analizar imágenes incluso en tiempo real. El método LBP se ha empleado en aplicaciones como la recuperación de imágenes, la teledetección, el análisis de imágenes biomédicas, el análisis de imágenes faciales, análisis de movimiento, el modelado de entornos, entre otras [68].

3 | Detección de anomalías en imágenes aéreas de ecosistemas cafetaleros

Las arquitecturas de autocodificador descritas en el capítulo anterior, poseen la capacidad de reconstruir imágenes y proveer un error de reconstrucción a partir del cual, se determina si las imagenes de entrada pueden clasificarse como anomalías o no. Si la clasificación de los datos es binaria, los autocodificadores pueden ser entrenados utilizando imágenes que representan solamente una de las dos clases de interés. De esta manera, las imágenes que no pertenecen a dicha clase tendrán mayor probabilidad de ser clasificadas como anomalías.

En la figura 3.1, se presenta un diagrama de la solución propuesta para la detección de anomalías en imágenes aéreas de ecosistemas cafetaleros, donde es necesario considerar como anomalías a todas las regiones de dichas imágenes que no corresponden a plantaciones de café. En este capítulo, se describen los componentes principales que forman parte del sistema propuesto para la detección de anomalías en imágenes aéreas de fincas cafetaleras costarricenses.

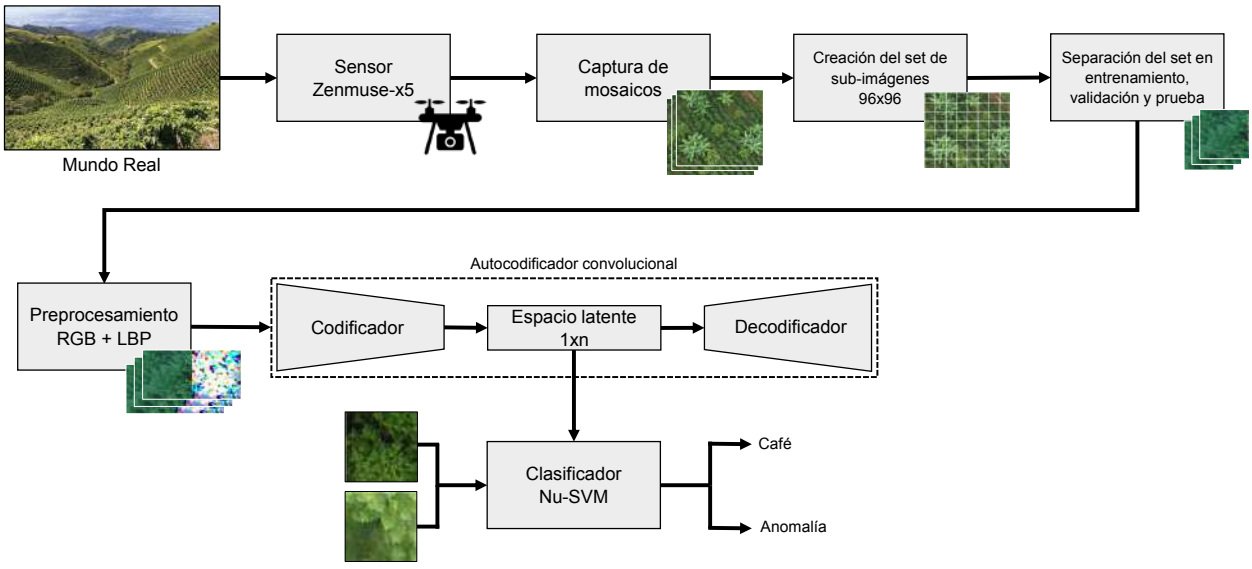


Figura 3.1: Sistema detector de anomalías en imágenes aéreas de ecosistemas cafetaleros.

La solución propuesta está conformada por un autocodificador convolucional y un clasificador binario basado en una SVM. A pesar de que el clasificador binario representa un enfoque de aprendizaje supervisado, su objetivo principal es generar resultados para evaluar el comportamiento de los datos en el espacio latente del autocodificador durante un proceso de pruebas. Para dicho proceso, se requiere la adición de un conjunto de sub-imágenes consideradas como anomalías. Sin embargo, la base fundamental de la solución radica en que el proceso de aprendizaje del autocodificador convolucional para la detección de anomalías, se desarrolla utilizando únicamente imágenes cuyo contenido corresponde a partes de plantas de café.

3.1. Captura de mosaicos

Los mosaicos corresponden a imágenes aéreas de 15.9 megapixeles ($4\,608 \times 3\,456$), capturadas con un sensor *Zenmuse-x5* adherido a un UAV. Los mosaicos presentan un formato de color RGB y están comprimidas mediante JPEG, con un tamaño entre 6 y 8 MB por archivo de imagen. El período de los vuelos de captura está comprendido entre julio de 2018 y agosto de 2019, realizados sobre plantaciones que contienen variedades de café de interés para el programa de semilla del ICAFÉ. En la tabla 3.1, se presenta la cantidad de vuelos y mosaicos capturados en cada una de las tres fincas seleccionadas para realizar los vuelos con el UAV.

Tabla 3.1: Fincas donde se realizaron los vuelos de captura de mosaicos.

Nombre	Ubicación	Vuelos	Mosaicos
ICAFÉ	Santa Bárbara de Heredia	10	5 648
Coopetarrazú	Zona de Los Santos	1	230
Finca Tres Ríos	Cartago	1	243

Dado que los mosaicos corresponden a capturas obtenidas en días y lugares distintos, algunas características como la textura, la variedad de vegetación adicional, la cantidad de luz solar y la densidad uniforme de las plantas, son variables presentes en las imágenes capturadas.

3.2. Creación del set de datos

El set de datos utilizado está conformado por un total de 1 350 000 sub-imágenes con un tamaño de 96×96 pixeles, extraídas a partir de los mosaicos capturados durante dos vuelos realizados en las fincas del ICAFÉ y Coopetarrazú. Dicho set de datos ha sido generado mediante la implementación de un algoritmo que permite recortar sub-imágenes de forma interactiva, permitiendo al usuario seleccionar regiones y definir el tamaño de las sub-imágenes

como parámetro de entrada. Para cada sub-imagen, se cuenta con información sobre las coordenadas que describen su ubicación en el mosaico a partir del cual se extrajo.

Una vez seleccionadas las regiones en los mosaicos por parte del usuario, el algoritmo toma dichas regiones y las recorre de izquierda a derecha, generando sub-imágenes de acuerdo con el tamaño definido por el usuario. El proceso de generación de sub-imágenes se realiza considerando un traslape entre dos sub-imágenes consecutivas horizontalmente, como se muestra en la figura 3.2. De esta manera, la sub-imagen 2 tendrá su primera mitad igual a la segunda mitad de la sub-imagen 1 y así sucesivamente.

Figura 3.2: Generación de sub-imágenes traslapadas a partir de las regiones seleccionadas en los mosaicos.

Del total de sub-imágenes generadas, se emplean 209 280 unidades para establecer dos versiones del set de datos: la versión I, compuesta por un total de 55 552 sub-imágenes y la versión II, con 153 728 sub-imágenes. Además de la cantidad de sub-imágenes, la diferencia entre ambas versiones radica en los algoritmos utilizados durante las etapas de preprocesamiento desarrolladas para entrenar, validar y probar diferentes versiones del autocodificador convolucional. En la tabla 3.2, se detalla la cantidad de sub-imágenes correspondientes a plantas de café y la cantidad de sub-imágenes consideradas como anomalías, utilizadas para propósitos de entrenamiento, evaluación y pruebas. Los sub-conjuntos de entrenamiento y validación no contienen anomalías, de modo que las redes neuronales sean entrenadas únicamente a partir de sub-imágenes de plantas de café.

Tabla 3.2: Sub-conjuntos de datos empleados en el proceso de experimentación.

	Versión I			Versión II		
Propósito	Sub-imágenes	Café	Anomalías	Sub-imágenes	Café	Anomalías
Entrenamiento	49 408	49 408	0	126 080	126 080	0
Validación	3 072	3 072	0	13 824	13 824	0
Prueba	3 072	1 536	1 536	13 824	6 912	6 912

Los tres sub-conjuntos establecidos con propósitos de entrenamiento, validación y pruebas,

contienen sub-imágenes asignadas de forma aleatoria. De esta manera, las unidades presentes en un sub-conjunto determinado, son extraídas de todos los mosaicos utilizados para dichos propósitos. Lo anterior implica que en los tres sub-conjuntos definidos, es posible tener sub-imágenes extraídas de un mismo mosaico.

El sub-conjunto de pruebas es el único que contiene sub-imágenes a considerar como anomalías, siendo posible evaluar la capacidad de los modelos entrenados para descartar sub-imágenes que no corresponden a plantas de café. En la figura 3.3, se pueden apreciar dos grupos con ejemplos de sub-imágenes, donde el grupo de la izquierda muestra sub-imágenes de anomalías y el grupo de la derecha presenta sub-imágenes de plantas de café. Las sub-imágenes de anomalías contienen información de árboles, personas, caminos, carreteras, autos, techos y de otros elementos que también forman parte de los mosaicos capturados.

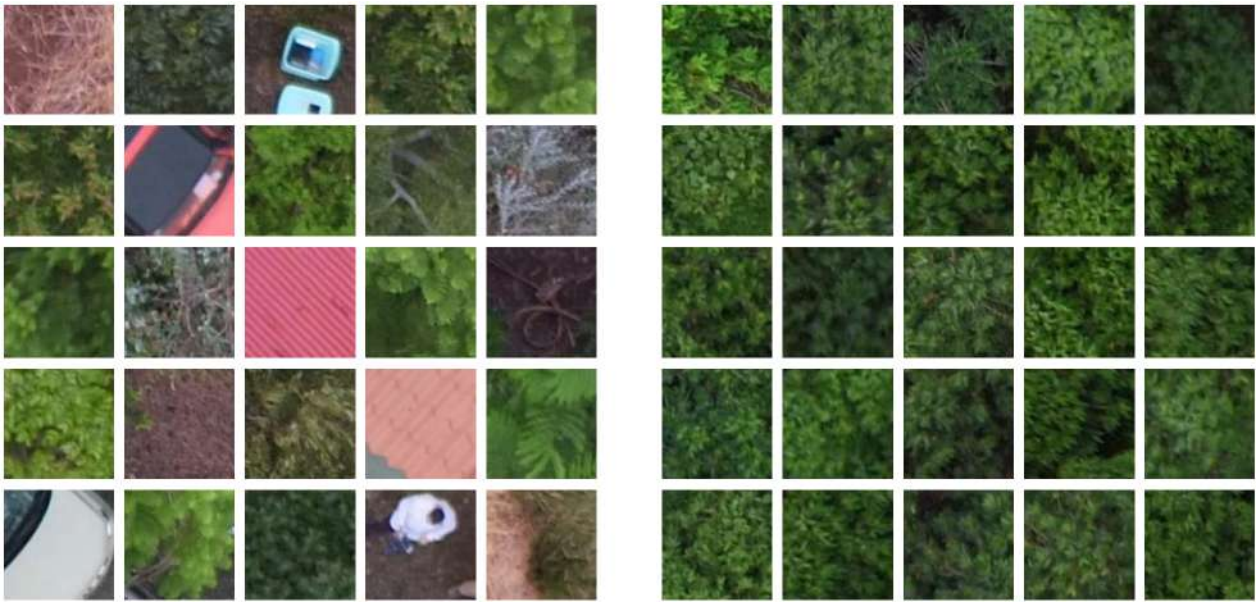


Figura 3.3: Grupos de sub-imágenes de anomalías (izquierda) y café (derecha).

3.3. Preprocesamiento de sub-imágenes

Existe otra diferencia entre las versiones I y II del set de datos utilizado, además de la cantidad de sub-imágenes que las conforman. En la versión I del set de datos, se tienen sub-imágenes que contienen secciones con detalles que no pertenecen a las plantas de café, como partes de caminos, raíces e incluso otras plantas. En la versión II del set de datos, las sub-imágenes con dichas secciones han sido descartadas como parte de un proceso de limpieza del set. En la figura 3.4, se presentan dos grupos de sub-imágenes que forman parte de las versiones I y II del set de datos respectivamente. En la fila superior del grupo de la izquierda correspondiente a la versión I, se aprecian tres sub-imágenes que contienen secciones que no corresponden a

las plantas de café, mientras que en la fila inferior de la izquierda y en ambas filas del gupo de la derecha, todo el contenido de las sub-imágenes representa partes de las plantas de interés.

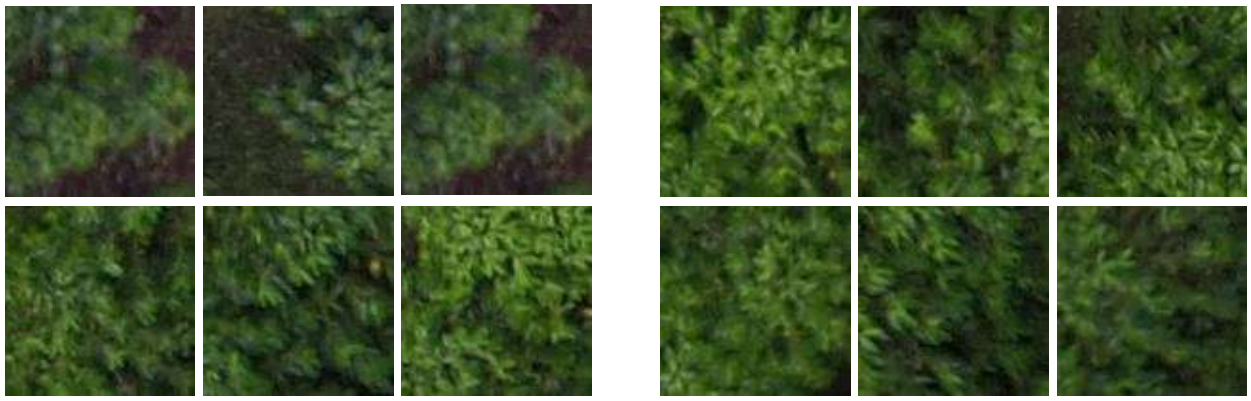


Figura 3.4: Detalles descartados de la versión I (izquierda) del set de datos con respecto a la versión II (derecha).

Las sub-imágenes de ambas versiones corresponden a imágenes RGB que poseen tres canales con intensidades en el rango $[0, 255]$. Como parte del preprocesamiento, se aplica el descriptor LBP a cada sub-imagen de la versión II del set, utilizando una cantidad de 24 pixeles que definen un vecindario circularmente simétrico, así como un radio de 8 pixeles para dicho vecindario circular. Para generar tres canales de la versión LBP de cada sub-imagen, el algoritmo se aplica cada uno, dos y cuatro pixeles de la sub-imagen original, obteniendo tres resultados parciales que fueron apilados para conformar una sub-imagen filtrada con tres canales. Finalmente, la versión RGB original de cada sub-imagen se concatena horizontalmente con su versión LBP, obteniendo sub-imágenes de tres canales con el doble del ancho de las sub-imágenes originales.

En la figura 3.5, se muestra el resultado de aplicar el preprocesamiento mencionado a una sub-imagen de café y a una anomalía respectivamente. La adición de las versiones LBP de las sub-imágenes, permite que las redes neuronales sean entrenadas con información sobre la textura presente en los datos y no solamente sobre las características del color. Las intensidades de la parte LBP de cada sub-imagen se encuentran en el rango $[0, 25]$.

3.4. El autocodificador convolucional

El autocodificador convolucional, corresponde a la arquitectura de red neuronal seleccionada para formar parte de la solución propuesta en este trabajo. La selección de redes neuronales convolucionales como algoritmo base para el diseño e implementación del autocodificador, radica en su simplicidad en comparación con enfoques más elaborados como VAE y AAE. La simplicidad del autocodificador convolucional, facilita el análisis del comportamiento de

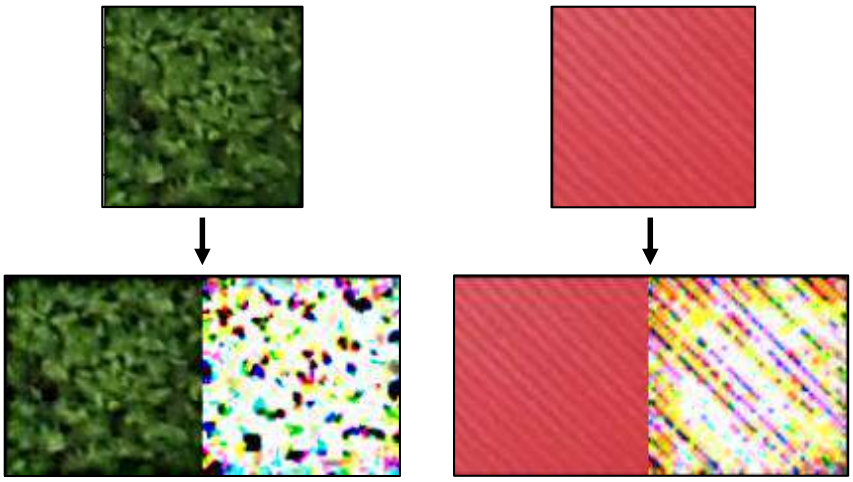


Figura 3.5: Preprocesamiento de sub-imágenes utilizando LBP en café (izquierda) y en una anomalía (derecha).

los datos en el espacio latente, considerada como una tarea clave para tomar decisiones con miras a mejorar el diseño del autocodificador.

Con el fin de desarrollar el proceso experimental en dos etapas principales, se establecen dos versiones de autocodificador convolucional: una versión α entrenada, validada y probada empleando las sub-imágenes de la versión I del set de datos, y una versión β donde se utiliza la versión II del set. La motivación de utilizar dichas versiones, está dada por la necesidad de evaluar el impacto en los resultados de reconstrucción y clasificación de sub-imágenes que presenta el realizar cambios significativos en los datos de entrada de los autocodificadores convolucionales, así como cambios en el tamaño de los espacios latentes que los conforman. A continuación, se presentan las principales características del diseño de ambas versiones de autocodificador convolucional.

3.4.1. Versión α

En esta primera versión de exploración, el autocodificador convolucional utiliza únicamente sub-imágenes RGB de la versión I del set de datos. En la figura 3.6, se presentan la capas que conforman la arquitectura del codificador α. Este codificador está compuesto por tres capas convolucionales que utilizan 64, 32 y 16 filtros de tamaño 3×3 respectivamente. En las capas de *max pooling* se definen ventanas de 2×2 y para las dos primeras capas de activación se emplea la función de activación $ReLU$.

Antes de ingresar a la red, las sub-imágenes de tres canales sufren modificaciones como parte de una etapa de preprocesamiento, donde el rango de sus intensidades se normaliza a $[-1, 1]$. Considerando a $\mathcal{I}$ como una sub-imagen RGB de tamaño 96×96 y a $i_{\max}$ como la intensidad

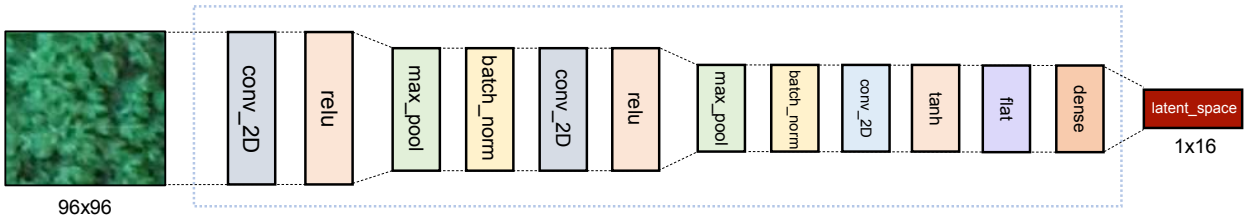


Figura 3.6: Arquitectura del codificador α.

máxima que pueden tomar sus pixeles, su versión normalizada $\hat{\mathcal{I}}$ se obtiene mediante la siguiente relación:

$$\hat{\mathcal{I}} = \mathcal{I}\left(\frac{2}{i_{\max}}\right) - 1, \tag{3.1}$$

donde $i_{\max}$ es igual a 255. Este nuevo rango ayuda en la convergencia del gradiente hacia un mínimo valor de error durante el proceso de aprendizaje. En la salida del codificador, se establece la función tangente hiperbólica como función de activación, con el objetivo de contar con valores en el rango $[-1, 1]$, de manera que sea coherente con el rango de los pixeles de la imagen de entrada. En esta etapa de exploración, se ha definido un único tamaño de espacio latente de 16 dimensiones para obtener un primer panorama sobre el comportamiento de los datos en dicho espacio.

En la figura 3.7, se muestran las capas que conforman la arquitectura del decodificador α, para el que se utilizan los mismos parámetros que en las capas equivalentes del codificador α. En la última capa del decodificador, también se emplea la función de activación tangente hiperbólica, para ubicar los pixeles de la sub-imagen resultante $\hat{\mathcal{I}}_R$ en el rango $[-1, 1]$.

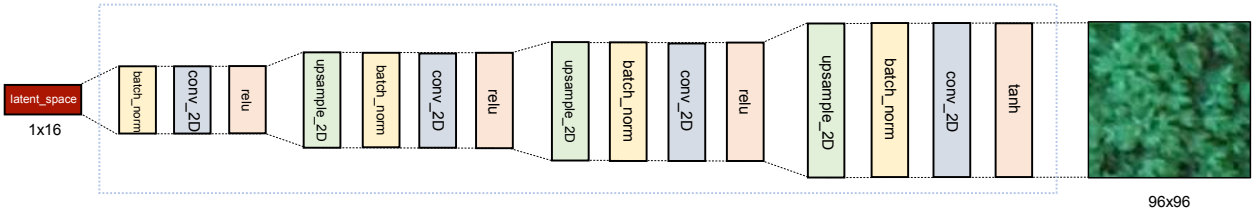


Figura 3.7: Arquitectura del decodificador α.

La salida final del decodificador, corresponde a la sub-imagen reconstruida $\mathcal{I}_R$ de 96×96 pixeles, obtenida a partir de la operación inversa de la ecuación (3.1), donde la sub-imagen $\hat{\mathcal{I}}_R$ se traslada al rango $[0, 255]$ mediante la siguiente ecuación:

$$\mathcal{I}_R = \frac{i_{\max}\left(\hat{\mathcal{I}}_R + 1\right)}{2}. \tag{3.2}$$

3.4.2. Versión β

El autocodificador convolucional β está compuesto por un codificador y un decodificador de cuatro capas convolucionales cada uno, donde el tamaño del espacio latente n se evalúa en 128, 256 y 512 dimensiones. En las figuras 3.8 y 3.9, se presentan las arquitecturas del codificador y decodificador β respectivamente.

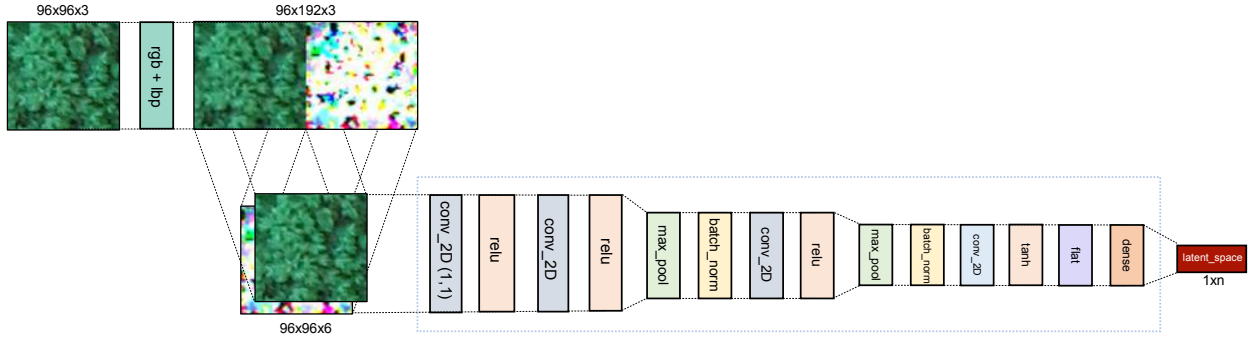


Figura 3.8: Arquitectura del codificador β.

Las entradas del codificador β corresponden a sub-imágenes RGB de la versión II del set de datos, cuyos pixeles son trasladados al rango $[-1, 1]$ mediante la ecuación (3.1). Para cada sub-imagen se obtiene su versión LBP, la cual también se traslada al rango $[-1, 1]$ utilizando $i_{\max}$ igual a 25. Cada sub-imagen es concatenada horizontalmente con su versión LBP para conformar una sub-imagen de 96×192 pixeles con tres canales.

Posteriormente, esta sub-imagen es separada a la mitad en dos secciones que son apiladas para conformar una sub-imagen de 96×96 con seis canales: tres canales de su versión RGB que representan información del color y tres canales de la versión LBP con información sobre su textura. El aprendizaje del autocodificador convolucional β, se desarrolla a partir de dichas sub-imágenes de seis canales.

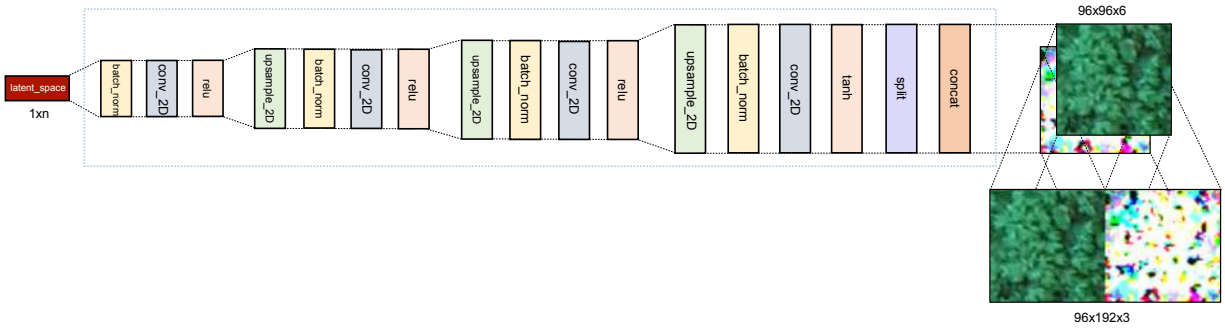


Figura 3.9: Arquitectura del decodificador β.

Para la primera capa convolucional del codificador β, se utilizan 64 filtros de tamaño 1×1. Esto permite a la red aplicar una primera convolución sobre la información contenida a lo

profundo de los seis canales de la sub-imagen. Para las otras tres capas convolucionales del codificador, se emplean 64, 32 y 16 filtros de tamaño 3×3 respectivamente.

En el decodificador se utilizan los mismos parámetros que en las capas equivalentes del codificador β, a excepción de la primera capa convolucional. La salida final del decodificador β, corresponde a una sub-imagen reconstruida donde las versiones RGB y LBP de la entrada están concateadas, lo que permite calcular el error de construcción de la salida con respecto a la entrada.

Para el proceso de entrenamiento de ambos autocodificadores convolucionales α y β, se aplica un aumento de los datos mediante tres transformaciones en línea de las sub-imágenes, las cuales están dadas por un acercamiento del 2 %, un volteo vertical y otro horizontal.

3.4.3. Hiperparámetros

Como parte del proceso de entrenamiento de ambos autocodificades convolucionales, se establece la métrica del error absoluto medio (MAE) como función de pérdida. Esta métrica también se utiliza para evaluar la capacidad de reconstrucción y la precisión de los autocodificadores. La selección de MAE como métrica, está dada por una premisa donde se asume que los datos siguen una distribución cercana a una distribución normal con un conjunto de datos atípicos que no forman parte de dicha distribución. Bajo este escenario, MAE permite cuantificar todas las desviaciones incluso cuando éstas se presentan en sentidos opuestos, como es posible que suceda al comparar sub-imágenes.

En cuanto a los hiperparámetros definidos para el entrenamiento, todos los modelos generados con ambas versiones de autocodificador, son configurados utilizando un optimizador ADAM con una tasa de aprendizaje inicial de 1×10^{-3} y una tasa de decaimiento dada por la razón entre la tasa de aprendizaje y el número de épocas. En la tabla 3.3, se presenta la cantidad de épocas y el tamaño de los lotes definidos como hiperparámetros de los autocodificadores convolucionales α y β. Estos hiperparámetros son seleccionados empíricamente como resultado de un ajuste de valores y su análisis durante experimentos preliminares sobre los procesos de entrenamiento y validación de ambos autocodificadores.

Tabla 3.3: Hiperparámetros utilizados para entrenar las versiones α y β de los autocodificadores convolucionales.

Versión	Tamaño Lotes	Épocas
α	32	60
β	64	40

3.5. El clasificador binario

El clasificador binario corresponde a la implementación de un modelo ν de una máquina de soporte vectorial (ν-SVM), empleado para clasificar los datos de salida del codificador β como café o como anomalías. Este modelo requiere de tres parámetros fundamentales para ser configurado. El primero de ellos corresponde al kernel, donde se emplea la función de base radial (RBF). El segundo parámetro corresponde a γ (gamma), utilizado por la función del kernel RBF para determinar qué tan rápido decrece la métrica de similitud entre los puntos. El tercer parámetro corresponde a ν (nu), que controla la cantidad de vectores de soporte y los errores de margen. Un error de margen corresponde a un dato que se encuentra en el lado equivocado de su límite de margen definido con respecto al hiperplano de separación.

Para definir el valor de los parámetros a utilizar en la configuración del modelo ν-SVM con la mayor exactitud posible, se aplica un método de ajuste exhaustivo de parámetros conocido como *grid search*. Este método toma un rango de valores para cada parámetro e inicia la búsqueda de los valores que producen los resultados de clasificación con la exactitud más alta. La tabla 3.4 presenta los rangos empleados en la búsqueda exhaustiva para definir los valores óptimos de los parámetros ν y γ, utilizando el kernel RBF. Estos rangos son determinados durante experimentos preliminares, donde se descartan valores de parámetros que producen resultados con menor exactitud.

Tabla 3.4: Rangos definidos para buscar los parámetros óptimos del clasificador ν-SVM.

Parámetro	Rango
ν	$[0.03 - 0.04]$, cada 0.005
γ	$[0.0010 - 0.0022]$, cada 0.0002

Además de las versiones I y II del set de datos, se establece un tercer conjunto compuesto por 33 024 sub-imágenes de café y anomalías con los propósitos de entrenar y probar el clasificador binario ν-SVM una vez configurado. La tabla 3.5 presenta la cantidad de sub-imágenes utilizadas para dichos propósitos. Los datos de entrada del clasificador corresponden a la salida del codificador cuando se ingresan dichas sub-imágenes al autocodificador β. Es decir, corresponden a datos que residen en el espacio latente, como se ilustra en la figura 3.1.

Tabla 3.5: Sub-conjuntos de datos empleados en el proceso de clasificación.

Propósito	Sub-imágenes	Café	Anomalías
Entrenamiento	22 016	22 016	0
Prueba	11 008	5 504	5 504

4 | Resultados

Este capítulo presenta los resultados experimentales obtenidos con los autocodificadores convolucionales α y β, comparando los procesos de entrenamiento y validación de los modelos generados utilizando ambos autocodificadores. También se describen los resultados obtenidos mediante un método de evaluación de predicciones, empleado para seleccionar los modelos con el mayor balance entre precisión y exhaustividad. Utilizando los modelos seleccionados, se obtienen resultados sobre la capacidad de reconstrucción de sub-imágenes de café y sobre la distribución de los datos en el espacio latente.

Además, se presentan los resultados cuantitativos de un análisis sobre la densidad de probabilidad del error de reconstrucción. También se describen los resultados obtenidos mediante el clasificador binario ν-SVM, al cuantificar los aciertos y los desaciertos durante la clasificación de sub-imágenes. Por último, se presentan los resultados cualitativos de reconstruir un conjunto de mosaicos en su totalidad, extrayendo todas las sub-imágenes que conforman dichos mosaicos y clasificándolas como café o anomalías.

4.1. Evaluación del entrenamiento y validación de los autocodificadores

Para evaluar los procesos de entrenamiento y validación de los autocodificadores convolucionales α y β, se obtienen las curvas de pérdida y exactitud. Estas curvas permiten analizar el comportamiento de ambos procesos durante las épocas definidas para generar los modelos con dichos autocodificadores. En este caso, se genera un modelo por cada época para facilitar su evaluación. Los procesos de entrenamiento y validación de ambos autocodificadores, se llevan a cabo utilizando únicamente sub-imágenes correspondientes a plantas de café.

La figura 4.1, muestra las curvas de pérdida obtenidas durante el entrenamiento y la validación de los modelos del autocodificador convolucional α durante 60 épocas. Estos modelos presentan un tamaño de espacio latente fijo equivalente a 16 dimensiones. La función de pérdida asociada corresponde a MAE y es posible determinar que ambas curvas convergen hacia un 1.25 % sin presencia de un sobreajuste evidente, dado que la curva de pérdida de validación no presenta ascensos pronunciados sobre la curva de entrenamiento.

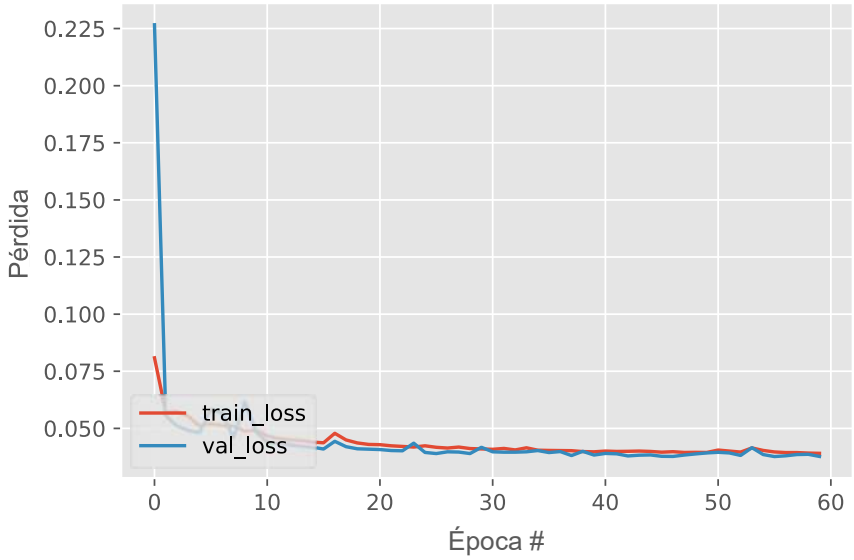


Figura 4.1: Curvas de pérdida de los procesos de entrenamiento y validación durante 60 épocas con el autocodificador convolucional α, para modelos con tamaño de espacio latente equivalente a 16 dimensiones.

En la figura 4.2 se observan las curvas de entrenamiento y validación para los modelos del autocodificador convolucional β, obtenidos en un proceso desarrollado durante 40 épocas. Las curvas muestran el comportamiento del entrenamiento y validación para tres modelos con tamaños de espacio latente equivalentes a 128, 256 y 512 dimensiones respectivamente.

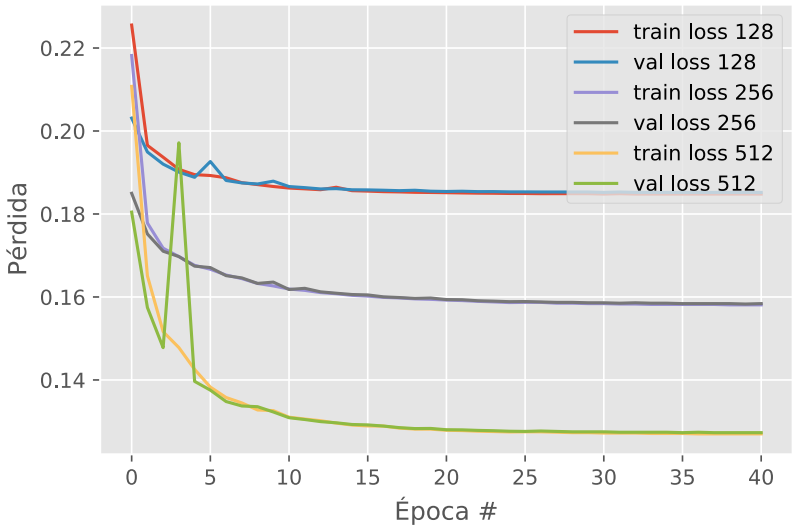


Figura 4.2: Curvas de pérdida de los procesos de entrenamiento y validación durante 40 épocas con el autocodificador convolucional β, para modelos con tamaños de espacio latente equivalentes a 128, 256 y 512 dimensiones.

Se determina que las curvas con la menor pérdida corresponden al modelo con espacio latente de 512 dimensiones, donde se observa un sobreajuste durante las primeras épocas. Posteriormente, el proceso se estabiliza convergiendo a un valor de pérdida equivalente al 12.6 %, determinado mediante la información presente en las curvas verde y amarilla. En este caso, también se emplea la métrica MAE como función de pérdida. Para los modelos con espacios latentes de 128 y 256 dimensiones, las curvas convergen a valores de pérdida de 18.6 % y 15.8 % respectivamente.

La figura 4.3 presenta las curvas de exactitud para el entrenamiento y la validación de los modelos del autocodificador convolucional α, obtenidos durante 60 épocas. Las curvas convergen a un valor de exactitud equivalente al 98 %, calculado con base a un umbral del 50 %.

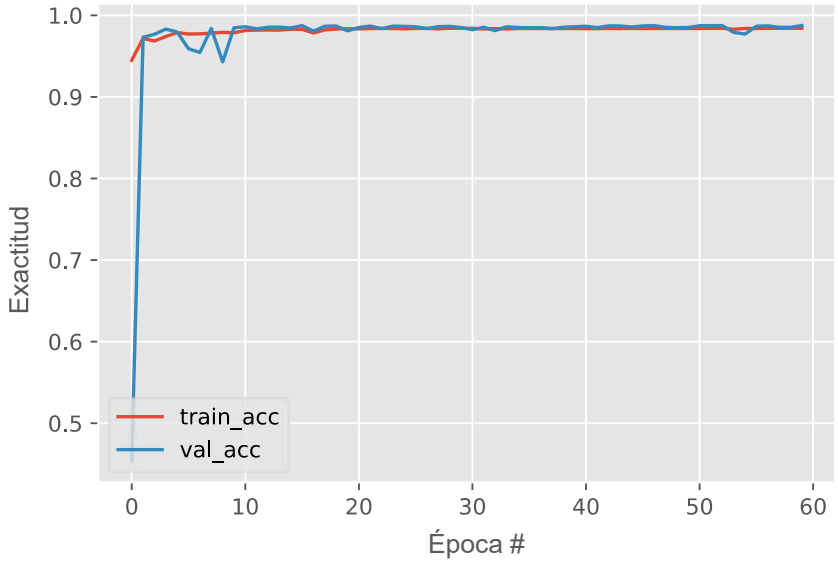


Figura 4.3: Curvas de exactitud de los procesos de entrenamiento y validación durante 60 épocas con el autocodificador convolucional α, para modelos con tamaño de espacio latente equivalente a 16 dimensiones.

En la figura 4.4, se observan las curvas de exactitud obtenidas en los procesos de entrenamiento y validación desarrollados durante 40 épocas, donde se generan los modelos del autocodificador convolucional β con tamaños de espacio latente de 128, 256 y 512 dimensiones. Las curvas con la mayor exactitud corresponden al modelo con espacio latente de 256 dimensiones, las cuales convergen a un valor de 84.25 %.

El modelo con espacio latente de 128 dimensiones presenta una convergencia a un 83.1 % de exactitud, mientras que el modelo con espacio latente de 512 dimensiones decae a partir de la quinta época hasta converger a un valor de 81.5 %, siendo el modelo con la menor exactitud durante los procesos de entrenamiento y validación. Las oscilaciones pronunciadas presentes en la curva de validación del modelo con espacio latente de 512 dimensiones, indican que el modelo puede tener problemas para generalizar debido a un set de validación que no es lo suficientemente representativo en comparación con el tamaño del set de entrenamiento.

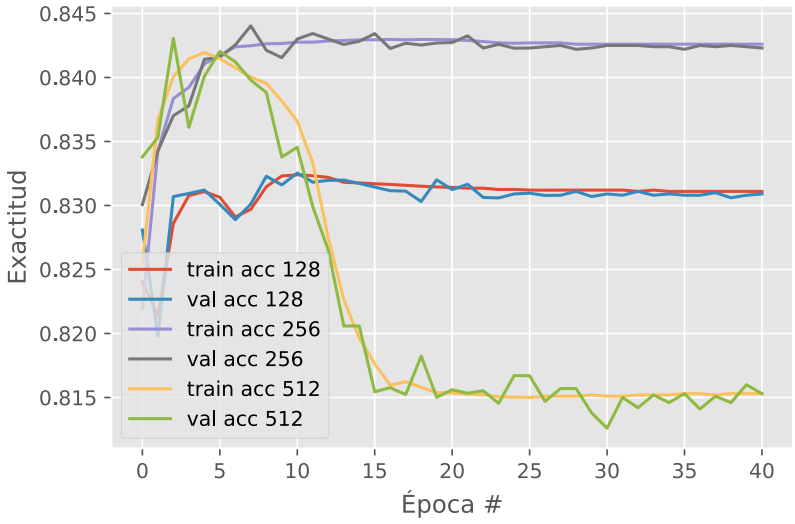


Figura 4.4: Curvas de exactitud de los procesos de entrenamiento y validación durante 40 épocas con el autocodificador convolucional β, para modelos con tamaños de espacio latente equivalentes a 128, 256 y 512 dimensiones.

La tabla 4.1 presenta información sobre las plataformas utilizadas para ejecutar los procesos de entrenamiento, validación y pruebas con ambas versiones de autocodificadores convolucionales. Además, se muestra el tiempo promedio de procesamiento en segundos t_{ep}, requerido para la ejecución de cada época en la generación de todos los modelos.

Tabla 4.1: Plataformas y tiempos de ejecución requeridos para entrenar y validar los modelos de los autocodificadores convolucionales α y β.

Versión	**Dim**	**Servicio**	**Plataforma**	**Épocas**	t_{ep}(s)
α	16	Google Colab	NVIDIA Tesla T4	60	223
β	128	Google Colab	NVIDIA Tesla V100-SXM2	40	1244
	256				1425
	512				1416

4.2. Selección de los modelos con el mayor balance entre precisión y exhaustividad

Durante los procesos de entrenamiento y validación de los autocodificadores convolucionales, se genera un modelo por cada época. Una vez analizadas las curvas de pérdida y exactitud

para los autocodificadores α y β con diferente tamaño de espacio latente, se evalúan todos los modelos para identificar en cuáles épocas se obtiene el mayor balance entre precisión y exhaustividad. Dicho balance está dado por el resultado que provee la mayor precisión y la mayor exhaustividad.

Como herramienta de comparación de modelos, se utilizan gráficos de precisión versus exhaustividad o gráficos PR. El punto con el mayor balance entre precisión y exhaustividad, corresponde a uno de los puntos más cercanos a la esquina superior derecha de los gráficos PR, donde se da prioridad al punto con la mayor precisión. Para determinar si la predicción de un modelo es falsa o verdadera, se establece un rango de valores de umbral de anomalías a partir de la métrica MAE. Esto implica que si el valor de MAE de una sub-imagen reconstruida a partir de la original es superior al umbral, dicha sub-imagen se considera como una anomalía. La tabla 4.2 presenta los rangos de umbral de anomalía y los rangos de épocas empleados en la búsqueda de los modelos con el mayor balance entre precisión y exhaustividad.

Tabla 4.2: Rangos de umbral MAE de anomalía utilizados para calcular precisión y exhaustividad.

Versión	**Dim**	**Rango épocas**	**Rango umbral**
α	16	$[16-40]$	$[0.10-0.28]$ cada 0.02
β	128 256 512	$[25-40]$	$[0.20-1.50]$ cada 0.10

Los rangos utilizados en las predicciones con ambas versiones de autocodificadores convolucionales son diferentes, ya que la inclusión de la parte LBP en las sub-imágenes utilizadas para entrenar, validar y probar los modelos del autocodificador β, produce valores de error MAE en un rango más amplio que con los modelos del autocodificador α. Para el caso de la versión α, los gráficos PR se construyen considerando únicamente los modelos generados después de la época 16, para un total de 25 puntos por cada gráfico y 10 gráficos en total. Para el autocodificador convolucional β, los gráficos se construyen a partir de la época 15, con un total de 16 puntos por cada gráfico y 14 gráficos por cada tamaño de espacio latente.

En cada gráfico PR se identifica con color rojo el punto dominante con el mayor balance entre precisión y exhaustividad. A la par de cada punto del gráfico, se indica el número de época en la que se obtiene un valor de precisión y un valor de exhaustividad para un umbral MAE determinado. Para ejemplificar este proceso, la figura 4.5 presenta los puntos que definen la curva PR para un umbral MAE de 0.16, utilizando los modelos del autocodificador α con tamaño de espacio latente de 16 dimensiones. En este caso, el punto dominante seleccionado representa el resultado de realizar predicciones con un modelo generado durante 34 épocas de entrenamiento. Con dicho modelo, se obtiene una precisión y una exhaustividad equivalentes

a 0.76. Esta igualdad, implica que para dicho modelo se obtiene una cantidad de falsos positivos igual a la cantidad de falsos negativos.

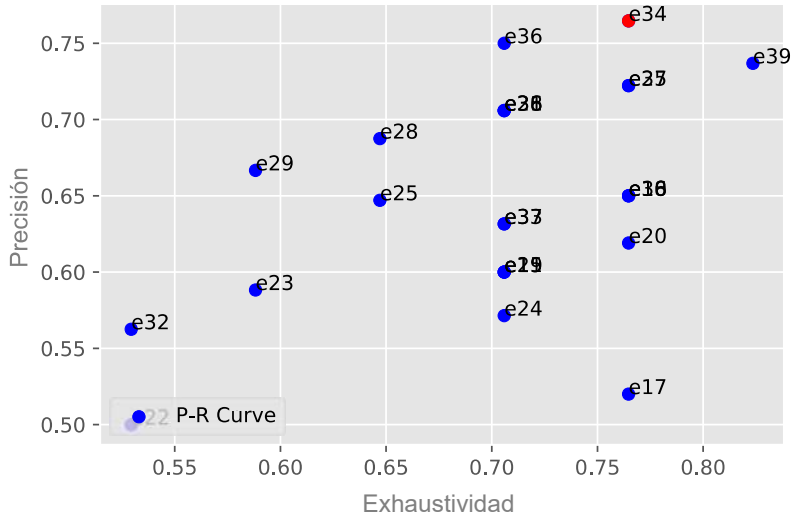


Figura 4.5: Gráfico PR para un umbral MAE de 0.16 usando el autocodificador α con tamaño de espacio latente igual a 16 dimensiones.

En la figura 4.6, se presenta una curva PR con todos los puntos dominantes obtenidos con los modelos del autocodificador α. Estos puntos conforman el frente de Pareto que denota los puntos óptimos de los 10 valores de umbral MAE evaluados. En el frente de Pareto también se identifica el punto dominante con color rojo, el cual representa un modelo entrenado durante 34 épocas y un umbral MAE de 0.16.

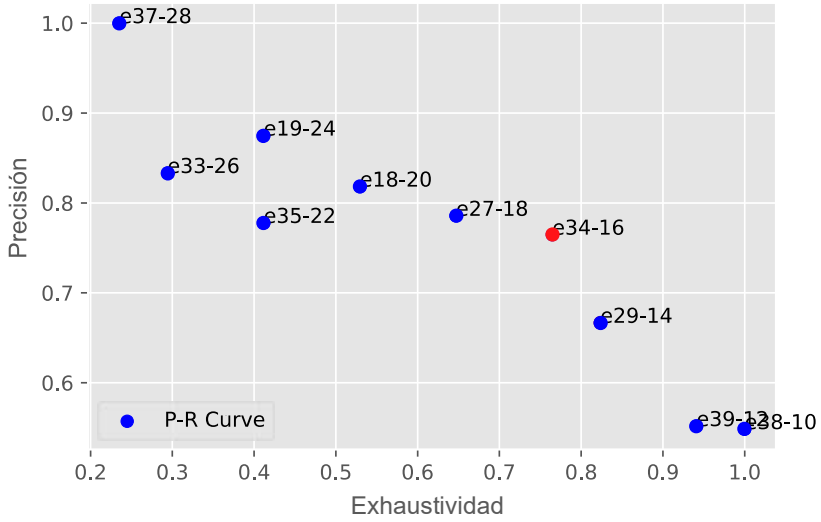


Figura 4.6: Frente de Pareto con los 10 mejores resultados de precisión y exhaustividad usando el autocodificador α con tamaño de espacio latente igual a 16 dimensiones.

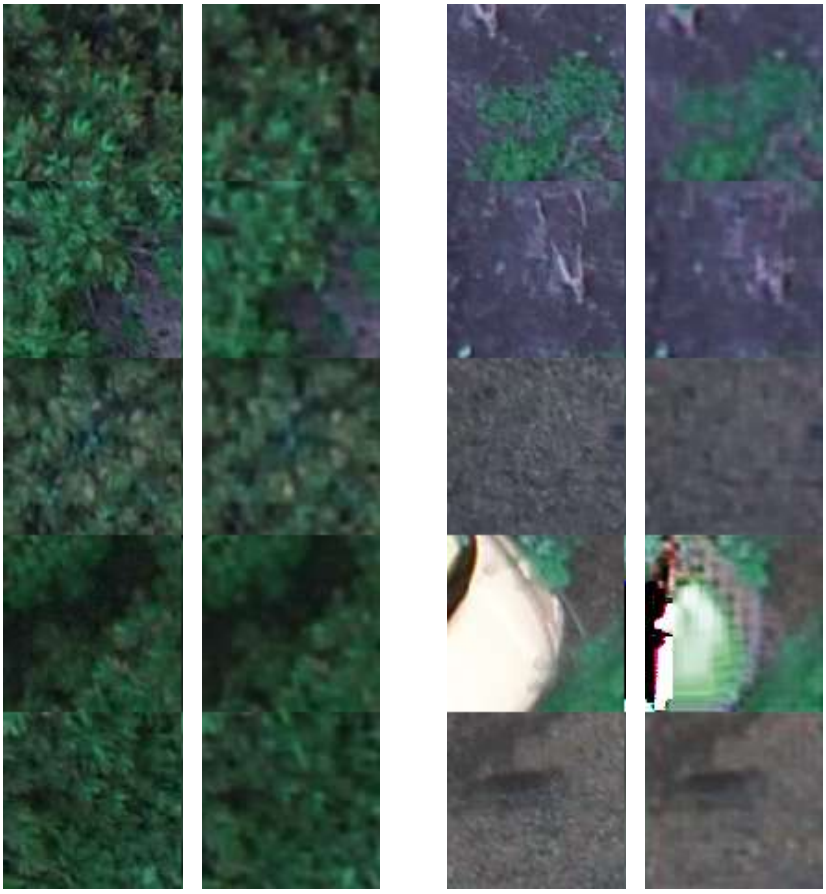
La tabla 4.3 presenta los números de época, los valores de precisión y exhaustividad obtenidos con los modelos de las versiones de los autocodificadores convolucionales α y β, que representan los puntos dominantes en cada frente de Pareto. En el caso del autocodificador convolucional β, los resultados muestran la ausencia de falsos positivos y una alta cantidad de falsos negativos para los modelos que presentan el mayor balance entre precisión y exhaustividad en sus predicciones.

Tabla 4.3: Precisión y exhaustividad de los modelos dominantes en los frentes de Pareto.

Versión	**Dim**	**Época**	**Umbral MAE**	**Precisión (%)**	**Exhaustividad (%)**
α	16	34/40	0.16	76.5	76.5
β	128	34/40	1.25	100	31.25
	256	37/40	1.25	100	21.87
	512	33/40	1.25	100	18.75

4.3. Reconstrucción de sub-imágenes

Una vez seleccionados los modelos de los autocodificadores α y β con el mayor balance entre precisión y exhaustividad en sus predicciones, se realiza una comparación cualitativa de la capacidad de reconstrucción de dichos modelos. En la figura 4.7, se muestran los resultados de reconstruir cinco sub-imágenes de café y cinco sub-imágenes de anomalías con el modelo seleccionado del autocodificador convolucional α. En este caso, el modelo es capaz de reconstruir tanto las sub-imágenes de café como las sub-imágenes de anomalías, incluyendo características de color y textura. Para las anomalías cuyo contenido no corresponde a plantas, el modelo introduce artefactos en la versión reconstruida, tal y como se observa en la cuarta sub-imagen de la figura 4.7d.

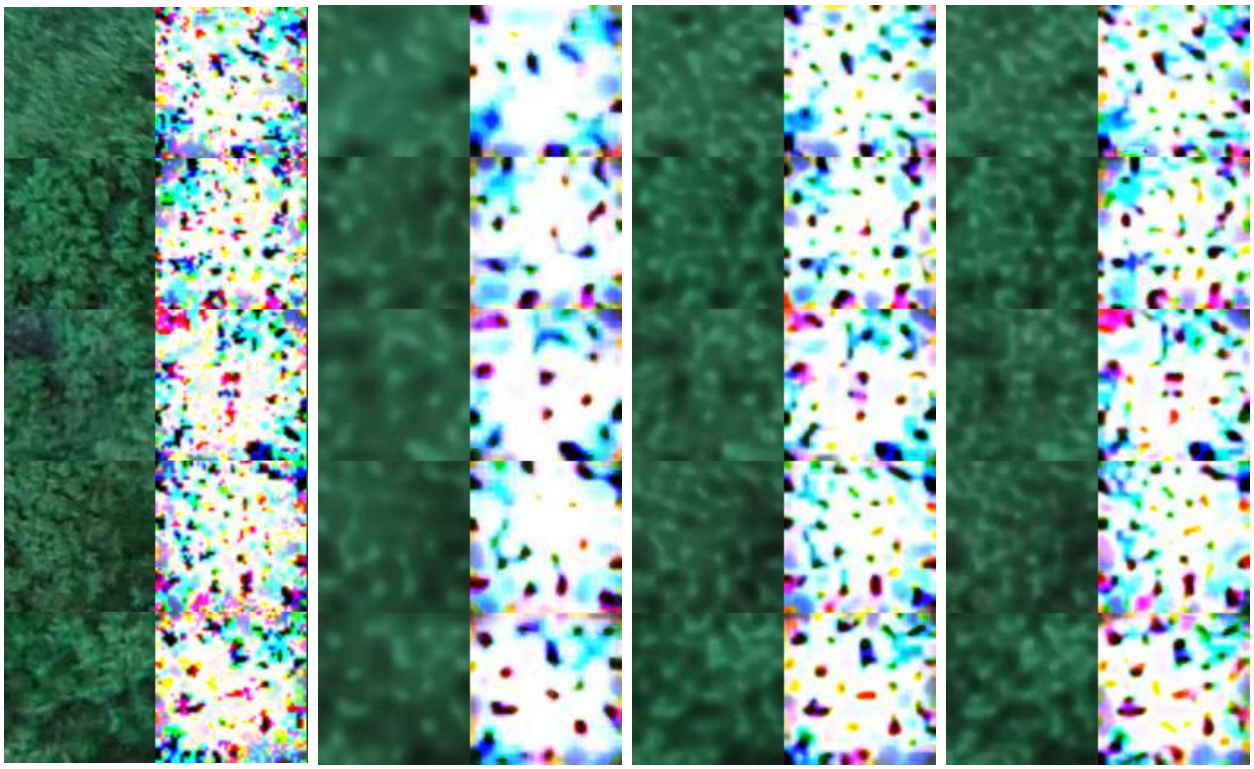


(a) café **(b)** reconst **(c)** anomalía **(d)** reconst

Figura 4.7: Ejemplos de reconstrucciones de café y anomalías al utilizar el autocodificador convolucional α con tamaño de espacio latente igual a 16 dimensiones.

La figura 4.8 presenta los resultados de reconstruir cinco sub-imágenes de café con los tres modelos seleccionados del autocodificador β, los cuales poseen tamaños de espacio latente

equivalentes a 128, 256 y 512 dimensiones respectivamente. Como es esperado, al incrementar el tamaño del espacio latente, se observan reconstrucciones con mayor detalle en las texturas.

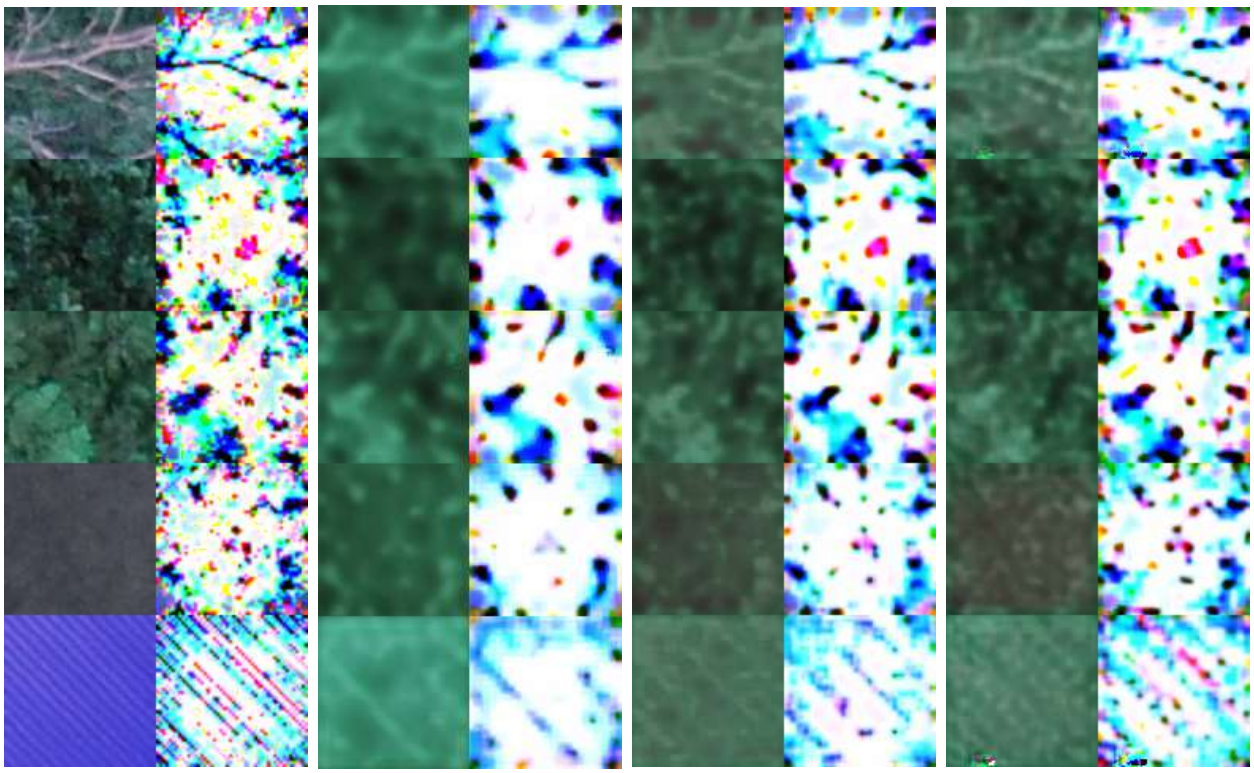


(a) original **(b)** lat dim: 128 **(c)** lat dim: 256 **(d)** lat dim: 512

Figura 4.8: Ejemplos de reconstrucciones al utilizar el autocodificador convolucional β con diferentes tamaños de espacio latente: 128, 256 y 512 dimensiones, a partir de las imágenes de entrada correspondientes a plantas de café.

La figura 4.9 muestra los resultados de reconstruir cinco sub-imágenes, esta vez de anomalías, con los tres modelos del autocodificador β. Al incrementar el tamaño del espacio latente, se observan reconstrucciones con mayor detalle. Sin embargo, puede notarse que los modelos aprenden a reconstruir de forma correcta únicamente información donde predomina el color verde. Por lo tanto, la rama café de la primera sub-imagen y la lámina púrpura de la última sub-imagen en la figura 4.9a, se intentan reconstruir del color incorrecto. Este aspecto se considera como positivo, ya que representa un indicador de que los modelos poseen la capacidad de descartar una gran mayoría de anomalías.

Dado el comportamiento de las reconstrucciones usando el autocodificador β con respecto a los resultados de la versión α, se reconoce al tamaño del espacio latente y a la versión del set de datos como factores clave en el proceso de predicción y reconstrucción de las sub-imágenes. Por esta razón, las siguientes etapas experimentales se desarrollan utilizando únicamente los modelos del autocodificador convolucional β.



(a) original **(b)** lat dim: 128 **(c)** lat dim: 256 **(d)** lat dim: 512

Figura 4.9: Ejemplos de reconstrucciones al utilizar el autocodificador convolucional con diferentes tamaños de espacio latente: 128, 256 y 512 dimensiones, a partir de las imágenes de entrada correspondientes a anomalías.

4.4. Análisis estadístico del error de reconstrucción

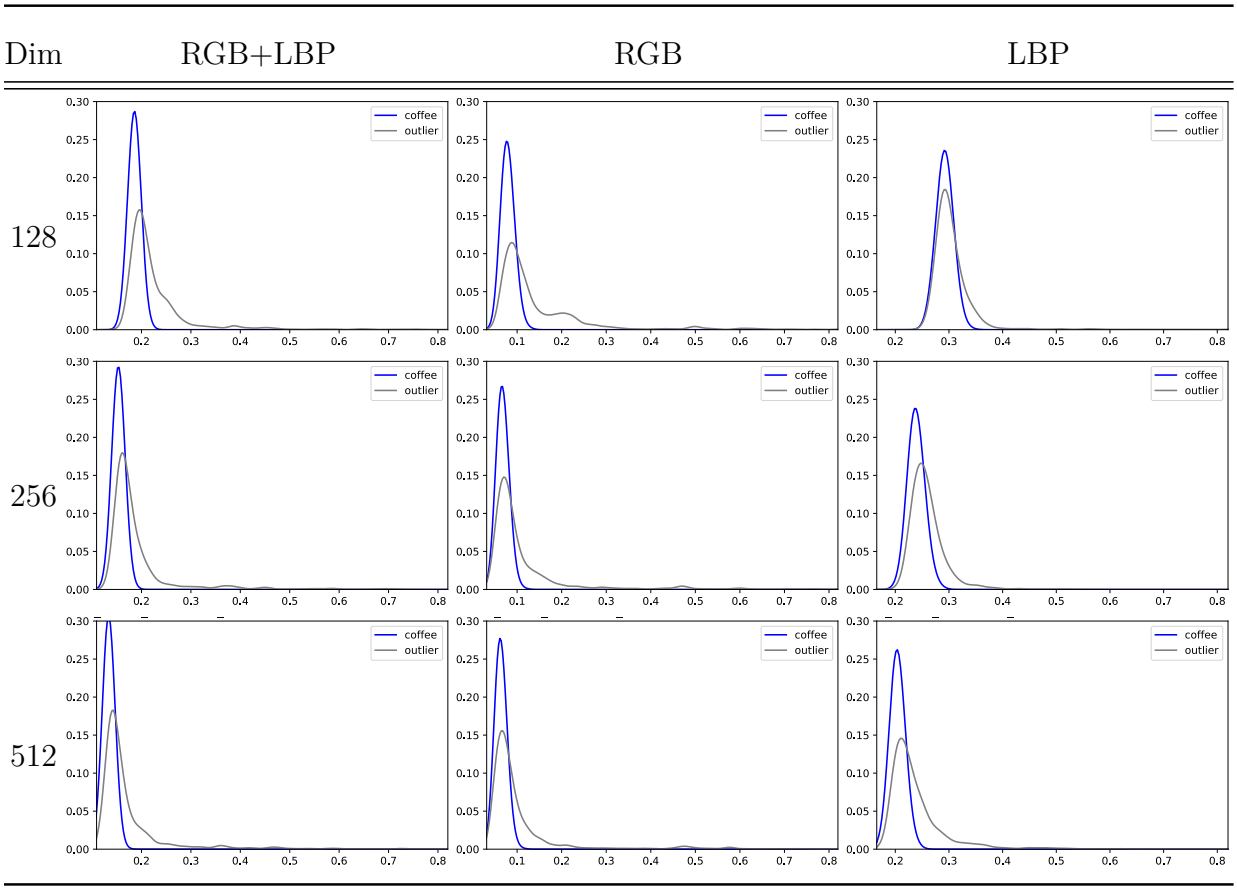
Los tres modelos seleccionados del autocodificador convolucional β son entrenados con sub-imágenes de café de 96×192 pixeles, compuestas por la captura RGB original concatenada horizontalmente con su versión LBP, como se ejemplifica en la primera columna de la figura 4.8. Para evaluar la capacidad de reconstrucción de dichos modelos de forma cuantitativa, se utiliza la métrica MAE como error de reconstrucción entre las sub-imágenes de entrada y las de salida del autocodificador convolucional. La capacidad de reconstrucción de los tres modelos seleccionados se evalúa en tres escenarios. En el primer escenario, se calcula el error de reconstrucción al comparar las sub-imágenes original y reconstruida completas (RGB+LBP). En el segundo escenario, el error de reconstrucción se calcula al comparar únicamente la mitad RGB de las sub-imágenes. De manera similar, en el tercer escenario se calcula el error de reconstrucción al comparar únicamente la mitad LBP de las sub-imágenes.

En este caso, se toman 3456 sub-imágenes de café y 3456 sub-imágenes de anomalías que pertenecen al set de pruebas de la versión II del set de datos y se utilizan para realizar las

predicciones con los tres modelos del autocodificador convolucional β. Como herramienta para el análisis estadístico, se utiliza el algoritmo de estimación de densidad con kernels (KDE), que corresponde a un método no paramétrico empleado para estimar la función de densidad de probabilidad de una variable aleatoria a partir de un número finito de observaciones.

Para aplicar KDE sobre los valores numéricos de los errores de reconstrucción obtenidos en cada escenario, se define un kernel gaussiano con un ancho de banda igual a 0.01. Dicho valor es seleccionado como una estimación del ancho de banda óptimo empleando correlación cruzada de 5 veces, a través de un método de ajuste exhaustivo de parámetros conocido como *grid search*. El criterio de selección del valor para el ancho de banda está basado en una medida de la verosimilitud de los datos con el kernel estimado. En la figura 4.10 se presentan los gráficos de la función de densidad de probabilidad para los errores de reconstrucción obtenidos utilizando MAE, al realizar las predicciones con los modelos de los tres tamaños de espacio latente en los tres escenarios descritos anteriormente. En cada gráfico, la función de probabilidad representada por la curva azul corresponde a $p(\text{MAE} \mid x = \text{café})$, mientras que la curva gris representa la función de probabilidad $p(\text{MAE} \mid x = \text{anomalía})$.

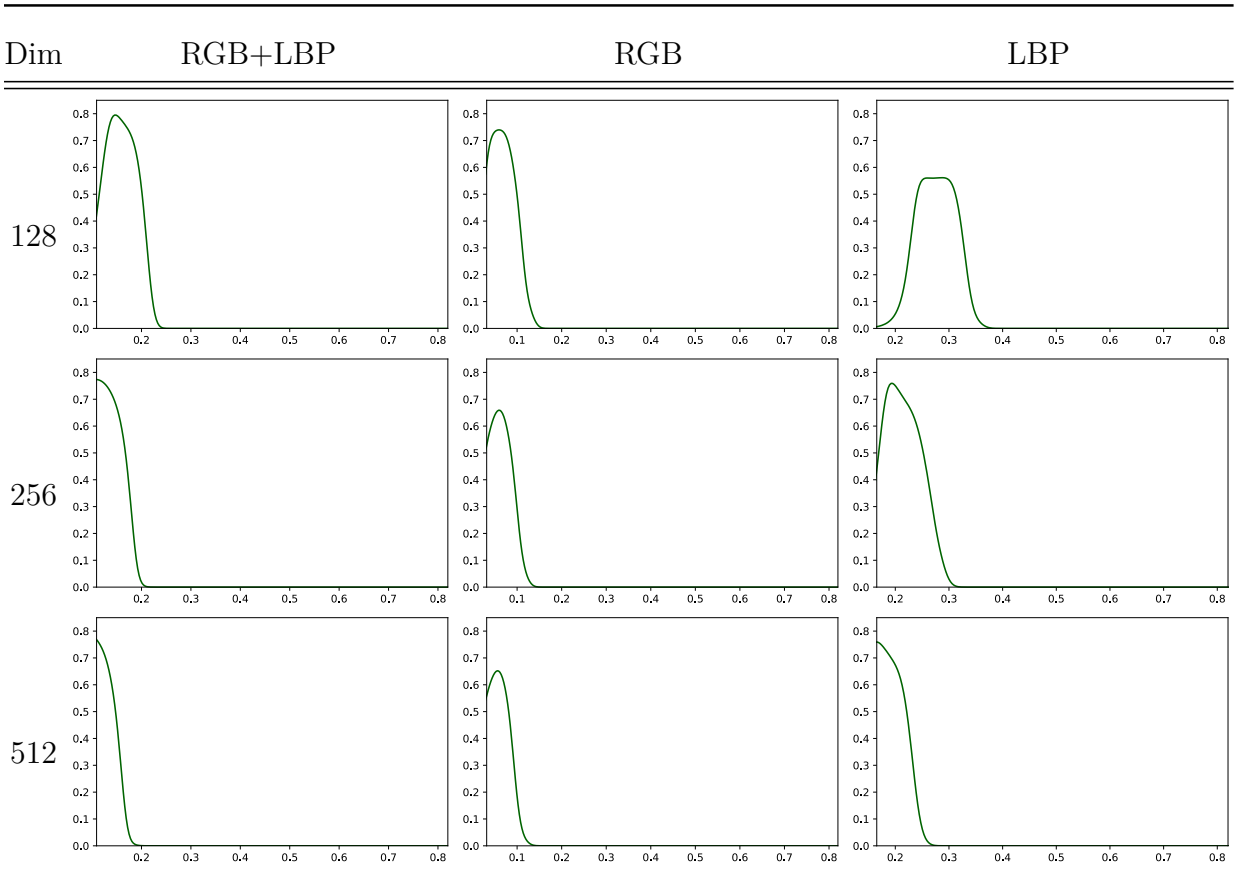
Figura 4.10: Función de densidad de probabilidad del error de reconstrucción MAE utilizando KDE. Dado que una entrada corresponde a café o a una anomalía, la función provee la probabilidad de obtener un determinado valor de MAE.



A partir de estos gráficos, se puede determinar que al incrementar el tamaño del espacio latente, la densidad de probabilidad se concentra en valores más bajos de MAE. Este resultado es esperable, puesto que el error de reconstrucción se calcula al comparar imágenes con mayor cantidad de detalles y texturas, tendiendo a ser un error más bajo. Los gráficos también permiten observar que efectivamente se tiene un traslape no deseado entre los datos de café y de anomalías. En el escenario RGB+LBP, las curvas presentan sus valores máximos más separados en comparación con los escenarios RGB y LBP, por lo que es considerado como el escenario que puede proveer los mejores resultados de predicción con los modelos del autocodificador β.

La figura 4.11 presenta la función de densidad de probabilidad $p(x = \text{café} \mid \text{MAE})$, obtenida a partir de los resultados de la figura 4.10 mediante el teorema de Bayes [70]. Dado que para esta etapa experimental se utiliza la misma cantidad de sub-imágenes de café y de sub-imágenes de anomalías, se asume una probabilidad simple de obtener café o anomalías igual al 50 %.

Figura 4.11: Función de densidad de probabilidad de café utilizando KDE. Dado cierto valor de error MAE, la función provee la probabilidad de que los datos correspondan a café.



Dicha función de densidad de probabilidad permite obtener la probabilidad de que una entrada sea café, dado un valor de error MAE determinado. En este caso, las mayores proba-

bilidades de que los datos sean de café, se obtienen con los modelos que poseen tamaños de espacio latente de 128 y 256 dimensiones, al utilizar sub-imágenes RGB+LBP con valores de error MAE menores a 0.2.

4.5. Gráficos de dispersión del espacio latente con UMAP

Al seleccionar los tres modelos del autocodificador convolucional β que proveen el mayor balance entre precisión y exhaustividad en los frentes de Pareto y evaluar su capacidad de reconstrucción de sub-imágenes, se implementa un método de visualización del espacio latente para cada modelo. Para esto, se utiliza el set de pruebas de la versión II del set de datos, compuesto por sub-imágenes de café y de anomalías, a partir de las cuales se realizan las predicciones.

Como método de exploración y visualización de los datos a la salida del codificador, se utiliza la versión supervisada del algoritmo UMAP. Al emplear este algoritmo como herramienta de visualización, es posible analizar la distribución de los datos en el espacio latente de los tres modelos del autocodificador convolucional β. Es decir, se generan gráficos de dispersión para los modelos con tamaños de espacio latente igual a 128, 256 y 512 dimensiones. Además, se evalúan diferentes configuraciones de parámetros de UMAP supervisado con cada uno de los tres modelos de autocodificador.

En la tabla 4.4 se presentan los parámetros utilizados en las diferentes configuraciones de UMAP. Para el número de vecinos, se utilizan únicamente tres valores, mientras que para la distancia mínima se emplean diez valores. En total, se generan treinta gráficos por cada modelo.

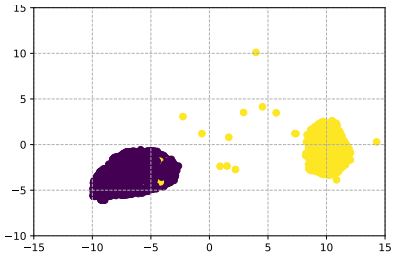
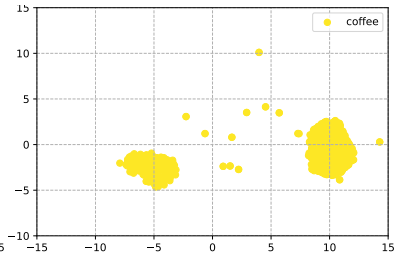
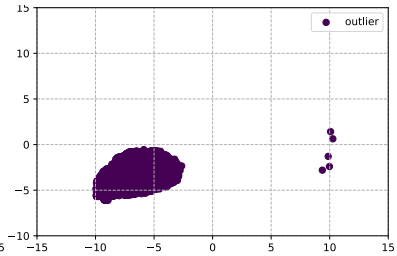
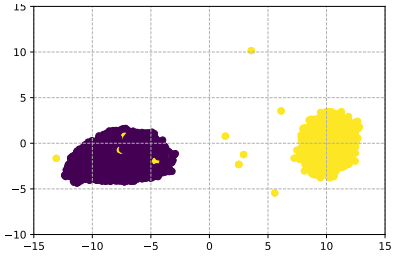
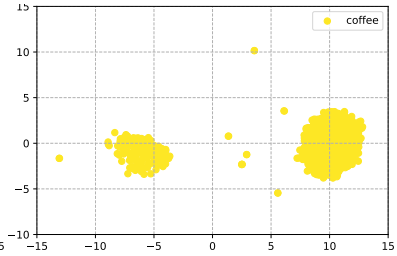
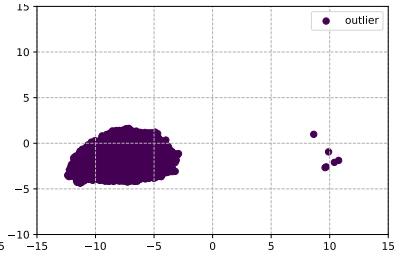
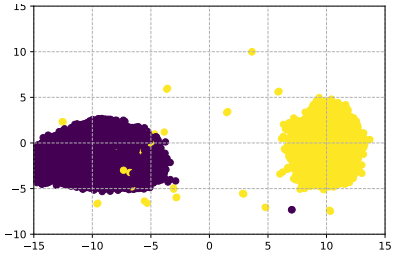
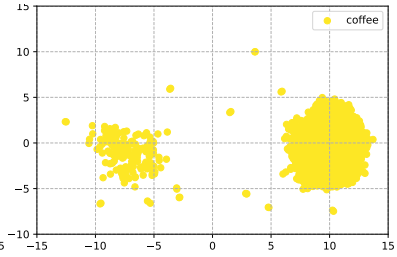
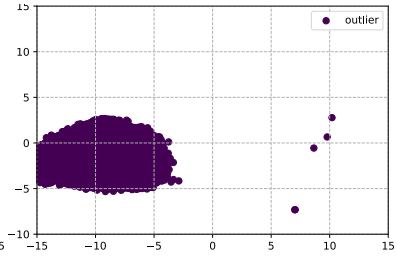
Tabla 4.4: Valores utilizados para los parámetros de UMAP supervisado.

Parámetro	Valor
Número de vecinos	$\{5, 8, 10\}$
Distancia mínima	$[0.1 - 1]$, cada 0.1
Número de componentes	2
Métrica	`'euclidean'`

Para ejemplificar el análisis realizado con el algoritmo UMAP supervisado, en la tabla 4.5 se muestran tres resultados de predicciones sobre el set de datos de café y anomalías, tomando la salida del codificador correspondiente al modelo con espacio latente de 256 dimensiones. Las figuras muestran la distribución de los datos en el espacio latente proyectado en 2 dimensiones al usar UMAP configurado con un número de vecinos igual a 8 y una distancia mínima

de 0.2, 0.4 y 0.8 respectivamente. En los gráficos, los puntos amarillos representan datos de café y los puntos morados las anomalías.

Tabla 4.5: Gráficos de dispersión para datos de café y anomalías con UMAP supervisado.

Dist	UMAP	Café	Anomalías
0.2			
0.4			
0.8			

Con el objetivo de analizar qué tan traslapados se encuentran ambos conjuntos de datos en el espacio bidimensional, también se muestran gráficos de dispersión para los conjuntos de café y anomalías por separado en la segunda y tercera columnas de la tabla 4.5. A partir de la información presentada en estos gráficos, es posible notar que existe un traslape importante entre ambos conjuntos de datos. En este caso, una parte de los datos de café se encuentra mezclada con los datos correspondientes a anomalías. A pesar de haber obtenido resultados utilizando todas las combinaciones de los parámetros presentados en la tabla 4.4, para los tres tamaños de espacio latente explorados se presenta dicho traslape.

Analizando el contenido de las sub-imágenes de anomalías representadas por los puntos que forman parte del traslape, se logra determinar que una gran mayoría de estas sub-imágenes corresponde a partes de otras plantas o de alguna otra región de los escenarios agrícolas capturados donde predomina el color verde. También se determina que las zonas donde predominan los puntos morados afuera del traslape, representan secciones de caminos, raíces, techos, autos e incluso otras plantas con características visibles muy diferentes a las características de las plantas de café, tal y como se ejemplifica en las anomalías de la figura 3.3.

4.6. Clasificación y ubicación de sub-imágenes en mosaicos parciales

Una vez analizado el comportamiento de las funciones de densidades de probabilidad, se utilizan clasificadores binarios ν-SVM en sub-imágenes de café y anomalías. Los datos empleados para entrenar y clasificar con las ν-SVM corresponden a la salida del codificador de cada uno de los tres modelos seleccionados del autocodificador convolucional β. Es decir, se utilizan datos de los espacios latentes de 128, 256 y 512 dimensiones.

La tabla 4.6 presenta los parámetros óptimos para la configuración de las ν-SVM, así como los resultados de exactitud obtenidos para cada tamaño de espacio latente, mediante el método de ajuste exhaustivo de parámetros *grid search*. En esta tabla, se observa que la exactitud aumenta conforme se utilizan datos de mayor dimensión. Sin embargo, la mejora sobre la exactitud obtenida usando el espacio latente de 512 dimensiones con respecto a los datos de 256 dimensiones, no es significativa como lo es con respecto a los datos de 128 dimensiones.

Tabla 4.6: Parámetros óptimos para ν-SVM según las dimensiones de los datos.

Dim	ν óptimo	γ óptimo	Exactitud
128	0.040	0.0010	0.86
256	0.035	0.0010	0.92
512	0.030	0.0014	0.93

Gracias a que se cuenta con las coordenadas de las sub-imágenes en los mosaicos originales de 4608×3456 pixeles, es posible generar las imágenes completas incluyendo las sub-imágenes clasificadas mediante las ν-SVM. Las sub-imágenes utilizadas en tareas de clasificación son extraídas originalmente de diferentes mosaicos, por lo que cada mosaico utilizado en esta etapa experimental contiene solamente algunas sub-imágenes clasificadas. Por esta razón, las imágenes se han denominado mosaicos parciales.

Para ejemplificar la generación de los mosaicos parciales de un escenario agrícola específico, la figura 4.12 presenta el mosaico parcial generado al agregar las sub-imágenes clasificadas a partir de datos de 512 dimensiones. En este mosaico se incluyen 131 sub-imágenes que fueron clasificadas correctamente en su totalidad. Las sub-imágenes clasificadas como café son agregadas de color verde y las clasificadas como anomalías de color rojo. El mosaico se presenta en escala de grises para facilitar la ubicación visual de las sub-imágenes clasificadas.

El proceso de clasificación de sub-imágenes con clases conocidas se aplica a 25 mosaicos parciales. Para este propósito, se utiliza parte del set de prueba de 11 008 sub-imágenes descrito en la tabla 3.5, siendo posible definir si una clasificación determinada representa

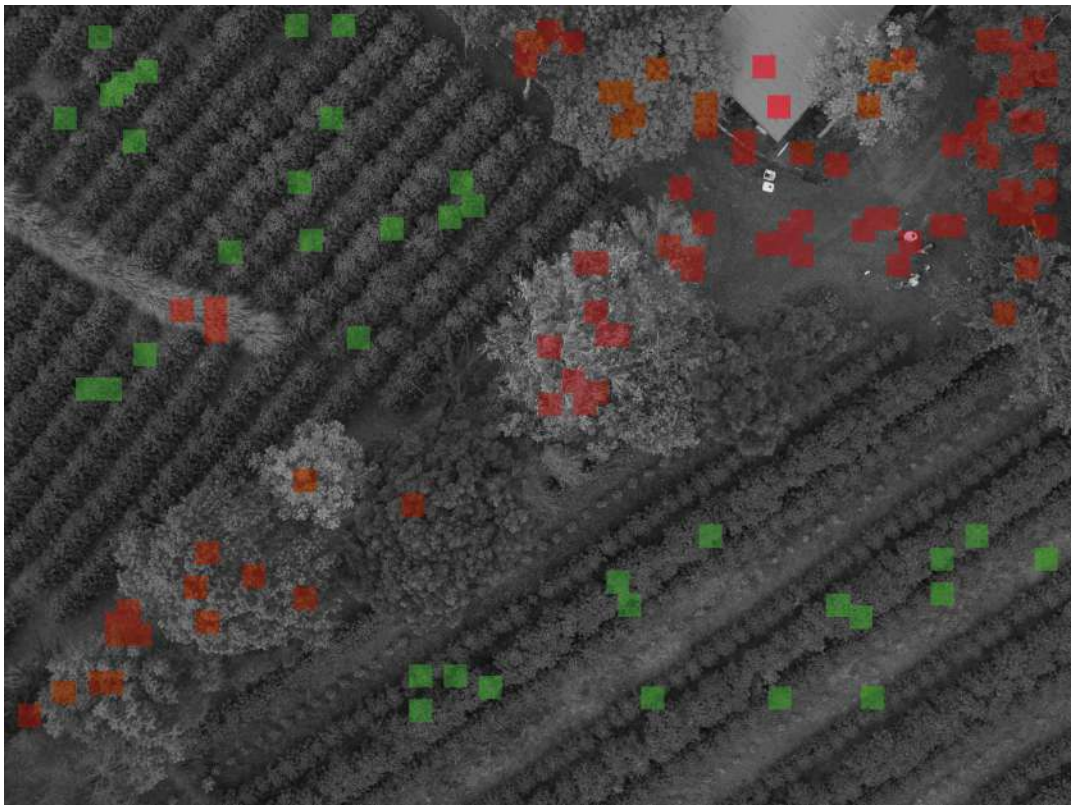


Figura 4.12: Mosaico parcial de un escenario agrícola compuesto por sub-imágenes clasificadas a partir de datos de 512 dimensiones. Un total de 131 sub-imágenes clasificadas.

un acierto o un desacierto. En total se clasifican 1508 sub-imágenes que forman parte de los 25 mosaicos utilizados en esta etapa experimental. La figura 4.13 presenta la cantidad de aciertos obtenidos durante la clasificación de sub-imágenes en los mosaicos parciales.

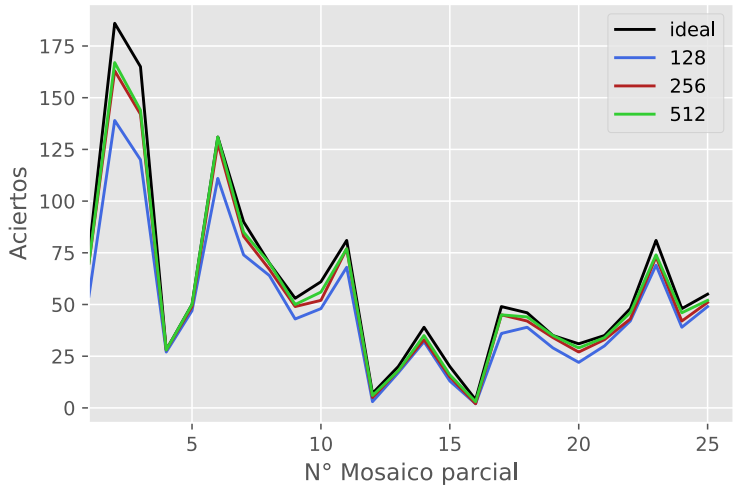


Figura 4.13: Aciertos obtenidos en la clasificación de sub-imágenes en 25 mosaicos parciales.

También se agrega el caso ideal como punto de referencia para comparar los tres casos de interés. Se observa que los datos de 512 dimensiones producen más o igual cantidad de aciertos que los datos de 256 y 128 dimensiones en todos los casos, estando más cerca del caso ideal. En la figura 4.14 se presenta la cantidad de desaciertos obtenidos en la clasificación de sub-imágenes de los mosaicos parciales. Se observa que la diferencia entre los datos de 512 y 256 dimensiones puede considerarse como mínima y ambos representan los casos más cercanos al caso ideal.

El análisis de los resultados cuantitativos y cualitativos en esta sección, permite identificar que una mayoría de desaciertos están asociados a partes de plantas. Este detalle confirma que en el traslape de ambos conjuntos de datos observado en la figura 4.5, predominan los puntos que representan partes de plantas, tanto de café como de árboles y arbustos.

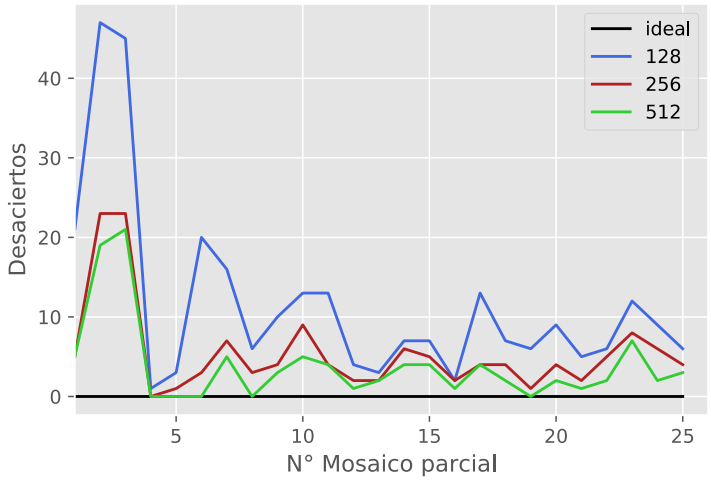


Figura 4.14: Desaciertos obtenidos en la clasificación de sub-imágenes en 25 mosaicos parciales.

En la tabla 4.7 se muestra la cantidad total de aciertos y desaciertos obtenidos al clasificar todas las sub-imágenes consideradas de los 25 mosaicos parciales, a partir de datos de 128, 256 y 512 dimensiones. También se incluye el caso ideal como información de referencia.

Tabla 4.7: Resultados de la clasificación de sub-imágenes en 25 mosaicos parciales.

Dim	Aciertos	Desaciertos
128	1217 (80.7 %)	291 (19.3 %)
256	1371 (90.9 %)	137 (9.1 %)
512	1411 (93.6 %)	97 (6.4 %)
Ideal	1508 (100 %)	0 (0 %)

4.7. Clasificación y ubicación de sub-imágenes en mosaicos completos

Además de ubicar las sub-imágenes clasificadas en los mosaicos parciales, se generan todas las sub-imágenes posibles a partir de un conjunto de mosaicos de interés. Por esta razón se denominan mosaicos completos. En esta sección, se presentan los resultados de clasificar todas las sub-imágenes que conforman los mosaicos en cuatro escenarios diferentes. En total se clasifican 1564 sub-imágenes por cada mosaico. En este caso, las sub-imágenes generadas no cuentan con ninguna clase asignada, por lo que los resultados son meramente cualitativos. Para cada escenario se generan los mosaicos utilizando clasificadores ν-SVM, entrenados con sub-imágenes obtenidas a partir de los datos de espacios latentes de 128, 256 y 512 dimensiones. Al ser un análisis cualitativo, se presentan los tres mosaicos resultantes por escenario para compararlos entre sí.

4.7.1. Primer escenario cafetalero

En la figura 4.15, se muestran los tres mosaicos completos del primer escenario, al clasificar las sub-imágenes a partir de datos de 128, 256 y 512 dimensiones respectivamente. Este primer escenario de interés corresponde a una captura realizada en la Zona de Los Santos, donde el mosaico original está conformado por sub-imágenes nuevas que no son utilizadas para entrenar los autocodificadores convolucionales ni los clasificadores ν-SVM. Además de la fecha y el escenario de captura, se tienen otros elementos como la iluminación, la altura del vuelo, el tipo de vegetación y el relieve, que cambian en este escenario con respecto a los escenarios utilizados para extraer las sub-imágenes empleadas en el entrenamiento de los autocodificadores y de los clasificadores.

(a) dim: 128 **(b)** dim: 256 **(c)** dim: 512

Figura 4.15: Mosaicos totales del primer escenario con sub-imágenes clasificadas a partir de datos de 128, 256 y 512 dimensiones.

En la figura 4.15b, se muestra una segunda versión del mosaico completo del primer escenario después de clasificar las sub-imágenes a partir de datos de 256 dimensiones. En este segundo

mosaico, se puede observar una mejoría en la clasificación de las plantas de banano y el camino como anomalías, con respecto a lo observado en el mosaico de la figura 4.15a. También es posible observar una mejoría en cuanto a la cantidad de sub-imágenes clasificadas por la ν-SVM como café. En la figura 4.15c, se presenta una tercera versión del mosaico completo del primer escenario al clasificar las sub-imágenes a partir de datos de 512 dimensiones. En este mosaico, se puede observar una mejoría en la clasificación de las plantas de banano y el camino como anomalías, con respecto a los dos mosaicos anteriores de este escenario. Sin embargo, también incrementó la cantidad de sub-imágenes correspondientes a café clasificadas erróneamente como anomalías.

En la tabla 4.8, se muestran los resultados de la clasificación de los tres mosaicos totales, donde se indica la cantidad de sub-imágenes clasificadas como café y como anomalías en cada mosaico. A partir de estos resultados cualitativos, se puede determinar que al aumentar el tamaño del espacio latente, también incrementa la información en los datos con la que cuenta el clasificador binario para realizar su tarea. Sin embargo, la adición de mayor detalle como texturas y sombras, también provoca un mayor número de clasificaciones erróneas de sub-imágenes que corresponden a plantas de café. El incremento de clasificaciones erróneas se atribuye en parte a los cambios en la información que poseen las sub-imágenes de café de este escenario con respecto a las sub-imágenes utilizadas para entrenar tanto los modelos de los autocodificadores convolucionales, como los clasificadores ν-SVM.

Tabla 4.8: Resultados de la clasificación de los mosaicos totales del primer escenario.

Dim	Café	Anomalías
128	977	587
256	1099	465
512	779	785

4.7.2. Segundo escenario cafetalero

En la figura 4.16, se muestran los mosaicos completos del segundo escenario al clasificar las sub-imágenes a partir de datos de 128, 256 y 512 dimensiones. En este segundo escenario se presenta un nuevo mosaico que también fue capturado en la Zona de Los Santos, donde se tiene ausencia total de plantas de café. En este escenario tan particular, se tiene únicamente una fila diagonal de plantas similares a café en la esquina inferior izquierda. Se tomó la decisión de utilizar este escenario, ya que interesa evaluar de forma cualitativa la capacidad de cada ν-SVM para clasificar como anomalías las regiones del mosaico que corresponden a terreno e incluso, las plantas que no corresponden a café. Es decir, el resultado ideal en este caso está dado por un mosaico pintado de color rojo en su totalidad.

(a) dim: 128 **(b)** dim: 256 **(c)** dim: 512

Figura 4.16: Mosaicos totales del segundo escenario con sub-imágenes clasificadas a partir de datos de 128, 256 y 512 dimensiones.

En la figura 4.16b, se muestra la segunda versión del mosaico completo del segundo escenario después de clasificar las sub-imágenes a partir de datos que poseen 256 dimensiones. En este segundo mosaico, se observa una mejoría en cuanto a la clasificación del terreno y de las plantas como anomalías, con respecto a los resultados representados en el mosaico de la figura 4.16a. En la figura 4.16c, se presenta la tercera versión del mosaico completo para el segundo escenario al clasificar las sub-imágenes a partir de datos de 512 dimensiones. En este mosaico, se puede observar una nueva mejoría notable en la clasificación del terreno y las plantas como anomalías. En este caso, se clasifican únicamente 7 sub-imágenes como plantas de café.

En la tabla 4.9, se observan los resultados obtenidos en la clasificación binaria de las sub-imágenes que conforman los tres mosaicos totales para este segundo escenario. En este caso, la cantidad de sub-imágenes clasificadas como anomalías aumenta conforme se incrementa el tamaño del espacio latente.

Tabla 4.9: Resultados de la clasificación de los mosaicos totales del segundo escenario.

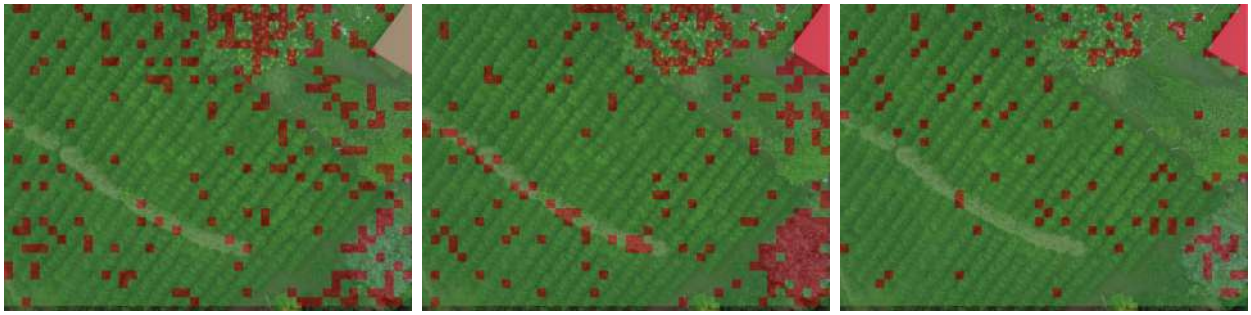
Dim	Café	Anomalías
128	131	1433
256	19	1545
512	7	1557

En este escenario en específico, se identifica que el uso de datos de 128 dimensiones produce los resultados más deficientes, al clasificar 131 sub-imágenes como café cuando el resultado correcto es equivalente a 0. A pesar de que la mayoría de sub-imágenes clasificadas como café utilizando datos de 128 dimensiones son de un color diferente al verde, el clasificador también considera información sobre la textura y produce resultados erróneos. La información de la textura es definida en parte por la mitad LBP de las sub-imágenes de entrada. Este resultado indica que la cantidad de información presente en los datos de 128 dimensiones,

no es suficiente para clasificar correctamente algunas anomalías que son identificadas al usar datos de mayores dimensiones.

4.7.3. Tercer escenario cafetalero

En la figura 4.17, se muestran los mosaicos completos del tercer escenario, el cual fue capturado en la finca del ICAFÉ en Santa Bárbara de Heredia, en el mismo día de la captura de los mosaicos de los cuales se extrajeron las sub-imágenes de entrenamiento para los autocodificadores convolucionales y los clasificadores binarios. En este escenario interesa identificar cuatro elementos principales dados por los árboles, la fila de plantas ornamentales de color claro, las plantas de café y una pequeña región trapezoidal que corresponde a una esquina de un techo.



(a) dim: 128 **(b)** dim: 256 **(c)** dim: 512

Figura 4.17: Mosaicos totales del tercer escenario con sub-imágenes clasificadas a partir de datos de 128, 256 y 512 dimensiones.

En la figura 4.17b, se presenta una segunda versión del mosaico completo correspondiente al tercer escenario, después de clasificar las sub-imágenes a partir de datos de 256 dimensiones. En este segundo mosaico, se puede observar una mejora significativa en cuanto a la clasificación del techo, los árboles y las plantas ornamentales como anomalías, con respecto a los resultados obtenidos en el mosaico de la figura 4.17a. Sin embargo, las clasificaciones erróneas de plantas de café como anomalías, son similares en los dos primeros mosaicos de este tercer escenario. En la figura 4.17c, se presenta una nueva versión del mosaico completo para el tercer escenario de interés, como resultado de clasificar las sub-imágenes a partir de datos que poseen 512 dimensiones. En este tercer mosaico, el resultado representa una clasificación deficiente de sub-imágenes que corresponden a anomalías, puesto que la cantidad de sub-imágenes que contienen partes de árboles y de plantas ornamentales que fueron clasificadas como café, aumenta con respecto al mosaico presentado en la figura 4.17b. Por lo tanto, en este caso específico los datos de 256 dimensiones producen mejores resultados que los datos de 512 dimensiones.

Este resultado permite evidenciar que no necesariamente un incremento en las dimensiones del espacio latente, implica una mejora en la clasificación de las sub-imágenes del mosaico. Por

lo tanto, la capacidad para clasificar correctamente mediante modificaciones a las dimensiones de los datos, es dependiente del escenario de interés y de las características que presenten sus elementos. En la tabla 4.10, se presentan los resultados obtenidos en la clasificación binaria de las sub-imágenes que conforman los tres mosaicos totales para el tercer escenario. En esta tabla, se logra identificar que el crecimiento de las sub-imágenes clasificadas como anomalías no es lineal con respecto a las dimensiones del espacio latente.

Tabla 4.10: Resultados de la clasificación de los mosaicos totales del tercer escenario.

Dim	Café	Anomalías
128	1328	236
256	1302	262
512	1427	137

4.7.4. Cuarto escenario cafetalero

En la figura 4.18, se presentan los mosaicos completos correspondientes al cuarto y último escenario. En este cuarto escenario, se muestra un mosaico que también fue capturado en la finca del ICAFÉ en Santa Bárbara de Heredia; pero en una fecha distinta al día de captura del mosaico presentado en el tercer escenario de interés. Este es un escenario complejo, ya que las condiciones de iluminación y las sombras son totalmente diferentes a las condiciones de los mosaicos anteriores. En este caso interesa identificar cinco regiones de elementos en el mosaico, dadas por el techo de la esquina inferior derecha, el invernadero, la cerca de plantas ornamentales y todo lo que abarca el sector de la calle a la izquierda del mosaico.

(a) dim: 128 **(b)** dim: 256 **(c)** dim: 512

Figura 4.18: Mosaicos totales del cuarto escenario con sub-imágenes clasificadas a partir de datos de 128, 256 y 512 dimensiones.

En la figura 4.18b, se puede observar una segunda versión del mosaico completo para este cuarto escenario, como resultado de clasificar las sub-imágenes a partir de datos de 256

dimensiones. En este segundo mosaico, se observa una mejora significativa en cuanto a la clasificación del techo, el invernadero, la cerca ornamental y los elementos de la calle como anomalías, con respecto a los resultados del mosaico mostrado en la figura 4.18a. Sin embargo, se puede identificar que persiste una gran cantidad de clasificaciones erróneas de sub-imágenes que corresponden a plantas de café. En la figura 4.18c, se presenta una última versión del mosaico completo para el cuarto escenario de interés, posterior a la clasificación de sub-imágenes a partir de datos que poseen 512 dimensiones. En este último mosaico, se observa un incremento significativo en la cantidad de sub-imágenes clasificadas como anomalías, tanto erróneas como correctas. A su vez, la cantidad de clasificaciones de sub-imágenes como café disminuye drásticamente.

En este caso, se logra evidenciar que las condiciones externas de iluminación y las sombras, representan factores que afectan directamente el contenido de los datos de baja dimensionalidad y por lo tanto, su respectiva clasificación. Para mitigar estos efectos no deseados, podría ser necesario entrenar las redes neuronales y los clasificadores con sub-imágenes que presenten diferentes condiciones de luz. Sin embargo, los escenarios cafetaleros de Costa Rica son tan complejos y variados, que representa un gran reto generar un set de imágenes que sea lo suficientemente completo y robusto, como para obtener clasificaciones correctas ante todas las condiciones externas posibles. La tabla 4.11, se presentan los resultados obtenidos durante la clasificación binaria de las sub-imágenes que conforman los tres mosaicos totales para el cuarto escenario.

Tabla 4.11: Resultados de la clasificación de los mosaicos totales del cuarto escenario.

Dim	**Café**	**Anomalías**
128	730	834
256	337	1227
512	153	1411

5 | Conclusiones

El desarrollo de este proyecto ha permitido proponer un algoritmo capaz de clasificar en imágenes aéreas de plantaciones costarricenses, las regiones que corresponden a plantas de café y las regiones consideradas como anomalías. El algoritmo emplea técnicas de aprendizaje automático clásico, así como técnicas de aprendizaje profundo para lograr dicha clasificación. La solución propuesta emplea una estrategia de detección de anomalías donde solamente es necesario contar con sub-imágenes cuyo contenido sea de plantas de café, permitiendo reducir el trabajo requerido para segmentar los mosaicos manualmente. Dicho trabajo manual representa un paso a realizar en métodos supervisados tradicionales.

También se ha construido un set de sub-imágenes extenso y depurado, que representa una herramienta a ser empleada en procesos experimentales donde se busque dotar de mejores capacidades al sistema clasificador de regiones. Sin embargo, se considera que dicho set debe extenderse aún más, incluyendo sub-imágenes capturadas bajo condiciones más variadas de iluminación, vegetación adicional, altura de vuelo y relieve.

Las etapas experimentales desarrolladas como parte de este proyecto, han permitido identificar un conjunto de oportunidades de mejora para obtener resultados más precisos en la clasificación de las regiones. El sistema propuesto presenta dependencia de condiciones ambientales específicas, que son requeridas durante la captura de las imágenes a utilizar en las predicciones con los autocodificadores convolucionales. Una oportunidad para mejorar este aspecto, corresponde a la inclusión de rotaciones, traslaciones y acercamientos como parte del aumento de datos en el proceso de entrenamiento de los modelos.

Se logró evidenciar que el tamaño del espacio latente representa un factor fundamental en la capacidad de reconstrucción de los autocodificadores, así como en los procesos de clasificación de sub-imágenes. Sin embargo, fue posible determinar que un incremento en el tamaño del espacio latente de los modelos, no garantiza una mejora en la clasificación de regiones para todos los escenarios agrícolas.

La inclusión de un descriptor de textura como LBP en los procesos de aprendizaje de los autocodificadores convolucionales, trajo mejoras en cuanto al nivel de detalle en la reconstrucción de las subimágenes y por lo tanto, en su clasificación como café o anomalías. Utilizando las versiones LBP de las sub-imágenes, fue posible determinar que la probabilidad de separar correctamente los datos correspondientes a café de las anomalías, incrementa cuando se utilizan

sub-imágenes RGB concatenadas horizontalmente con su versión LBP.

Como trabajo futuro, se propone repetir las etapas experimentales presentadas en este proyecto utilizando imágenes con más información capturada a partir de los ecosistemas agrícolas. Por ejemplo, utilizando imágenes multi-espectrales. Se considera que contar con información adicional, podría favorecer la separación correcta entre los datos de café y las anomalías, disminuyendo e incluso eliminando el traslape observado en los espacios latentes explorados.

También se propone la exploración de otros algoritmos de aprendizaje, donde se incluyan enfoques supervisados y semi-supervisados que favorezcan a la separación entre los datos que representan plantas de café y los datos correspondientes a otro tipo de plantas con características fenotípicas similares, que también forman parte de los ecosistemas cafetaleros costarricenses.

Referencias

[1] Jaime Gamboa, Yazmín Ross Lemus, and Luciano Capelli. *Café de Costa Rica: el Espíritu de una Nación*. Editorial ICAFÉ, 2014.

[2] Ximena Olmos. El comercio internacional como incentivo a la sostenibilidad: la experiencia de la red latinoamericana y del caribe de la huella ambiental del café. *Comisión Económica para América Latina*, 2020.

[3] Oscar Arias. Retos para la agricultura en costa rica. *Agronomía Costarricense*, 29(2):157–166, 2005.

[4] Andreas Kamilaris and Francesc X Prenafeta-Boldú. Deep learning in agriculture: A survey. *Computers and electronics in agriculture*, 147:70–90, 2018.

[5] Huasheng Huang, Jizhong Deng, Yubin Lan, Aqing Yang, Xiaoling Deng, and Lei Zhang. A fully convolutional network for weed mapping of unmanned aerial vehicle (uav) imagery. *PloS one*, 13(4):e0196302, 2018.

[6] Ulderico Neri and Rosa Francaviglia. Comparison of rusle model and uav-gis methodology to assess the effectiveness of temporary ditches in reducing soil erosion. In *GLOBAL SYMPOSIUM ON SOIL EROSION*, page 273, 2019.

[7] Delegados a la 50 Edición del Congreso Nacional Cafetalero Ordinario. Informe sobre la actividad cafetalera de costa rica. *Agronomía Costarricense*, Nov 2021.

[8] Raffaele Vignola, William Watler, Karina Poveda, and Armando Vargas. Prácticas efectivas para la reducción de impactos por eventos climáticos en el cultivo de café en costa rica. *Ministerio de Agrigultura y Ganadería*, 2019.

[9] FIDA FAO, PMA OMS, UNICEF, et al. El estado de la seguridad alimentaria y la nutrición en el mundo 2019. *Protegerse frente a la desaceleración y el debilitamiento de la economía. Roma: FAO*, 2019.

[10] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

[11] Liheng Zhong, Lina Hu, and Hang Zhou. Deep learning based multi-temporal crop classification. *Remote sensing of environment*, 221:430–443, 2019.

[12] Edna Chebet Too, Li Yujian, Sam Njuki, and Liu Yingchun. A comparative study of fine-tuning deep learning models for plant disease identification. *Computers and Electronics in Agriculture*, 161:272–279, 2019.

[13] Alvaro Fuentes, Sook Yoon, Sang Cheol Kim, and Dong Sun Park. A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. *Sensors*, 17(9):2022, 2017.

[14] Horea Mureşan and Mihai Oltean. Fruit recognition from images using deep learning. *Acta Universitatis Sapientiae, Informatica*, 10(1):26–42, 2018.

[15] Adam Umaltovich Mentsiev, Zelimkhan Abuevich Gerikhanov, and Albert Rashidovich Isaev. Automation and iot for controlling and analysing the growth of crops in agriculture. In *Journal of Physics: Conference Series*, volume 1399, page 044022. IOP Publishing, 2019.

[16] Michael I Jordan and Tom M Mitchell. Machine learning: Trends, perspectives, and prospects. *Science*, 349(6245):255–260, 2015.

[17] Adrian Carrio, Carlos Sampedro, Alejandro Rodriguez-Ramos, and Pascual Campoy. A review of deep learning methods and applications for unmanned aerial vehicles. *Journal of Sensors*, 2017, 2017.

[18] Weijia Li, Haohuan Fu, Le Yu, and Arthur Cracknell. Deep learning based oil palm tree detection and counting for high-resolution remote sensing images. *Remote Sensing*, 9(1):22, 2017.

[19] Steven W Chen, Shreyas S Shivakumar, Sandeep Dcunha, Jnaneshwar Das, Edidiong Okon, Chao Qu, Camillo J Taylor, and Vijay Kumar. Counting apples and oranges with deep learning: A data-driven approach. *IEEE Robotics and Automation Letters*, 2(2):781–788, 2017.

[20] Calvin Hung, Zhe Xu, and Salah Sukkarieh. Feature learning based approach for weed classification using high resolution aerial images from a digital camera mounted on a uav. *Remote Sensing*, 6(12):12037–12054, 2014.

[21] Julien Rebetez, Héctor F Satizábal, Matteo Mota, Dorothea Noll, Lucie Büchi, Marina Wendling, Bertrand Cannelle, Andrés Pérez-Uribe, and Stéphane Burgos. Augmenting a convolutional neural network with local histograms-a case study in crop classification from high-resolution uav imagery. In *ESANN*, 2016.

[22] Fidel Aznar, Mar Pujol, and Ramón Rizo. Visual navigation for uav with map references

using convnets. In *Conference of the Spanish Association for Artificial Intelligence*, pages 13–22. Springer, 2016.

[23] Peter Christiansen, Lars N Nielsen, Kim A Steen, Rasmus N Jørgensen, and Henrik Karstoft. Deepanomaly: Combining background subtraction and deep learning for detecting obstacles and anomalies in an agricultural field. *Sensors*, 16(11):1904, 2016.

[24] Fan Jiang, Junsong Yuan, Sotirios A Tsaftaris, and Aggelos K Katsaggelos. Anomalous video event detection using spatiotemporal context. *Computer Vision and Image Understanding*, 115(3):323–333, 2011.

[25] Sujata Butte, Aleksandar Vakanski, Kasia Duellman, Haotian Wang, and Amin Mirkouei. Potato crop stress identification in aerial images using deep learning-based object detection. *Agronomy Journal*, 113(5):3991–4002, 2021.

[26] Michael Gomez Selvaraj, Alejandro Vergara, Frank Montenegro, Henry Alonso Ruiz, Nancy Safari, Dries Raymaekers, Walter Ocimati, Jules Ntamwira, Laurent Tits, Aman Bonaventure Omondi, et al. Detection of banana plants and their major diseases through aerial images and machine learning methods: A case study in dr congo and republic of benin. *ISPRS Journal of Photogrammetry and Remote Sensing*, 169:110–124, 2020.

[27] Adriaan Jacobus Prins and Adriaan Van Niekerk. Crop type mapping using lidar, sentinel-2 and aerial imagery with machine learning algorithms. *Geo-spatial Information Science*, 24(2):215–227, 2021.

[28] Richard O Duda, Peter E Hart, et al. *Pattern classification and scene analysis*, volume 3. Wiley New York, 1973.

[29] Alexandre Pereira Marcos, Natan Luis Silva Rodovalho, and André R Backes. Coffee leaf rust detection using convolutional neural network. In *2019 XV Workshop de Visão Computacional (WVC)*, pages 38–42. IEEE, 2019.

[30] Alexandre J Oliveira, Gleice A Assis, Vitor Guizilini, Elaine R Faria, and Jefferson R Souza. Segmenting and detecting nematode in coffee crops using aerial images. In *International Conference on Computer Vision Systems*, pages 274–283. Springer, 2019.

[31] Rafael Baeta, Keiller Nogueira, David Menotti, and Jefersson A dos Santos. Learning deep features on multiple scales for coffee crop recognition. In *2017 30th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, pages 262–268. IEEE, 2017.

[32] Sankaran S, Khot L R, Espinoza C Z, Jarolmasjed S, Sathuvalli V R, Vandemark G J, Miklas P N, Carter A H, Pumphrey M O, Knowles N R, and Pavek M J. Low-altitude, high-resolution aerial imaging systems for row and field crop phenotyping: a review. *European Journal of Agronomy*, 70:112–123, 2015.

[33] Francisco J Yandún Narváez, Jaime Salvo del Pedregal, Pablo A Prieto, Miguel Torres-Torriti, and Fernando A Auat Cheein. Lidar and thermal images fusion for ground-based 3d characterisation of fruit trees. *Biosystems Engineering*, 151:479–494, 2016.

[34] Alexis Maizel, Daniel von Wangenheim, Fernán Federici, Jim Haseloff, and Ernst HK Stelzer. High-resolution live imaging of plant growth in near physiological bright conditions using light sheet fluorescence microscopy. *The Plant Journal*, 68(2):377–385, 2011.

[35] Stefan Paulus, Jan Behmann, Anne-Katrin Mahlein, Lutz Plümer, and Heiner Kuhlmann. Low-cost 3d systems: suitable tools for plant phenotyping. *Sensors*, 14(2):3001–3018, 2014.

[36] Kenta Itakura and Fumiki Hosoi. Estimation of leaf inclination angle in three-dimensional plant images obtained from lidar. *Remote Sensing*, 11(3):344, 2019.

[37] Wanneng Yang, Lingfeng Duan, Guoxing Chen, Lizhong Xiong, and Qian Liu. Plant phenomics and high-throughput phenotyping: accelerating rice functional genomics using multidisciplinary technologies. *Current opinion in plant biology*, 16(2):180–187, 2013.

[38] Santiago Andrés López. *Seguimiento de plantaciones de café a través de fotogrametría UAS*. Bachelor's thesis, Instituto Tecnológico de Costa Rica, 2019.

[39] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[40] Julie Gaertner, Vanessa Brooks Genovese, Christopher Potter, Kelvin Sewake, and Nicholas C Manoukis. Vegetation classification of coffea on hawaii island using worldview-2 satellite imagery. *Journal of Applied Remote Sensing*, 11(4):046005, 2017.

[41] Jefersson Alex dos Santos. A comparative study on unsupervised domain adaptation for coffee crop mapping. In *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications: 23rd Iberoamerican Congress, CIARP 2018, Madrid, Spain, November 19-22, 2018, Proceedings*, volume 11401, page 72. Springer, 2019.

[42] Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. Unsupervised domain adaptation with residual transfer networks. In *Advances in Neural Information Processing Systems*, pages 136–144, 2016.

[43] Otávio AB Penatti, Keiller Nogueira, and Jefersson A Dos Santos. Do deep features generalize from everyday objects to remote sensing and aerial scenes domains? In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 44–51, 2015.

[44] Renato O Stehling, Mario A Nascimento, and Alexandre X Falcão. A compact and efficient image retrieval approach based on border/interior pixel classification. In *Proceedings of the eleventh international conference on Information and knowledge management*, pages 102–109. ACM, 2002.

[45] Daniel B Mesquita, Ronaldo F dos Santos, Douglas G Macharet, Mario FM Campos, and Erickson R Nascimento. Fully convolutional siamese autoencoder for change detection in uav aerial images. *IEEE Geoscience and Remote Sensing Letters*, 2019.

[46] Gabriel LA Carrijo, Danilo E Oliveira, Gleice A de Assis, Murillo G Carneiro, Vitor C Guizilini, and Jefferson R Souza. Automatic detection of fruits in coffee crops from aerial images. In *2017 Latin American Robotics Symposium (LARS) and 2017 Brazilian Symposium on Robotics (SBR)*, pages 1–6. IEEE, 2017.

[47] Leif E Peterson. K-nearest neighbor. *Scholarpedia*, 4(2):1883, 2009.

[48] Mahesh Pal. Random forest classifier for remote sensing classification. *International Journal of Remote Sensing*, 26(1):217–222, 2005.

[49] Yann Le Cun, Lionel D Jackel, Brian Boser, John S Denker, Henry P Graf, Isabelle Guyon, Don Henderson, Richard E Howard, and William Hubbard. Handwritten digit recognition: Applications of neural network chips and automatic learning. *IEEE Communications Magazine*, 27(11):41–46, 1989.

[50] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

[51] Vladimir Golovko, Mikhno Egor, Aliaksandr Brich, and Anatoliy Sachenko. A shallow convolutional neural network for accurate handwritten digits classification. In *International Conference on Pattern Recognition and Information Processing*, pages 77–85. Springer, 2016.

[52] Abien Fred Agarap. Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375*, 2018.

[53] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[54] Mirza Cilimkovic. Neural networks and back propagation algorithm. *Institute of Technology Blanchardstown, Blanchardstown Road North Dublin*, 15, 2015.

[55] Jamie Hayes, Luca Melis, George Danezis, and ED Cristofaro. Logan: Evaluating information leakage of generative models using generative adversarial networks. *arXiv preprint arXiv:1705.07663*, 2017.

[56] Charles A Holt and Alvin E Roth. The nash equilibrium: A perspective. *Proceedings of the National Academy of Sciences*, 101(12):3999–4002, 2004.

[57] Chaoqun Hong, Jun Yu, Jian Wan, Dacheng Tao, and Meng Wang. Multimodal deep autoencoder for human pose recovery. *IEEE Transactions on Image Processing*, 24(12):5659–5670, 2015.

[58] Raghavendra Chalapathy and Sanjay Chawla. Deep learning for anomaly detection: A survey. *arXiv preprint arXiv:1901.03407*, 2019.

[59] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[60] Jinwon An and Sungzoon Cho. Variational autoencoder based anomaly detection using reconstruction probability. *Special Lecture on IE*, 2(1):1–18, 2015.

[61] Alireza Makhzani, Jonathon Shlens, Navdeep Jaitly, Ian Goodfellow, and Brendan Frey. Adversarial autoencoders. *arXiv preprint arXiv:1511.05644*, 2015.

[62] Alaa Tharwat. Classification assessment methods. *Applied Computing and Informatics*, 2020.

[63] William S Noble. What is a support vector machine? *Nature biotechnology*, 24(12):1565–1567, 2006.

[64] Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*, volume 4. Springer, 2006.

[65] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

[66] David I Spivak. Metric realization of fuzzy simplicial sets. *Preprint*, page 4, 2009.

[67] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

[68] Matti Pietikäinen. Image analysis with local binary patterns. In *Scandinavian Conference on Image Analysis*, pages 115–118. Springer, 2005.

[69] Nuh Alpaslan and Kazim Hanbay. Multi-resolution intrinsic texture geometry-based local binary pattern for texture classification. *IEEE Access*, 8:54415–54430, 2020.

[70] Richard Swinburne. Bayes'theorem. *Revue Philosophique de la France Et de l*, 194(2), 2004.