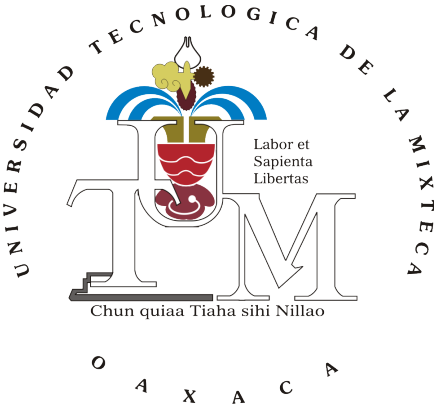


UNIVERSIDAD TECNOLÓGICA DE LA MIXTECA

Reconstrucción de entornos exteriores mediante imágenes aéreas y técnicas de Deep Learning

T E S I S

Para obtener el grado en:

DOCTOR EN ROBÓTICA

Presenta:

M. R. ARMANDO LEVID RODRÍGUEZ SANTIAGO

Directores de Tesis:

DR. JOSÉ ANÍBAL ARIAS AGUILAR
Universidad Tecnológica de la Mixteca, México
DR. ALBERTO ELÍAS PETRILLI BARCELÓ
Universidad de Tohoku, Japón

Huajuapan de León, Oaxaca, México, Septiembre del 2025

Tesis presentada el 12 de Septiembre de 2025
Directores de Tesis:

Dr. José Aníbal Arias Aguilar.
Universidad Tecnológica de la Mixteca, México.

Dr. Alberto Elías Petrilli Barceló.
Universidad de Tohoku, Japón.

A mi madre,

Este logro no es solo mío: es tuyo, mamá.
Es el fruto de tus desvelos, tus consejos, tu ternura
y tu entrega absoluta.

Gracias a Dios por haberme dado la bendición de ser tu hijo.
Gracias por enseñarme que, con amor y fe, todo es posible.
Tu amor ha sido una luz en mi camino, sosteniéndome firme en la noche más oscura,
y tu fe, la voz que me animó a seguir.

Gracias por cada sacrificio silencioso, por tus palabras de aliento,
por tus abrazos que curan y por tus oraciones que sostienen el alma.

Durante este camino lleno de desafíos y momentos difíciles,
nunca dejaste de creer en mí, ni siquiera cuando todo parecía derrumbarse.
Tú me enseñaste que la verdadera fortaleza se encuentra en el amor
y que el amor de una madre es el reflejo más puro del amor de Dios en la Tierra.

Gracias, Mamá.

Agradecimientos

A Dios, por concederme la dicha de culminar este proceso académico, por fortalecerme en los momentos de adversidad y por iluminar cada etapa de este camino con fe y esperanza.

A mis madres Alberta, Lourdes y Teresa, por su amor incondicional, su apoyo constante y los sacrificios silenciosos que han hecho a lo largo de mi vida. Su guía, entrega y fortaleza han sido pilares fundamentales en mi formación personal y profesional. Este logro es tanto mío como de ustedes.

A mis tíos y tías, por su acompañamiento, consejos y apoyo sincero durante mi vida académica. Sus palabras y su ejemplo me han inspirado a perseverar con disciplina y convicción. A toda mi familia, por ser fuente constante de alegría, afecto y motivación. Su presencia ha hecho de este trayecto una experiencia más llevadera y significativa.

A mis directores de tesis, el Dr. José Aníbal Arias Aguilar y el Dr. Alberto Elías Petrilli Barceló, por su valiosa orientación, compromiso y profesionalismo. Su paciencia, exigencia académica y experiencia fueron determinantes para la culminación de este trabajo.

A los integrantes del comité revisor, por sus observaciones y sugerencias, las cuales contribuyeron a mejorar la calidad del presente proyecto.

A mis colegas y amistades, con quienes compartí retos, aprendizajes y logros, les agradezco por su compañía y apoyo a lo largo de esta etapa. A mi amigo y compañero Omar, gracias por tu respaldo y acompañamiento. Una mención especial a Roberto, cuya memoria permanece viva; su amistad, generosidad y apoyo fueron determinantes en momentos clave de este camino.

A la Universidad Tecnológica de la Mixteca, por ser el espacio donde encontré crecimiento académico, intelectual y personal. A mis profesores, por su compromiso con la enseñanza y por sembrar en mí conocimientos y valores que me acompañarán siempre.

Al Consejo Nacional de Ciencia y Tecnología (CONACYT), por el apoyo económico otorgado durante mis estudios de posgrado, el cual resultó esencial para la realización de esta investigación.

Finalmente, a todas las personas que, de manera directa o indirecta, brindaron su tiempo, conocimientos, ánimo o compañía para la culminación de este proyecto académico, les expreso mi más profundo agradecimiento.

A todas y todos, mi más sincero agradecimiento.

Resumen

En este documento se presenta el enfoque a través de una arquitectura de Aprendizaje Profundo (Deep Learning) para la reconstrucción bidimensional y tridimensional en exteriores de terrenos en condiciones desafiantes mediante imágenes aéreas adquiridas por un vehículo aéreo no tripulado (UAV) comúnmente conocido como Dron. La arquitectura propuesta se configura con base en una arquitectura Autoencoder. Sin embargo, en lugar de las típicas capas convolucionales se proponen algunas diferencias para realizar el emparejamiento de imágenes. La etapa del Encoder se configura como una Red Residual o ResNet (del inglés: *Residual Network*) con cuatro bloques residuales mientras que, la etapa del Decoder se establece como una Red Generativa Adversaria o GAN (del inglés: *Generative Adversarial Networks*). Cada etapa ha sido dotada del conocimiento necesario para para extraer los mapas de características de imágenes aéreas de entornos exteriores y realizar el emparejamiento (en inglés: *stitching*) de éstas imágenes y obtener un ortomosaico con detalles en alta resolución. Por otra parte, para la reconstrucción tridimensional la arquitectura de red neuronal propuesta utiliza una etapa Encoder para la extracción del vector de características que describen la imagen de entrada mientras que el Decoder, llamado GAN-Decoder, genera una nube de puntos a partir de la información obtenida en la etapa anterior. Al proporcionar una secuencia de fotogramas con un porcentaje de superposición entre cada uno de ellos, es posible determinar la ubicación espacial de cada punto generado. Los experimentos muestran que, con esta propuesta es posible realizar una representación 3D de un área sobrevolada por un dron utilizando una nube de puntos generada con una arquitectura profunda que tiene como entrada una secuencia de imágenes aéreas 2D.

En comparación con otros trabajos, el sistema propuesto es capaz de realizar reconstrucciones tridimensionales de paisajes urbanos desafiantes. Comparado con los resultados obtenidos usando software comercial, nuestra propuesta fue capaz de generar reconstrucciones en menor tiempo de procesamiento, trabaja con un menor porcentaje de superposición entre las imágenes 2D de entrada y es invariable al tipo de trayectoria de vuelo establecida para el dron.

Abstract

This document presents an approach employing a Deep Learning architecture for the reconstruction of terrain in both two-dimensional and three-dimensional dimensions under challenging conditions. The proposed architecture is based on an Autoencoder architecture, with modifications to facilitate image pairing.

The Encoder stage is configured as a Residual Network (ResNet) comprising four residual blocks. The Decoder stage is established as an Adversarial Generative Network (GAN). Each stage is equipped with the requisite knowledge to extract characteristics from aerial images of outdoor environments and perform the matching (stitching) of these images to obtain an orthomosaic with high-resolution details.

For three-dimensional reconstruction, the proposed neural network architecture utilizes an Encoder stage to extract a feature vector describing the input image. The Decoder, known as the GAN-Decoder, generates a cloud of points from the information obtained in the preceding stage. By providing a sequence of frames with a percentage of overlap between each frame, the spatial location of each generated point is determined.

Experiments demonstrate that this proposal facilitates the creation of a three-dimensional representation of an area flown over by a drone using a cloud of points generated by a deep architecture that processes a sequence of two-dimensional aerial images.

In comparison to other works, the proposed system demonstrates the capability to perform three-dimensional reconstructions of intricate urban landscapes. Notably, it surpasses the results obtained using commercial software by generating reconstructions in a shorter processing time. Additionally, it operates with a lower percentage of overlap between input two-dimensional images and exhibits invariance to the flight path established for the drone.

Índice

Índice de figuras

Índice de Tablas

Capítulo 1

Introducción

La reconstrucción bidimensional y tridimensional, como representación visual de un objeto o una zona de interés, es un tema ampliamente estudiado y de gran utilidad en diversas aplicaciones, como el reconocimiento de objetos y la comprensión de escenas. Este proceso involucra técnicas de fotogrametría, destacando la fotogrametría aérea, que permite determinar las propiedades geométricas de un terreno y las ubicaciones espaciales a partir de imágenes fotográficas capturadas desde el aire, generalmente mediante un vehículo aéreo no tripulado (UAV, por sus siglas en inglés: *Unmanned Aerial Vehicle*), comúnmente conocido como dron.

El resultado de esta técnica es un mapa u ortofotografía que ofrece información actualizada de la zona de interés, proporcionando una representación visual precisa de la superficie terrestre en una única fotografía. Esta imagen permite apreciar con detalle todos los elementos presentes en la superficie, con una exactitud comparable a la de un plano cartográfico.

Por otro lado, las técnicas de fotogrametría también permiten generar modelos tridimensionales del terreno o la zona de interés sobrevolada por un dron. Los algoritmos de reconstrucción 3D de última generación (Ren et al., 2019; Schonberger and Frahm, 2016) han logrado avances significativos, proponiendo soluciones a problemas como la Estructura a partir de Movimiento (SfM, por sus siglas en inglés: *Structure from Motion*) (Beardsley et al., 1997; Häming and Peters, 2010; Mouragnon et al., 2009) y la Localización y Mapeo Simultáneo (SLAM, por sus siglas en inglés: *Simultaneous Localization And Mapping*) (Cadena et al., 2016; Carlone et al., 2015; Pollefeys et al., 2004). Estas técnicas han mostrado resultados prometedores al abordar estos desafíos, combinando sensores activos y pasivos.

Sin embargo, las técnicas actuales de reconstrucción 3D no están diseñadas específicamente para escenarios exteriores con imágenes aéreas. Este tipo de entornos presenta desafíos únicos, ya que las condiciones pueden ser variables y la información disponible suele ser limitada. Además, la naturaleza de las imágenes aéreas puede generar correspondencias ambiguas entre píxeles y puntos espaciales 3D, lo que complica la proyección de 2D a 3D (Golparvar-Fard et al., 2011; Zhang et al., 2019b).

Por otra parte, los modelos tradicionales para la generación de modelos tridimensionales a menudo no logran producir coincidencias confiables en regiones con patrones repetitivos, apariencia homogénea o cambios significativos de iluminación. Este problema, característico de la fotogrametría, ha sido ampliamente documentado (Nikolakopoulos et al., 2017; Rothermel et al., 2012).

El problema se vuelve más desafiante cuando se trabaja con imágenes aéreas de ambientes exteriores. Sin embargo, gracias a los avances actuales en Deep Learning, es posible emplear diversas arquitecturas para obtener resultados similares o incluso superiores a las técnicas clásicas de generación de modelos tridimensionales, combinando configuraciones avanzadas de redes neuronales profundas. Por ejemplo, mediante el uso de grandes conjuntos de datos existentes y la fusión de sensores como cámaras estéreo y LiDAR 3D activo (Cheng et al., 2020; Choe et al., 2021; Wang et al., 2019a), es factible realizar estimaciones precisas de profundidad a largo alcance y llevar a cabo reconstrucciones 3D de alta calidad.

En este trabajo, nuestro principal objetivo es realizar reconstrucciones bidimensionales y tridimensionales de una zona de interés a partir de imágenes aéreas 2D capturadas por un UAV. Debido a las complejas condiciones orográficas del estado de Oaxaca, México, resulta particularmente difícil obtener datos de ciertas regiones. Por ello, un modelo tridimensional podría proporcionar información clave sobre áreas de interés específicas. En particular, las condiciones del terreno en la Universidad Tecnológica de la Mixteca, ubicada en el altiplano del estado de Oaxaca, presentan variaciones significativas en la altura y extensas áreas con vegetación homogénea. Un modelo tridimensional basado en nubes de puntos, generado exclusivamente a partir de imágenes 2D mediante una arquitectura de red neuronal profunda, permitiría obtener información detallada y relevante sobre estas áreas universitarias.

La reconstrucción 3D precisa del campus es importante para varios proyectos, que van desde la expansión de la infraestructura hasta la recuperación de agua en los techos y la ubicación de las plantas de energía verde, pasando por los recorridos virtuales propuestos a los nuevos estudiantes. Con una superficie de más de 104 hectareas y muchos edificios altos en el campus, necesitamos el apoyo de imágenes aéreas para abarcar la dimensión de la universidad y con la información brindada desde diferentes perspectivas, realizar una reconstrucción digital.

La motivación que impulsa esta investigación radica en que, mientras las propuestas del estado del arte actual tienden a emplear técnicas clásicas de visión por computadora y combinan información de sensores complementarios como LiDAR para generar modelos tridimensionales (Cheng et al., 2020; Choe et al., 2021; Wang et al., 2019a), nuestra propuesta plantea una solución alternativa. Específicamente, proponemos generar modelos bidimensionales y tridimensionales a partir de imágenes 2D aéreas capturadas en entornos exteriores no estructurados, reemplazando las técnicas tradicionales de visión por arquitecturas basadas en redes neuronales profundas. Esto promete una solución más eficiente en comparación con otros enfoques y algunos softwares comerciales.

Para lograr esto, proponemos el uso de una arquitectura basada en Autoencoders (Makhzani et al., 2015; Pinaya et al., 2020; Zamorski et al., 2020) y Redes Generativas Adversarias (GANs, por sus siglas en inglés: *Generative Adversarial Networks*) (Creswell et al., 2018; Goodfellow et al., 2020; Metz et al., 2016). Estas arquitecturas procesarán una secuencia de imágenes aéreas 2D como entrada, para producir como salida una reconstrucción tridimensional basada en nubes de puntos.

1.1. Planteamiento del Problema

La Universidad Tecnológica de la Mixteca (UTM) cuenta con un campus de 104 hectáreas, de las cuales más de la mitad son zonas que se encuentran sin construcción y de difícil acceso. Por tal motivo, contar con una ortofotografía y un modelo o representación tridimensional del lugar permiten tener un conocimiento vigente de los datos y de las diferentes áreas de la Universidad.

Tradicionalmente, este tipo de modelos se realizan mediante el uso de complejas técnicas de visión por computadora y herramientas o sensores costosos tales como los Sistemas de Posicionamiento Global (GPS) de exactitud o sensores LiDAR (del inglés: *Light Detection and Ranging o Laser Imaging Detection and Ranging*) y licencias de software especializado. Sin embargo, en los últimos años los avances tecnológicos en drones, cámaras digitales y ordenadores con mayor capacidad computacional han ayudado a simplificar las tareas a realizar y a reducir costos que esto implica. No obstante, los costos económicos referentes a licencias mensuales de software siguen siendo difíciles de solventar.

En este contexto, la División de Estudios de Posgrado de la UTM, en particular los laboratorios de robótica, cuentan con un vehículo aéreo no tripulado (ver figura 1.1) el DJI Mavic Pro. Una aeronave de diseño compacto y plegable con cámara y estabilizador que consta de un controlador de vuelo, transmisión de vídeo, un sistema de propulsión y una batería de vuelo inteligente; entre otras características de las que se destaca el poder capturar vídeos 4K y fotos de 12 megapíxeles, alcanzando una velocidad de vuelo máxima de 65 Km/h (40 mph) y un tiempo máximo de vuelo de 27 minutos. Gracias a su diseño compacto y la calidad de la cámara que posee, es idóneo para el monitoreo y vigilancia de zonas remotas. Por lo tanto, al contar con este tipo de plataformas robóticas, se propone la reconstrucción bidimensional y tridimensional digital de superficies en entornos exteriores en zonas del terreno transitado por las trayectorias del dron a partir de imágenes aéreas y la aplicación de técnicas de Deep Learning, así como el uso del servidor de alto rendimiento computacional para el entrenamiento de modelos de redes neuronales profundas.

Figura 1.1: DJI Mavic Pro. UAV portátil de diseño compacto con cámara de captura de vídeos 4K y fotos de 12 megapíxeles.

1.2. Justificación

La reconstrucción digital ofrece un panorama detallado y preciso que resulta fundamental para el diseño, la construcción y la toma de decisiones en proyectos como la reforestación y la planeación urbana, entre otros. En este contexto, contar con una ortofotografía del terreno se convierte en una herramienta de gran relevancia.

Por esta razón, la realización de este trabajo contribuirá a la reconstrucción digital del campus de la Universidad Tecnológica de la Mixteca, incluyendo tanto zonas construidas como áreas sin construcción y de de difícil acceso. Esto será de gran utilidad para la planificación de proyectos de reforestación y edificación dentro de la institución. A diferencia de las soluciones comerciales, esta propuesta optimiza el tiempo, la capacidad de procesamiento y los costos. Además, se espera que esta investigación fomente el interés por el estudio y la aplicación de técnicas de aprendizaje profundo en proyectos de impacto real.

1.3. Hipótesis

A partir de una serie de fotografías aéreas tomadas por un UAV y arquitecturas profundas de redes neuronales, es posible la reconstrucción digital bidimensional y tridimensional de una superficie terrestre en exteriores no estructurada.

1.4. Objetivos

1.4.1. Objetivo General

Desarrollar una arquitectura de Deep Learning para la generación de un modelo bidimensional que permita obtener una ortofotografía, así como un modelo tridimensional basado en nubes de puntos de diversas regiones del campus de la Universidad Tecnológica de la Mixteca, utilizando como insumo una serie de imágenes aéreas capturadas por un Vehículo Aéreo No Tripulado (UAV).

1.4.2. Objetivos Específicos

Para conseguir el objetivo principal se presentan los siguientes objetivos específicos.

1. Generar una base de datos de imágenes aéreas de la UTM.
2. Seleccionar y utilizar un modelo de red neuronal profunda que ayude en la obtención de información (mapas de características) de imágenes aéreas.
3. Ajustar el modelo de red neuronal pre-entrenado con la nueva base de datos de imágenes aéreas.
4. Correlacionar los mapas de características de cada par de imágenes utilizado para la reconstrucción bidimensional digital de la zona de interés, con el modelo de red neuronal ajustado.

5. Reconstruir en alta resolución la zona de interés mediante mapas de características.

6. Ajustar el modelo de red neuronal pre-entrenado con la nueva base de datos de imágenes aéreas para la generación de nubes de puntos tridimensionales.

7. Extraer mapas de características para la generación de nubes de puntos tridimensionales.

8. Correlacionar los puntos puntos tridimensionales.

9. Reconstruir un modelo tridimensional de la zona sobrevolada por el UAV mediante nubes de puntos tridimensionales generadas a partir del modelo de red neuronal profundo ajustado.

10. Comparar el modelo tridimensional generado con el obtenido por un software comercial.

1.5. Alcances

El trabajar con técnicas de Deep Learning implica la colaboración de un equipo interdisciplinario de investigadores y el uso de infraestructura computacional especializada para el manejo de grandes volúmenes de datos, el cual es esencial para garantizar el desempeño óptimo de las redes neuronales profundas. Si bien hoy en día se dispone de una gran cantidad de datos para la investigación, no toda esta información es adecuada para resolver problemas específicos, por lo que, en muchos casos, es necesario generar bases de datos propias para abordar el problema en cuestión.

En este contexto, para este proyecto será necesario generar una base de datos de imágenes aéreas. Estas imágenes serán bidimensionales, a color, y capturadas mediante una única cámara montada en el UAV. La base de datos estará compuesta por imágenes de resolución y compresión uniformes, obtenidas a diferentes alturas de vuelo considerando las condiciones del terreno sobrevolado y que sean seguras para la aeronave.

En cuanto al tipo de red neuronal profunda a emplear, se utilizarán arquitecturas conocidas como las Redes Neuronales Convolucionales (CNNs, por sus siglas en inglés: *Convolutional Neural Networks*), Redes Generativas Adversarias (GANs, por sus siglas en inglés: *Generative Adversarial Networks*) y Autoencoders (AEs, por sus siglas en inglés: *AutoEncoders*). Las cuales, servirán de base para poder desarrollar una arquitectura propia enfocada en poder optimizar la reconstrucción bidimensional y tridimensional en el contexto planteado.

1.6. Organización del documento

Este documento se encuentra organizado de la siguiente manera:

En el Capítulo 2 se presentan los trabajos relacionados con el tema de investigación en cuestión. Se revisan tanto reconstrucciones bidimensionales como tridimensionales, exponiendo los métodos más relevantes desarrollados para resolver este tipo de problemáticas. Además, se incluyen investigaciones enfocadas en el Deep Learning que abordan desafíos similares.

En el Capítulo 3 se introducen los tópicos teóricos necesarios para fundamentar la solución propuesta. Entre los temas destacados se encuentran conceptos sobre inteligencia artificial y su relación con el Deep Learning, así como la composición y estructura de las redes neuronales profundas. También se aborda la fotogrametría, un área en la que el aprendizaje profundo ha comenzado a incursionar, presentando diversas áreas de oportunidad para trabajar con enfoques basados en inteligencia artificial.

El Capítulo 4 describe detalladamente el desarrollo de la metodología propuesta para la reconstrucción utilizando un modelo de Redes Neuronales Convolucionales (CNN). Se abordan aspectos como la concepción de la solución, la metodología empleada y la generación de la base de datos necesaria para la arquitectura de red a utilizar.

La generación de modelos bidimensionales, tanto en baja como en alta resolución, se describe detalladamente en el Capítulo 5. Por otro lado, la generación del modelo tridimensional mediante nubes de puntos de entornos exteriores se describe en el Capítulo 6.

En el Capítulo 7 se presentan los resultados obtenidos a partir de la implementación de las redes neuronales profundas propuestas. Se muestran los resultados de las reconstrucciones bidimensionales realizadas en diferentes condiciones de vuelo, como la altura, el tipo de trayectoria seguida por el dron durante la captura de información, y el porcentaje de superposición entre imágenes capturadas. Además, se incluyen los resultados de las reconstrucciones tridimensionales, generadas a partir de nubes de puntos mediante la arquitectura de red neuronal propuesta.

Finalmente, en el Capítulo 8 se presentan las conclusiones generales de este trabajo. Se analiza la robustez y eficiencia de la arquitectura propuesta para enfrentar condiciones desafiantes, como las que presentan los terrenos del campus de la Universidad Tecnológica de la Mixteca. Asimismo, se identifican áreas de oportunidad y posibles mejoras que pueden ser desarrolladas en trabajos futuros.

Capítulo 2

Estado del Arte

En la navegación autónoma de Vehículos Aéreos No Tripulados (UAV, por sus siglas en inglés: *Unmanned Aerial Vehicle*), resolver las tareas de localización, navegación y aterrizaje seguro de estas plataformas robóticas es de vital importancia (Anitha and Kumar, 2012; Conte and Doherty, 2008; Saripalli et al., 2002). La aplicación de un sistema de visión por computadora es fundamental para abordar estas tareas, utilizando diversas técnicas. Una de ellas es el registro de imágenes, un proceso esencial y ampliamente utilizado para obtener datos sobre las zonas o terrenos de operación de la plataforma robótica utilizada.

Para resolver y realizar una descripción general, rápida y precisa de un área de interés, es crucial contar con herramientas efectivas que puedan asistir en el proceso. En particular, los Modelos Digitales de Superficie (DSM, por sus siglas en inglés: *Digital Surface Models*) y los ortomosaicos, que son representaciones de mapas compuestos por varias ortofotos unidas entre sí para formar una imagen panorámica de la zona de interés. En visión por computadora, esto se logra mediante la sobreposición de imágenes, un proceso conocido como ***stitching***. Estas herramientas son esenciales para obtener un conocimiento detallado de zonas de interés, especialmente cuando estas áreas son de difícil acceso para un operador humano.

Un ortomosaico proporciona una visión general amplia del entorno y facilita que un operador humano pueda inspeccionar regiones de interés a través de imágenes aéreas. Esto permite tomar decisiones informadas sobre las zonas de interés mediante el análisis y procesamiento digital de las imágenes aéreas (Ahrens et al., 2009).

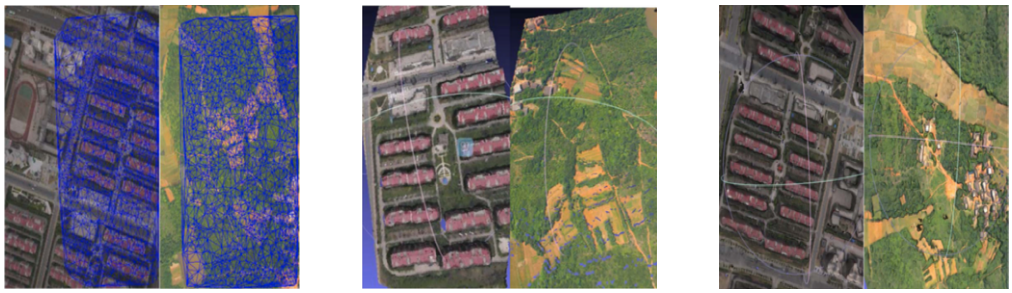
Sin embargo, la generación de un ortomosaico es una tarea compleja. Por ello, se han desarrollado numerosas investigaciones para abordar su generación utilizando diversas técnicas y procesos. Investigaciones como la de Zhang and Li (2014) presentan una metodología para el reensamblaje de fragmentos de imágenes, mientras que Lingua et al. (2009a) describen un procedimiento para la generación de un DSM. Estos enfoques aprovechan tanto las técnicas de visión por computadora como los algoritmos de coincidencia de múltiples imágenes para la extracción de puntos y bordes de las imágenes, y su posterior coincidencia para generar DSMs. Este procedimiento permite obtener resultados confiables en términos de puntos extraídos, exactitud de ubicación y rapidez en la producción de resultados, incluso en condiciones geométricas difíciles.

El método propuesto en este trabajo debe ser rápido, confiable y asegurar un buen rendimiento en la generación automática de Modelos Digitales de Superficie (DSM) utilizando técnicas fotogramétricas. No debe requerir conocimiento previo de un conjunto de puntos, y se espera que funcione rápidamente, descartando errores graves de forma completamente automática. El rendimiento del algoritmo depende principalmente de las características geométricas de los objetos presentes en las escenas a procesar, lo que puede afectar su eficiencia.

Yeu et al. (2006) presentan el paradigma de aprendizaje extremo ELM (del inglés: *Extreme Learning Machine*) para el acceso a la información de multiresolución en la determinación de la alta resolución necesaria para la navegación de los UAV, basándose en un clasificador de red neuronal (NN por sus siglas en inglés: *Neural Networks*). Con este paradigma, los autores demuestran que es posible reducir significativamente el tiempo necesario para entrenar una red neuronal. El algoritmo utiliza parámetros de neuronas ocultas inicializados aleatoriamente e iterativamente calcula un vector de ponderación de salida. El número de neuronas ocultas es considerablemente menor que el número de muestras de entrenamiento, lo que permite que los parámetros de las neuronas ocultas no necesiten ser sintonizados y puedan generarse aleatoriamente con cualquier distribución de probabilidad continua. Esta consideración especial de los parámetros de la red lleva al entrenamiento a simplemente encontrar una solución de mínimos cuadrados para la matriz de salida de la red en un sistema lineal. Los algoritmos de entrenamiento de redes neuronales (***NN***) optimizan iterativamente los vectores de ponderación de las capas de la red, lo que puede resultar en largos tiempos de entrenamiento y posiblemente hacer que una NN sea la opción menos atractiva para calcular representaciones de mapas del entorno.

Li et al. (2014) presentan un enfoque para la generación digital de Modelos de Superficie (DSM) a partir de imágenes de alta resolución provenientes de un UAV. Este método consta de varias etapas, comenzando con la extracción de mapas de características (feature maps) mediante el algoritmo SIFT (del inglés: Scale Invariant Features Transform) y el algoritmo de Poisson, que se basa en los gradientes de la intersección de los sólidos. Es decir, el algoritmo se basa en la orientación de un conjunto de puntos orientados, lo cual permite obtener una malla mediante la transposición de los puntos orientados a un campo vectorial continuo, donde los gradientes coinciden mejor con el campo vectorial. La Figura 2.1 muestra el resultado de la aplicación de este algoritmo. Este enfoque demuestra ser rápido y robusto en la extracción de puntos característicos; sin embargo, aún se busca mejorar la optimización del algoritmo de forma paralela y adaptarlo de manera eficiente a otros problemas.

De manera similar, Chau and Karol (2014) presentan una propuesta basada en características locales e invariantes de escala para la coincidencia de imágenes, logrando obtener imágenes panorámicas tanto de escenas interiores como exteriores. El método es robusto frente al cambio de escala, orientación y variaciones de iluminación entre imágenes. Sin embargo, el método es sensible al aumento considerable de coincidencias de puntos clave en una escena, como en el caso de multitudes, gran cantidad de oclusiones o una calidad deficiente de las imágenes, lo que es un problema común cuando se trabaja con imágenes de entornos exteriores. Por lo tanto, Bu et al. (2016) desarrollan una propuesta basada en la generación de nubes de puntos 3D a partir de imágenes (ver Figura 2.2)



(a) (b) (c)

Figura 2.1: Comparativa de los resultados obtenidos en el algoritmo propuesto por Li et al. (2014). (2.1a) Muestra la representación 2D mediante triangulación de Delaunay, la imagen (2.1b) muestra los puntos 3D obtenidos después del filtrado de valores atípicos, mientras que en (2.1c)se muestran dos DSM generados utilizando el método propuesto por los autores con dos conjuntos de datos.

El método consiste en cinco partes principales:

1. Adquisición de video en tiempo real y pre procesamiento de la información visual es decir, la eliminación de la distorsión en las imágenes, extracción de puntos característicos para realizar el seguimiento de ellos.

2. Seguimiento de puntos característicos y actualización de los mismos.

3. Mapeo: Se agrega nueva información.

4. Optimización de mapas, detectar y cerrar bucles para que se pueda obtener un mapa aplicando una función de similitud.

5. La fusión del mapa.

Los resultados (ver figura 2.2) demuestran que se puede lograr la reconstrucción del mapa con alta eficiencia y calidad. Además, es posible mostrar los resultados en forma inmediata incluso en terrenos no planos.

Tradicionalmente, los mapas de características (*feature maps*) han sido la herramienta fundamental en aplicaciones de visión por computadora, tanto en la recuperación como en el emparejamiento de imágenes. En este contexto, Altwaijry et al. (2016) presentan un enfoque para aprender a detectar y describir puntos clave (*key points*) de imágenes mediante arquitecturas profundas (ver Figura 2.3). Para este enfoque basado en aprendizaje automático, se emplea un conjunto de datos de parches a gran escala con puntos clave multiescala coincidentes. El modelo de red neuronal propuesto aprende de este conjunto de datos para identificar y describir puntos clave significativos.

Figura 2.2: Se muestran los resultados del algoritmo propuesto por (Bu et al., 2016). La combinación de fotogramas no es lo suficientemente suave en sus uniones, aunque algunos pueden llegar a ser aceptables o incluso de alta calidad en áreas con poca superposición. Sin embargo, se requieren horas para alinear las reconstrucciones bidimensionales. A pesar de esto, el método utiliza coordenadas GPS para estimar conjuntamente la pose de la cámara, lo que lo hace dependiente de información complementaria obtenida de otros sensores.

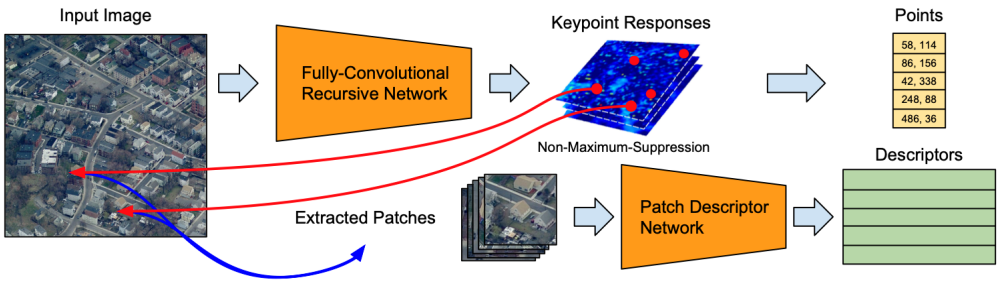


Figura 2.3: Arquitectura propuesta para detectar y describir puntos clave en múltiples escalas por Altwaijry et al. (2016)

Con un enfoque similar, Gordo et al. (2016) proponen un método para la recuperación de imágenes a nivel de instancia mediante el uso de una arquitectura profunda siamesa, entrenada con una función de costo tipo *ranking loss* y agregando múltiples descriptores regionales (ver Figura 2.4). En comparación con trabajos previos que emplean arquitecturas pre-entrenadas, aquí se entrena la red desde cero para la tarea específica de recuperación de imágenes, obteniendo características por región y determinando qué regiones deben agruparse para formar el descriptor global final. La arquitectura propuesta produce una representación de imagen global, superando significativamente los enfoques anteriores basados en la indexación de descriptores locales y verificación espacial.

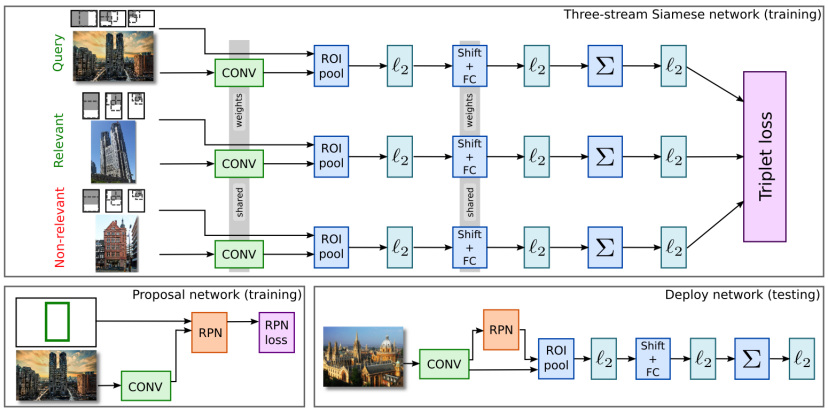


Figura 2.4: Arquitectura profunda propuesta por Gordo et al. (2016) basada en CNN adaptada para su recuperación. La red de propuesta de región (*RPN*) aprende qué regiones de imágenes deben agruparse (abajo a la izquierda).

Sin embargo, Chen et al. (2017) presentan un enfoque moderno para realizar el emparejamiento de imágenes mediante aprendizaje profundo, al aprender conjuntamente la representación de características y una métrica de similitud. La estructura del modelo consta de dos redes neuronales convolucionales (CNN, por sus siglas en inglés: *Convolutional Neural Network*), capas para la métrica de similitud y una capa *softmax*. Las dos redes CNN se entrenan simultáneamente a través del algoritmo de retropropagación (*backpropagation*). La Figura 2.5 presenta los resultados del método propuesto, mostrando que, al generar una imagen panorámica, el enfoque supera a los métodos más recientes en el emparejamiento de imágenes aéreas.

Figura 2.5: Los resultados muestra los detalles de unir dos imágenes basado en los *feature maps* obtenidos por el algoritmo *SIFT* y *RANSAC* para la eliminación de valores atípicos. Los resultados muestran que el método propuesto por Chen et al. (2017) es efectivo para hacer coincidir puntos, donde la mayoría de los puntos coincidentes son correctos. Algunos resultados típicos de unión de panoramas de imágenes de UAV muestran la efectividad del método propuesto.

Por su parte, Weerasekera et al. (2017) presentan un modelo eficaz para la reconstrucción de escenas 3D utilizando la entrada de una cámara monocular en movimiento. La técnica desarrollada aprovecha el alto nivel de aprendizaje de una CNN para proporcionar una solución simple pero eficiente que mejora la exactitud de las reconstrucciones densas, especialmente en situaciones donde hay poca evidencia fotométrica. La incorporación de orientaciones de superficie aprendidas permite obtener reconstrucciones suaves y precisas. Este trabajo representa un avance en la unificación de dos tareas principales: la reconstrucción 3D y la comprensión de la escena, contribuyendo significativamente a los robots autónomos basados únicamente en visión artificial.

Si bien el uso de *feature maps* es una herramienta fundamental para realizar el emparejamiento de imágenes, ya sea extraídos por métodos clásicos de visión por computadora o mediante técnicas de aprendizaje profundo, existen limitaciones en su aplicación. Una de estas limitaciones es la presencia de valores atípicos, que pueden presentar problemas en la creación de imágenes panorámicas. Para resolver este tipo de problemas, existen diversas propuestas, una de ellas es la presentada en Tang et al. (2018), quienes muestran un esquema para generar la correspondencia geométrica mediante puntos característicos.

El método combina una Red Neuronal Convolucional (CNN, por sus siglas en inglés: *Convolutional Neural Network*) y una Red Neuronal Recurrente (RNN, por sus siglas en inglés: *Recurrent Neural Network*) (ver Figura 2.6) para estimar la ubicación de los puntos característicos buscados y, a partir de ello, generar descriptores. El proceso consiste básicamente en alinear la salida de una CNN profunda con una red recurrente superficial para realizar tanto la extracción de características como la detección de puntos clave. Los resultados de la reconstrucción del entorno se muestran en la Figura 2.7.

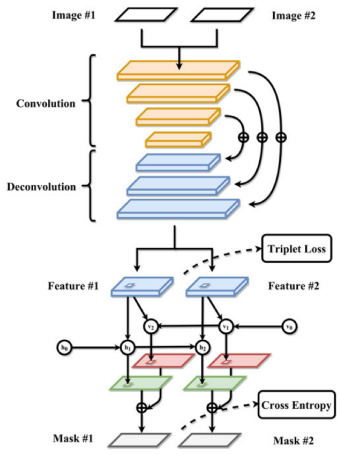


Figura 2.6: Arquitectura de la red neuronal presentada por Tang et al. (2018). Esta arquitectura está compuesta por una CNN en la parte superior que extrae características de forma densa. Consta de una parte totalmente convolucional para la representación de características a múltiples escalas y una parte de deconvolución para el refinamiento de detalles. La parte inferior es una estructura de red RNN bidireccional poco profunda para predecir la ubicación de los puntos clave en ambas imágenes.

(a) (b)

Figura 2.7: Se muestran dos ejemplos de reconstrucciones de nubes de puntos con el método presentado por Tang et al. (2018), el cual logra resultados competitivos en comparación con otros métodos. Se puede apreciar que, incluso con secuencias largas de nubes de puntos, el método mantiene la continuidad en la secuencia de reconstrucción.

De forma similar Ghamisi and Yokoya (2018) proponen *cGAN* (ver Figura 2.8), un enfoque para simular el modelo digital de superficie (DSM) a partir de una sola imagen óptica. La arquitectura utiliza una red de codificador-decodificador. La red se entrena en escenas donde tanto el DSM como los datos ópticos están disponibles para establecer una regla de traducción de imagen a DSM. La red entrenada se utiliza para simular información de elevación en escenas de destino donde no existe información de elevación correspondiente.

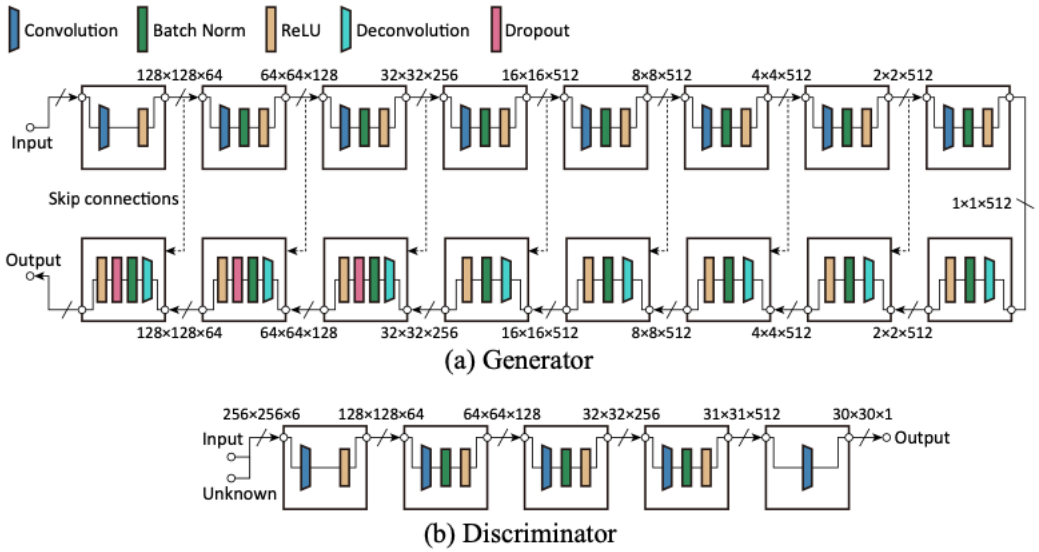
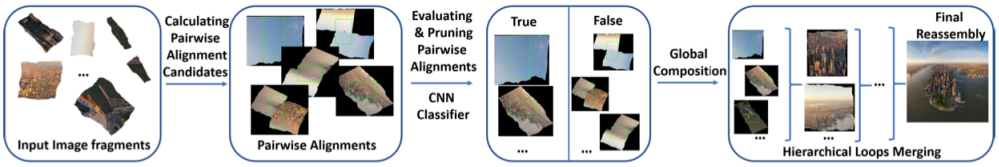


Figura 2.8: Arquitectura del generador y discriminador respectivamente del trabajo presentado por Chen et al. (2017). Para la red del generador, las entradas son imágenes en color (IRRG) con un tamaño de $256 \times 256 \times 3$ y las salidas son el DSM simulado correspondiente con un tamaño de $256 \times 256 \times 3$, donde el mismo componente DSM se concatena tres veces. Las líneas discontinuas indican los conectores de salto.

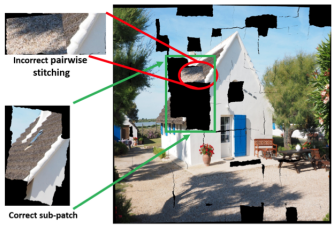
La capacidad del enfoque propuesto se evalúa tanto visualmente (en términos de interpretación fotográfica) como cuantitativamente en términos de errores de reconstrucción y exactitud de clasificación en conjuntos de datos de resolución espacial mediante el RMSE (del inglés: *Root Mean Square Error*) y el ZNCC (del inglés: *Zero-Mean Normalized Cross Correlation*). Los resultados demuestran claramente que, aunque es el primer estudio de este tipo, el enfoque propuesto puede producir información de elevación adecuada, lo que mejora significativamente la exactitud de la clasificación.

Enfoques de reconstrucción de mayor complejidad se presentan en el artículo de Le and Li (2019), donde proponen un algoritmo para reensamblar una imagen fragmentada a su estado original (ver Figura 2.9a). En contraste con los métodos existentes, que generalmente utilizan una etapa de correspondencia local y una etapa de composiciones globales, en este trabajo se construye una red neuronal convolucional profunda para detectar la compatibilidad de una unión por pares y se emplean dos algoritmos de búsqueda basados en el cierre de bucle (ver Figura 2.9b). Los experimentos muestran que el algoritmo supera significativamente los métodos existentes. Además, estas estrategias podrían extenderse potencialmente a otras tareas de reconstrucción (ver Figura 2.9c).



(a)

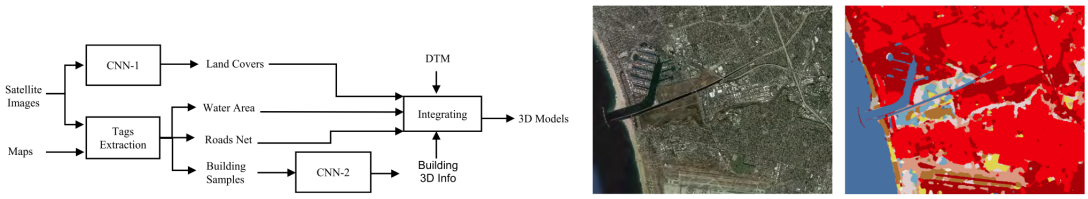
(b)



(c)

Figura 2.9: El enfoque propuesto por Le and Li (2019) contiene tres componentes: extracción de candidatos de alineación por pares, medición de compatibilidad por pares y composición global (Figura 2.9a). Este algoritmo de coincidencia produce un conjunto de alineaciones posibles, en las que existen alineaciones correctas e incorrectas, siendo mucho mayor el número de alineaciones incorrectas que el de las correctas. La arquitectura de la red neuronal convolucional propuesta en la Figura 2.9b se compone de bloques convolucionales (CB del inglés: *Convolutional Blocks*) y bloques residuales (RB del inglés: *Residual Blocks*). Los resultados en la Figura 2.9c muestran que la fusión de los fragmentos en una única imagen panorámica contiene buenas condiciones, aunque algunos fragmentos aún requieren mejoras.

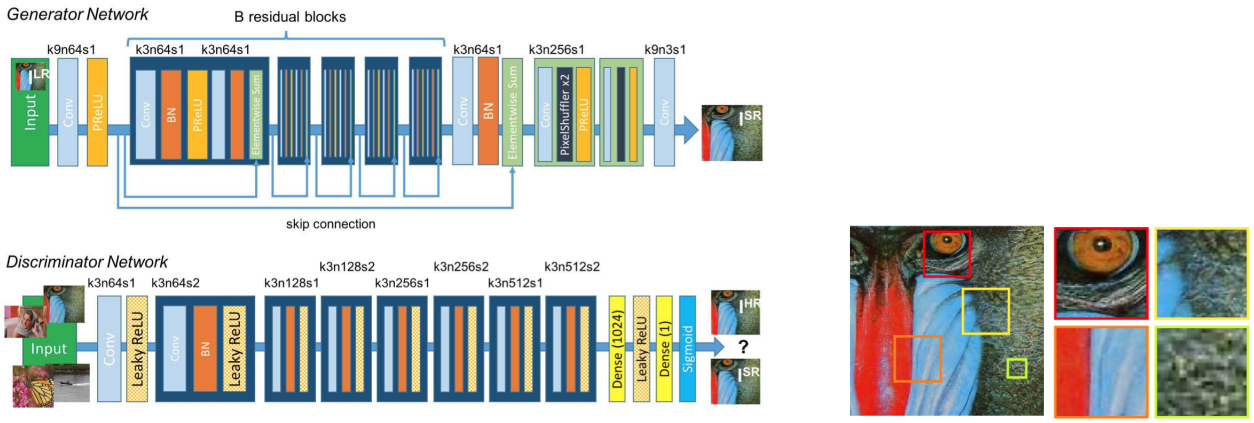
Por otra parte, Li et al. (2019) presentan un método eficiente para la reconstrucción de escenas urbanas virtuales en 3D basado en la detección remota de múltiples fuentes de big data y aprendizaje profundo. Al integrar mapas, imágenes ópticas satelitales y un modelo digital del terreno (DTM), el método propuesto logra un modelo 3D de zonas urbanas complejas. El método utiliza dos redes neuronales convolucionales independientes (CNN) para procesar la cobertura del suelo y la extracción de altura del edificio. El método propuesto se pone a prueba en una escena de 100 km^2 en San Diego, EE. UU., reconstruyendo una escena 3D virtual que incluye alrededor de $30,000$ edificios, con un tiempo aproximado de procesamiento de 10 minutos (utilizando una *GPU NVidia Titan X*). Los resultados obtenidos podrían utilizarse en la simulación de información geográfica. El módulo de clasificación de cobertura del suelo del sistema también puede procesar áreas desconocidas.



(a) (b)

Figura 2.10: El diagrama de flujo de Li et al. (2019) muestra el procedimiento de procesamiento propuesto para la reconstrucción 3D. El método utiliza mapas digitales, imágenes ópticas de satélite y un DTM como datos de entrenamiento para generar un modelo urbano 3D. Los resultados muestran una imagen satelital (Figura 2.10b izquierda) con su respectiva predicción de la cobertura del suelo (Figura 2.10b derecha).

En contraste con todas estas propuestas para la reconstrucción, el problema de recuperar los detalles de textura fina sigue siendo un campo de estudio. Por lo tanto, Ledig et al. (2017) se han centrado en minimizar el error de reconstrucción. En su artículo, presentan la SRGAN, una red de confrontación generativa (GAN del inglés: *Generative Adversarial Networks*) para la super-resolución de imágenes (*SR*). Utilizando una red discriminadora entrenada para diferenciar entre las imágenes con superresolución y las imágenes originales, la red residual profunda propuesta es capaz de recuperar texturas fotorrealistas de imágenes fuertemente muestreadas en puntos de referencia públicos. Sin embargo, aún se presentan algunos detalles desagradables. Para mejorar aún más la calidad visual, Wang et al. (2019b) presentan *ESRGAN*, un *SRGAN* mejorado mediante un Bloque Denso Residual. Con estas mejoras, el *ESRGAN* propuesto logra una calidad visual consistentemente mejor, con texturas más realistas y naturales que la *SRGAN*.



(a) (b) (c)

Figura 2.11: (Ledig et al., 2017) presentan una arquitectura de red generadora y discriminadora (Figura 2.11a) con el tamaño correspondiente para producir resultados perceptualmente más convincentes. Los mapas de características de estas capas profundas se centran exclusivamente en el contenido, dejando que la pérdida adversaria se enfoque en los detalles de la textura, que son la principal diferencia entre las imágenes súper resueltas sin pérdida adversaria y las imágenes fotorrealistas (Figura 2.11b y 2.11c). El desarrollo de funciones de pérdida de contenido que describen el contenido espacial de la imagen, pero que son más invariantes a los cambios en el espacio de píxeles, mejora aún más los resultados de imágenes fotorrealistas.

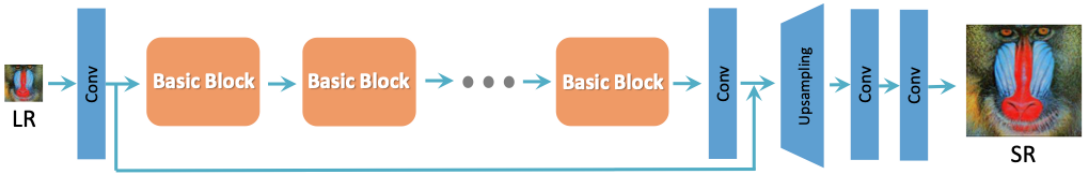


Figura 2.12: Wang et al. (2019b) emplean una arquitectura básica de *SRResNet*, donde la mayor parte del cálculo se realiza en el espacio de características. La red está diseñada con bloques residuales, bloques densos y *RRDB* para un mejor rendimiento en la obtención de imágenes (Ledig et al., 2017).

Capítulo 3

Marco Teórico

3.1. La Inteligencia Artificial

Definir de manera exacta lo que es la Inteligencia Artificial (IA) es una tarea compleja, especialmente porque depende de cómo definimos la inteligencia, un concepto que, hasta la fecha, tiene múltiples interpretaciones. Muchos autores han propuesto sus propias definiciones (AWS, 2020; Goodfellow et al., 2016). Sin embargo, si extraemos una idea común de todas ellas, podemos decir que ***la inteligencia artificial es la disciplina dentro del campo de la informática que busca crear máquinas capaces de imitar comportamientos inteligentes***.

Estos comportamientos pueden ser muy diversos, como el análisis de patrones, el reconocimiento de voces o la capacidad de ganar juegos. Son muchas las formas en las que una máquina puede simular un comportamiento inteligente, logrando en ciertas áreas alcanzar e incluso superar el rendimiento humano. La IA surge como una herramienta para resolver problemas que son intelectualmente desafiantes para los seres humanos, pero relativamente sencillos de implementar en una computadora, ya que se pueden describir mediante un conjunto de reglas formales y matemáticas. No obstante, el verdadero desafío radica en abordar tareas cognitivas complejas (Goodfellow et al., 2016). Esto nos permite clasificar las inteligencias en dos tipos: débiles y fuertes. La inteligencia artificial débil hace referencia a sistemas que solo pueden realizar un conjunto limitado de tareas, mientras que la inteligencia artificial fuerte se refiere a sistemas capaces de aplicarse a una amplia variedad de problemas y en diferentes dominios.

3.2. De la IA al Deep Learning

Para resolver tareas cognitivas, que nosotros solemos llevar a cabo de manera intuitiva y que incluso pueden parecer automáticas, como caminar y pensar al mismo tiempo, o ver y hablar con un amigo mientras paseamos por la calle, es necesario dotar a las computadoras de la capacidad de aprender de la experiencia. Además, deben poder comprender el mundo en términos de una jerarquía de conceptos, es decir, reunir conocimientos basados en la experiencia.

En el campo de la inteligencia artificial, existen diversas subcategorías que abordan diferentes comportamientos inteligentes. Por ejemplo, se busca dotar a un robot de la capacidad para moverse y adaptarse a su entorno o para desarrollar la habilidad de entender el lenguaje natural. Cada una de estas competencias conforma un área de estudio especializada dentro de la inteligencia artificial. Sin embargo, si hay una cualidad que verdaderamente nos define como agentes inteligentes, es la capacidad de aprender, que en el contexto de la robótica inteligente se conoce como *machine learning* o *aprendizaje automático*.

El aprendizaje automático es la rama de la inteligencia artificial que estudia cómo dotar a las máquinas de la capacidad de aprender. Este aprendizaje se entiende como la capacidad de generalizar el conocimiento a partir de un conjunto de experiencias. El aprendizaje automático se puede dividir en tres grupos: ***supervisado, no supervisado y reforzado o por refuerzo***. Dentro del aprendizaje automático, existen diversas técnicas que se enfocan en aplicaciones específicas, como los árboles de decisión, modelos de regresión, modelos de clasificación, técnicas de *clusterización*, entre muchas otras.

Sin embargo, en la última década, una de estas técnicas ha ganado gran popularidad: las redes neuronales. Lo interesante de las redes neuronales es que son capaces de aprender de manera jerárquica, es decir, la información se aprende en niveles. Las primeras capas aprenden conceptos concretos, como qué es un tornillo, un espejo o una rueda. En capas posteriores, esta información aprendida previamente se usa para aprender conceptos más abstractos, como qué es un coche, un camión o una moto. A medida que añadimos más capas, la información aprendida se vuelve más abstracta e interesante. Esta estructura jerárquica permite que la computadora aprenda conceptos complejos a partir de conceptos simples que se construyen uno encima del otro, capa por capa. La tendencia actual es añadir cada vez más capas, haciendo que las redes sean más complejas. Este incremento en el número de capas y en la complejidad es lo que da lugar a los algoritmos de ***Deep Learning*** (aprendizaje profundo) (Goodfellow et al., 2016; Murphy, 2012).

El *Deep Learning* es un conjunto de técnicas que ha cobrado gran popularidad en los últimos años. Estas técnicas se entrenan y aprenden a partir de datos, y hoy en día estamos inmersos en la era de la información. Con la llegada de la digitalización, hemos entrado en una tendencia de acumulación de datos, lo que se denomina *big data*.

En resumen, el *big data* hace referencia a la acumulación de grandes cantidades de datos y también se suele utilizar para referirse al proceso de análisis de todos estos datos para transformarlos en conocimiento, para esto se requieren de complejas técnicas como lo son las técnicas de *Deep Learning*, versión potenciadas de redes neuronales, una familia de algoritmos del aprendizaje automático o *machine learning* y por tanto al campo de la inteligencia artificial. Esta distribución de las diferentes áreas de las que se compone la IA se esquematiza en la Figura 3.1. Donde es posible identificar de forma general, simple y resumida las áreas aquí anteriormente descritas.

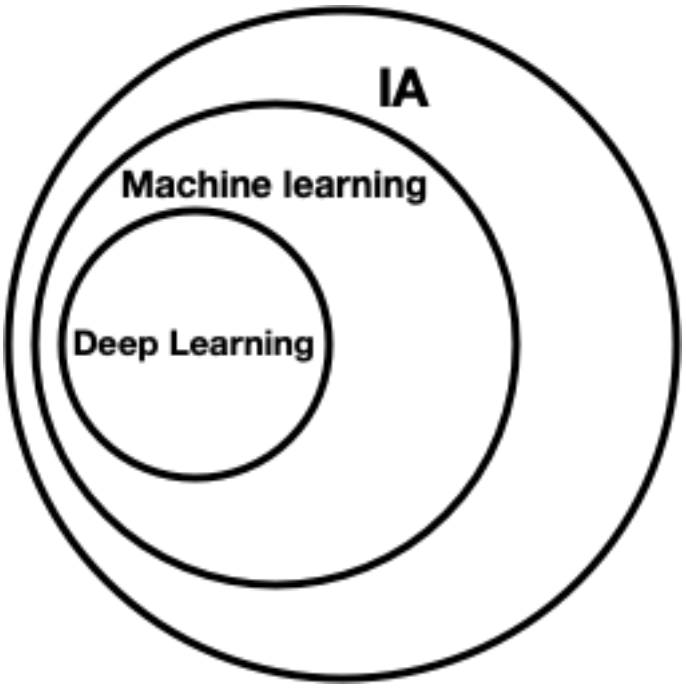


Figura 3.1: De la IA al Deep Learning. Este esquema basado en el diagrama de Venn de Goodfellow et al. (2016), donde se puede observar cómo el Deep Learning es un tipo de aprendizaje utilizado en el desarrollo de una IA.

3.3. Redes Neuronales Profundas

Los métodos de aprendizaje profundo son métodos de representación con múltiples niveles (capas), obtenidos mediante la composición de módulos simples, con las suficientes transformaciones de este tipo, se pueden aprender funciones muy complejas. Por ejemplo, para clasificación, las capas superiores de representación, amplifican aspectos de la entrada que son importantes para la discriminación y eliminan los irrelevantes. En una imagen, por ejemplo, las características aprendidas en la primera capa de representación generalmente representan la presencia o ausencia de bordes en orientaciones y ubicaciones particulares en la imagen. La segunda capa normalmente detecta disposiciones particulares de los bordes, independientemente de las pequeñas variaciones en las posiciones de los bordes. La tercera capa puede detectar partes de objetos, y las capas posteriores detectarían objetos como combinaciones de estas partes. El aspecto clave del aprendizaje profundo es que se aprenden de los datos mediante un procedimiento de aprendizaje.

Al agregar más capas y unidades dentro de cada una de éstas, una red profunda puede representar funciones de complejidad creciente. La mayoría de las tareas que consisten en mapear un vector de entrada a un vector de salida, y que son fáciles de realizar para una persona rápidamente, pueden llevarse a cabo a través del aprendizaje profundo, dados modelos suficientemente grandes y conjuntos de datos igualmente amplios de ejemplos de capacitación etiquetados. Para ello introducimos la red convolucional y la red neuronal recurrente.

3.3.1. Redes Neuronales Convolucionales

Las redes convolucionales (LeCun, 1989), también conocidas como redes neuronales convolucionales o CNN (del inglés: *Convolutional Neural Networks*), son un tipo especializado de red neuronal para procesar datos que tiene una topología similar a una cuadrícula. El nombre "Red Neuronal Convolucional" indica que la red emplea una operación matemática llamada cunvolución en lugar de la multiplicación matricial general en al menos una de sus capas. La convolución es un tipo especializado de operación lineal.

Una arquitectura CNN tradicional generalmente se compone de una capa de entrada, las capas de convolución, en la que se realizan varias convoluciones en paralelo para producir un conjunto de activaciones lineales mediante funciones de activación o de detección, cada activación lineal se aplica a través de funciones de activación por ejemplo ReLU. Después, se continua con una capa de *pooling*, donde se usa una función de agrupación para modificar la salida de la capa y poderla conectar con una capa totalmente conectada FC (del inglés: *Fully Connected*) o la salida de la red (ver figura 3.2).

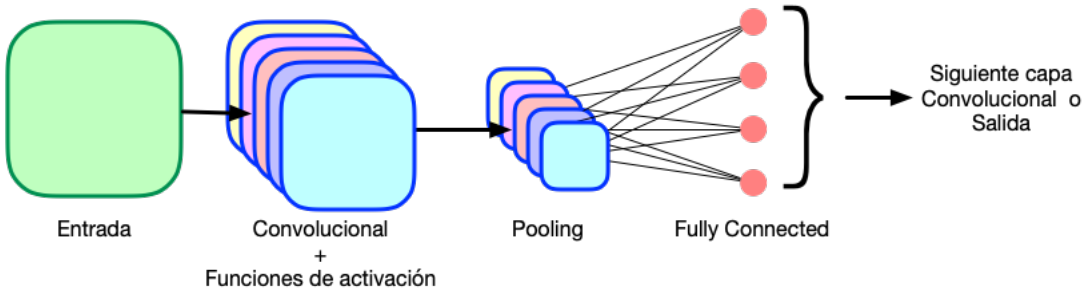


Figura 3.2: Arquitectura típica de una red CNN. Basado en el esquema que propone Amidi (2019) la arquitectura se compone de una capa de entrada, una capa de convolución, seguida de una capa de *pooling* y finalmente una capa totalmente concectada o la salida de la red.

Las redes convolucionales constituyen un ejemplo notable de cómo los principios neurocientíficos influyen en el aprendizaje profundo. La investigación en arquitecturas de redes convolucionales avanza a un ritmo tan acelerado que se presenta una nueva cada pocas semanas o meses, lo que dificulta la descripción de la mejor arquitectura. No obstante, las arquitecturas más eficaces presentadas hasta la fecha se han basado consistentemente en los bloques mostrados en la figura anterior. Tanto la capa de convolución como la de agrupación pueden ajustarse en función de los hiperparámetros que se describen en las secciones siguientes.

Capa de Convolución

La convolución es el componente esencial de las CNNs y consiste en la aplicación de filtros (o kernels) que se desplazan sobre la imagen de entrada para detectar patrones específicos, como bordes, texturas y formas. Cada filtro genera un mapa de características que resalta la presencia de un patrón particular en la imagen.

Para mejorar la capacidad de aprendizaje de la red, se aplican funciones de activación como ReLU, las cuales introducen no linealidad y permiten modelar relaciones complejas en los datos. En una CNN, se emplean múltiples capas convolucionales, cada una realizando operaciones de convolución sobre la imagen que entra a cada capa. Este proceso permite extraer progresivamente características de alto nivel, facilitando la identificación de patrones relevantes en la imagen. Una vez completada la etapa de convolución, las características detectadas se utilizan para la posterior clasificación o reconocimiento de objetos.

- ✓ Operación de Convolución La operación de convolución en dos dimensiones aplicada a imágenes puede expresarse formalmente mediante la siguiente ecuación. En ella, se calcula cada valor del mapa de características resultante $S(i, j)$ como una suma ponderada de una región local de la imagen de entrada I mediante un filtro (o *kernel*) K:

$$S(i,j) = (I * K)(i,j) = \sum_m \sum_n I(m,n)\, K(i-m, j-n) \tag{3.1}$$

donde m y n son los índices del núcleo (filtro), recorriendo sus filas y columnas, respectivamente. I representa la imagen de entrada, y K el núcleo aplicado. Las coordenadas (i, j) corresponden a la posición en la salida S, es decir, el mapa de características resultante de aplicar la convolución. Este proceso se ilustra en la Figura 3.3 (Goodfellow et al., 2016).

Funciones de activación ReLU

La función de activación ReLU (Rectified Linear Unit) es una operación no lineal que se aplica de forma elemento a elemento en las salidas de una capa. Su principal objetivo es introducir no linealidades en la red neuronal, permitiendo que el modelo aprenda funciones complejas más allá de combinaciones lineales. La función ReLU, denotada comúnmente como $g(x) = max(0, x)$, anula los valores negativos y deja pasar los positivos sin modificar.

Existen diversas variantes de ReLU que abordan algunas de sus limitaciones, como la "neurona muerta.[en] el caso de gradientes cero. Las versiones más comunes se resumen en la Tabla 3.1.

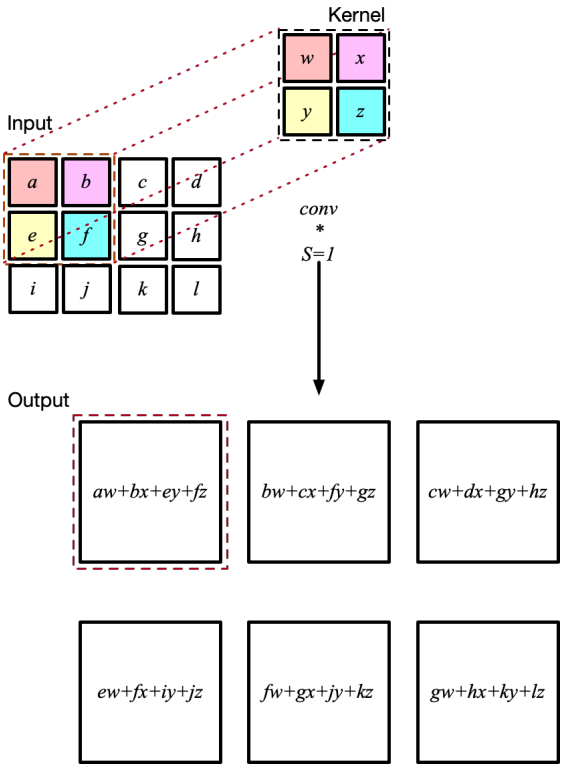
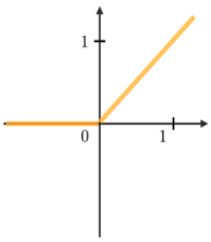
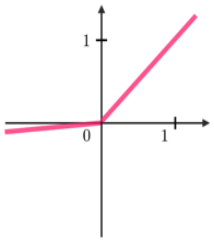
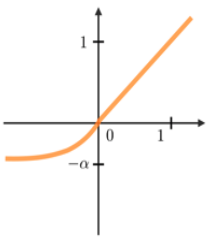


Figura 3.3: Representación de una operación de convolución bidimensional aplicada sobre una matriz de entrada. En este caso, un núcleo o *kernel* de dimensiones 22 se desplaza de forma sistemática sobre la matriz original (*Input*), generando una nueva matriz de salida (*Output*). La operación se restringe a las posiciones en las que el núcleo se encuentra completamente contenido dentro de los límites de la matriz de entrada. Cada elemento de la matriz de salida se obtiene mediante la suma ponderada de los valores de una subregión de la entrada, de acuerdo con los coeficientes del núcleo aplicado.

Tabla 3.1: La unidad lineal rectificada ReLu, se ha vuelto muy popular en los últimos años. En esta tabla se presentan algunas de sus variantes más populares empleadas para el entrenamiento de una CNN.

ReLU	**Leaky ReLU**	**ExponentialReLU (ELU)**
$g(z) = max(0, z)$	$g(z) = max(\epsilon z, z)$ con $\epsilon \ll 1$	$g(z) = max(\alpha(e^{\epsilon} - 1), z)$ con $\alpha \ll 1$
		

Tomada de (Amidi, 2019).

Pooling

La capa de *pooling* generalmente se aplica después de una capa de convolución, que produce cierta invariancia espacial es decir, si traducimos la entrada en una pequeña cantidad, los valores de la mayoría de las salidas agrupadas no cambian por ejemplo, al determinar si una imagen contiene una cara, no necesitamos saber la ubicación de los ojos con exactitud perfecta, con saber que hay un ojo puesto en el lado izquierdo y uno en el lado derecho es suficiente.

En otros contextos, es más importante preservar la ubicación de una característica por ejemplo, sí queremos encontrar una esquina definida por dos bordes que se encuentran en una orientación específica, debemos preservar la ubicación de los bordes lo suficiente para probar que se ha encontrado. Para muchas tareas, el *pooling* es esencial para manejar entradas de diferentes tamaños. Una función de *pooling* reemplaza la salida en alguna capa con una estadística por ejemplo, la operación de *max pooling* (Zhou and Chellappa, 1988) y *average pooling* presentan el valor máximo y promedio de un conjunto de vecinos más cercanos en un area respectivamente (ver Figura 3.4).

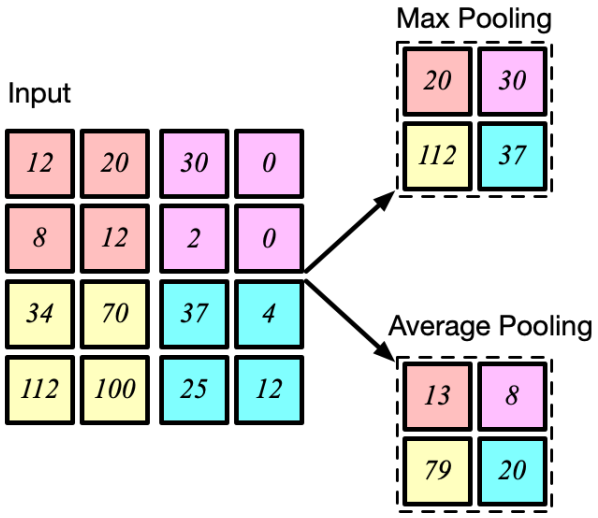


Figura 3.4: Representación de las operaciones de Max Pooling y Average Pooling. La figura muestra una representación visual del funcionamiento de las operaciones de pooling, que comúnmente siguen a las capas convolucionales en una red neuronal profunda. En la parte izquierda se presenta una matriz de activaciones dividida en regiones de $2x2$, a partir de las cuales se aplican dos funciones de pooling. La parte superior derecha muestra el resultado de aplicar max pooling, donde se conserva el valor máximo de cada subregión (por ejemplo, el valor 112 en la segunda subregión). En contraste, la parte inferior derecha ilustra el resultado de average pooling, en el que se calcula el promedio de cada subregión (por ejemplo, 79 en la segunda subregión). Estas operaciones permiten reducir la dimensionalidad espacial de las representaciones, introducir invariancia espacial local y mantener las características más relevantes, facilitando el aprendizaje eficiente en tareas de clasificación, detección o segmentación.

Fully Connected

La capa completamente conectada (FC, por sus siglas en inglés: *Fully Connected*) opera sobre una representación de entrada previamente aplanada (*flattened*), en la que cada unidad de entrada está conectada a todas las neuronas de la capa siguiente. Este tipo de capa introduce una transformación lineal global seguida, típicamente, de una función de activación no lineal. Las capas FC suelen ubicarse en las etapas finales de las arquitecturas convolucionales (CNN) y tienen como propósito integrar las características extraídas en etapas anteriores para producir una salida compatible con la tarea de clasificación, regresión o detección. En contextos de clasificación, estas capas son responsables de optimizar los puntajes asociados a cada clase objetivo.

Backpropagation

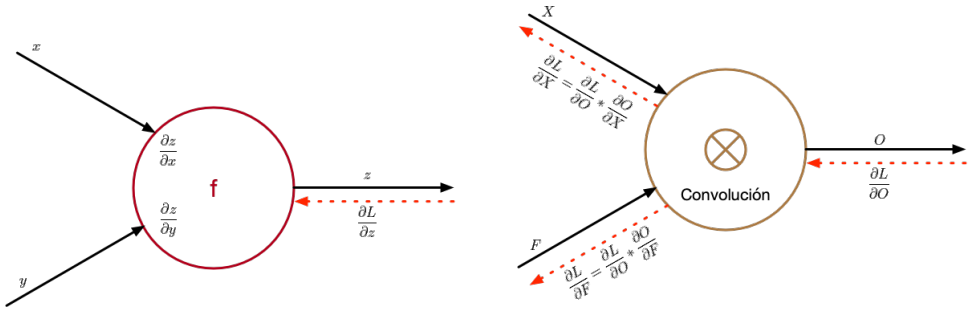
Las arquitecturas multicapa se pueden entrenar mediante un simple descenso de gradiente estocástico. Mientras que los módulos sean funciones relativamente suaves de sus entradas y de sus pesos internos, uno puede calcular el gradientes usando el procedimiento de *backpropagation*. La idea clave es que la derivada (o gradiente) del objetivo con respecto a la entrada de un módulo se puede calcular trabajando hacia atrás desde el gradiente con respecto a la salida de ese módulo.

La ecuación de *backpropagation* se puede aplicar repetidamente para propagar el gradiente a través de todos los módulos, comenzando desde la salida en la parte superior (donde la red produce su predicción) hasta la parte inferior (donde se alimenta la entrada externa). Una vez que se han calculado estos gradientes, es sencillo calcular los gradientes con respecto a los pesos de cada módulo (Amelie et al., 2020; Baydin et al., 2017; Goodfellow et al., 2016; Zhang, 2016). Podemos imaginar una CNN como un gráfico computacional masivo. Digamos que tenemos una neurona f en ese gráfico computacional con entradas x y y que produce una salida z. Donde $\partial L/\partial z$ es la pérdida de la capa de función anterior que tiene que volver a propagarse a otras capas con gradientes locales $\partial z/\partial x$ y $\partial z/\partial y$ obtenidos con la regla de la cadena (Solai, 2018).

Supongamos ahora que la función f es una convolución de la entrada X y un filtro F de la cual obtenemos la salida O (ver figura 3.5b). Como se mencionó anteriormente, obtenemos el gradiente de pérdida con respecto a la salida O de la siguiente capa como $\partial L/\partial O$, durante la propagación hacia atrás y combinando con el conocimiento previo para obtener todos los pesos de las conexiones de las neuronas. Los gradientes locales $\partial O/\partial X$ y $\partial O/\partial F$ con respecto a la salida O. Con el gradiente de pérdida de las capas anteriores $\partial L/\partial O$ y la regla de la cadena se calcula $\partial L/\partial X$ y $\partial L/\partial F$. Esta es la forma en que se realiza la propagación hacia atrás o *Backpropagation* dentro de la CNN a través de sus capas y con ello los pesos de cada capa usando la una funciónde costo.

3.3.2. Redes Neuronales Recurrentes

Las redes neuronales recurrentes o también conocidas como RNNs (del inglés: *Red Neuronal Recurrente*) son una familia de redes neuronales para procesar datos secuenciales, una red neuronal recurrente es una red especializada para procesar una secuencia de valores $x^{(1)}...x^{(\tau)}$.



(a) (b)

Figura 3.5: (3.5a) Representación de un modelo de una CNN mediante un gráfico computacional. (3.5b) Representación de la la propagación hacia atrás dentro de una CNN a través de sus capas medainte la obtenecíon de los gradiente de pérdida con respecto a la salida O de la capa (Solai, 2018).

Así como las redes convolucionales pueden escalar a imágenes de alta resolución o manejar imágenes de tamaños variables, las RNNs pueden escalar a secuencias largas y también procesar secuencias de longitud variable que para una red sin especialización no le sería práctico (Goodfellow et al., 2016).

Para pasar de redes convolucionales a redes recurrentes, debemos aprovechar una de las primeras ideas encontradas en el aprendizaje automático "compartir parámetros en diferentes partes de un modelo".

El uso compartido de parámetros permite extender y aplicar el modelo a ejemplos de diferentes formas o longitudes, en este caso. Este intercambio es particularmente importante cuando una información específica puede ocurrir en múltiples posiciones dentro de la secuencia. Supongamos que entrenamos una red *feedforward* que procesa oraciones de longitud fija. Esta red tradicional completamente conectada tendría parámetros separados para cada característica de entrada. En comparación, una red neuronal recurrente comparte los mismos pesos en varios pasos de tiempo. Estas redes permiten utilizar salidas anteriores como entradas mientras tiene estados ocultos.

Típicamente la arquitectura de una red RNN es como se muestra en la figura 3.6. Donde se puede apreciar como la información transcurre en pasos de tiempo t, con funciones de activación $a^{<t>}$ y salida $y^{<t>}$. Mismas que se expresan como 3.2 y 3.3. Donde $W_{ax}, W_{aa}, W_{ya}, b_a, b_y$ son coeficientes temporales y g_1, g_2 son funciones de activación (Amidi, 2019).

$$a^{<t>} = g_1(W_{aa}a^{<t-1>} + W_{ax}x^{<t>} + b_a) \tag{3.2}$$

$$y^{<t>} = g_2(W_{ya}a^{<t>} + b_y) \tag{3.3}$$

Las redes neuronales recurrentes (RNN, por sus siglas en inglés: *Recurrent Neural Networks*) presentan una arquitectura que, si bien puede implicar un procesamiento secuencial relativamente lento y una complejidad elevada en el acceso a la información a largo plazo, permite modelar secuencias de longitud variable sin que el número de parámetros crezca proporcionalmente con el tamaño de la entrada.

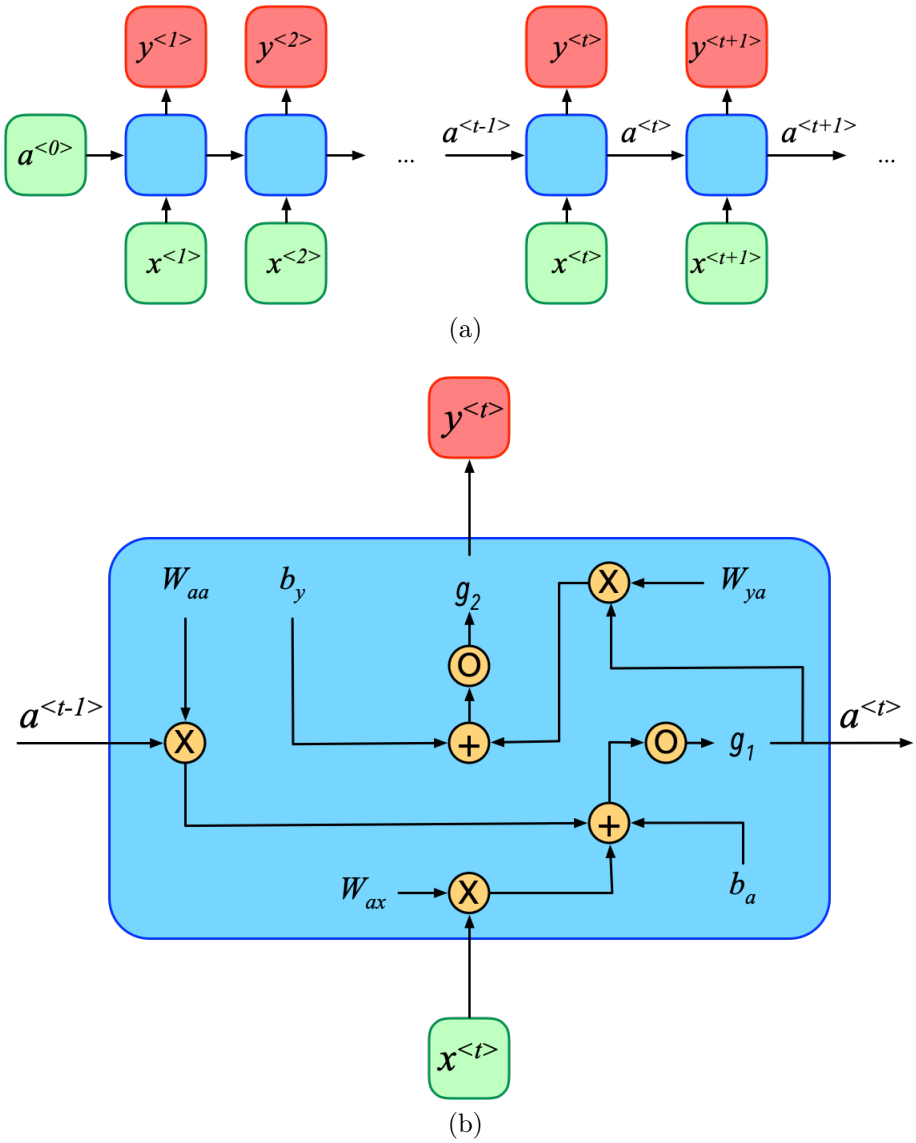


Figura 3.6: (3.6a)Modelo general de una arquitectura de una RNN. (3.6b) Modelo interno de una red RNN, se muestran las diferentes operaciones y procesos que se realizan internamente para el compartimiento de la información y poder trabajar con información secuencial. Por su estructura interna tiene un gran potencial principalmente con el procesamiento de lenguaje natural (Tomada de (Amidi, 2019)).

Esto es posible debido a que los parámetros de la red se comparten en cada paso temporal, y los cálculos computacionales se realizan de manera recurrente utilizando el historial de activaciones anteriores. Estas propiedades hacen que las RNN sean especialmente adecuadas para tareas secuenciales, como el procesamiento del lenguaje natural (PLN) y el reconocimiento automático del habla, donde la dependencia temporal entre elementos es fundamental para la comprensión del contexto.

Función de costo

Las redes neuronales recurrentes (RNN) se entrenan mediante la minimización de una función de costo que mide la discrepancia entre las salidas esperadas y las generadas por el modelo en cada paso de tiempo. Para una secuencia de longitud T, con entradas $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(T)}$ y salidas esperadas $\mathbf{y}^{(1)}, \ldots, \mathbf{y}^{(T)}$, la función de costo total se define como la suma de las pérdidas a través del tiempo:

$$\mathcal{L}(\theta) = \sum_{t=1}^{T} \mathcal{L}^{(t)} = -\sum_{t=1}^{T} \sum_{k=1}^{K} y_k^{(t)} \log \hat{y}_k^{(t)} \tag{3.4}$$

donde:

- ✓ T es la longitud de la secuencia.
- ✓ K es el número de clases posibles para cada paso de tiempo.
- ✓ $y_k^{(t)}$ es la variable indicadora de clase verdadera en el tiempo t.
- ✓ $\hat{y}_k^{(t)}$ es la probabilidad predicha por el modelo para la clase k en el tiempo t, obtenida mediante softmax:

$$\hat{y}_k^{(t)} = \frac{\exp(z_k^{(t)})}{\sum_{j=1}^{K} \exp(z_j^{(t)})}$$

- ✓ $z_k^{(t)}$ es el logit (salida antes de softmax) en el tiempo t para la clase k.
- ✓ θ representa los parámetros del modelo compartidos a través del tiempo.

Esta función de costo acumula la pérdida en todos los pasos de la secuencia, permitiendo el ajuste de los parámetros mediante algoritmos como **backpropagation through time (BPTT)**. El objetivo es minimizar $\mathcal{L}(\theta)$ para que las salidas predichas se alineen con las secuencias objetivo.

Funciones de activación para redes RNN

Las funciones de activación más comunes utilizadas en RNN son:

- ✓ Sigmoide: $g(z) = \frac{1}{1-e^{-z}}$
- ✓ Tanh: $g(z) = \frac{e^{z}-e^{-z}}{e^{z}+e^{-z}}$
- ✓ ReLU: $g(z) = max(0, z)$

3.3.3. Redes Residuales

En general, una red CNN típica contiene varias capas convolucionales. Estas capas aplican la operación de convolución entre un filtro y una imagen para generar mapas de características necesarios para realizar algún procesamiento posterior. Evidencia reciente revela que la profundidad de la red es de crucial importancia al trabajar con desafiantes conjuntos de datos lo que ha generado modelos "muy profundos"(Girshick, 2015; Karen, 2014; Szegedy et al., 2015).

Sin embargo, cuando las redes más profundas pueden comenzar a converger, se expone un problema de degradación: con el aumento de la profundidad de la red, la exactitud se satura (lo que podría no ser sorprendente) y luego se degrada rápidamente. Inesperadamente, tal degradación no es causada por el sobreajuste, y agregar más capas a un modelo adecuadamente profundo conduce a un mayor error de entrenamiento (He and Sun, 2015; Srivastava et al., 2015).

Para resolver el problema de degradación se presenta un nuevo tipo de arquitectura denominada Red Residual o ResNet (del inglés: *Residual Network*) (ver Figura 3.7), que fue propuesto en 2015 por investigadores de Microsoft Research. En esta red se usa una técnica llamada *skip connections* o conexiones de acceso directo, que permite saltar u omitir el entrenamiento de algunas capas y se conecta directamente a la salida, las conexiones de acceso directo simplemente realizan un mapeo de identidad y sus salidas se agregan a las salidas de las capas apiladas (Girshick et al., 2014; He et al., 2015b; Russakovsky et al., 2015).

Las conexiones *skip connections* no agregan parámetros adicionales ni complejidad computacional. Toda la red aún puede ser entrenada de extremo a extremo con backpropagation e implementarce en algun *framework* de IA como por ejemplor Tensorflow o Caffe, entre otros más. La red residual, contrario a una CNN típica, aproxima directamente una función subyacente $H(x)$, el aprendizaje residual se convierte entonces en un ajuste de mapeo residual $F(x)$, donde:

$$F(x) := H(x) - x \tag{3.5}$$

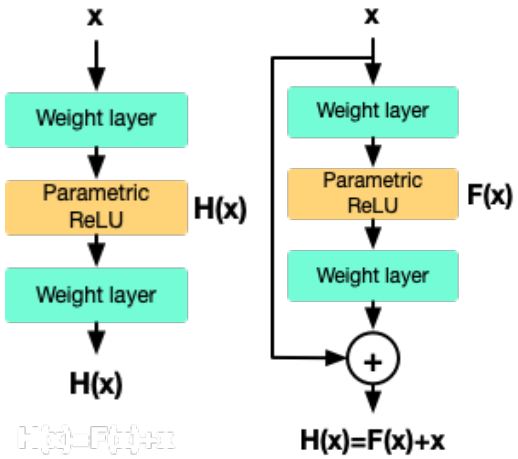
$$F(x) := H(x) - W_s x \tag{3.6}$$

La salida $F(x) + x$ de un bloque de aprendizaje residual se aproxima a la salida de una CNN típica $H(x)$. Sin embargo, es más fácil ajustar un mapeo residual $F(x)$ que el mapeo original $H(x)$, especialmente cuando $H(x)$ es un mapeo de identidad o casi identidad (He et al., 2016; Wu et al., 2018). Los parámetros de la red de aprendizaje residual se aprenden mediante una función *Parametric ReLU*, que genera una incrustación para toda la imagen de entrada.

La red residual esta construida con base en una red plana, insertando conexiones de acceso directo (Figura 3.7b) que convierten la red en su versión residual. Los atajos de identidad (ecuación (3.5)) se pueden usar directamente cuando la entrada y la salida tienen las mismas dimensiones. Cuando las dimensiones aumentan, se consideran dos opciones: (A) El acceso directo todavía realiza el mapeo de identidad, con entradas cero adicionales rellenadas para aumentar las dimensiones. En esta opción no se introduce ningún parámetro extra; (B) El atajo de proyección en la ecuación (3.6), donde W_s es la proyección lineal por las conexiones

de acceso directo para que coincida con las dimensiones mediante el uso de la coincidencia de las dimensiones. Para ambas opciones, cuando los atajos atraviesan mapas de características de dos tamaños, se realizan con un *stride* de 2.

Las redes residuales extremadamente profundas son fáciles de optimizar, contrario a las redes "planas", que simplemente apilan capa tras capa, con lo que exhiben un mayor error de entrenamiento cuando aumenta la profundidad al apilar capas consecutivamente; las redes residuales profundas, pueden disfrutar fácilmente de ganancias de exactitud a partir de una profundidad mucho mayor, produciendo resultados sustancialmente mejores que las redes anteriores (Long et al., 2015; Ren et al., 2015).



(a) (b)

Figura 3.7: Diagrama de bloques de dos modelos diferentes de CNN. En un modelo CNN típico, el bloque de aprendizaje combina unidades básicas en cascada (ver Figura 3.7a). Por el contrario, una red residual (ver Figura 3.7b), tiene una vía de acceso directo que conecta directamente la entrada y la salida en un bloque de construcción (He et al., 2016).

3.3.4. Redes Generativas Adversarias

Las Redes Generativas Adversarias o GANs por sus siglas en inglés: *Generative Adversarial Networks*. Son un tipo red neuronal convolucional profundo, una técnica emergente para el aprendizaje semi-supervisado y no supervisado. Lo logran mediante el modelado implícito de distribuciones de datos de alta dimensión. Propuesto por Goodfellow et al. (2014), brindan una manera de aprender representaciones profundas sin datos de entrenamiento extensivamente anotados. El uso de redes neuronales profundas como modelos generativos de datos complejos ha logrado grandes avances en los últimos años. Este éxito se ha logrado gracias a una sorprendente diversidad de arquitecturas de modelos, incluidos las Convolutional GANs, Adversarial Autoencoders, Fully Connected GANs, entre muchas otras más (Creswell et al., 2018; Goodfellow et al., 2020; Metz et al., 2016), sin embargo, siguen una estructura base como la mostrada en la figura 3.8.

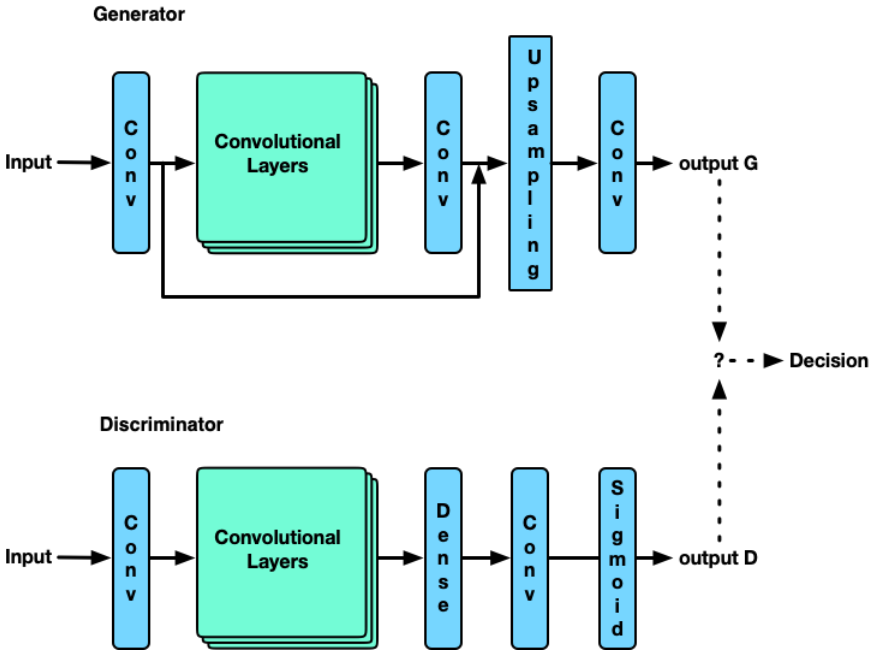


Figura 3.8: Arquitectura de la red GAN. De forma independiente se muestra la red Generadora (Generator) y la Discriminadora (Discriminator) y las capas que las componen (Goodfellow et al., 2014).

Lo que hace la arquitectura GAN, es plantear no solo un modelo sino dos inteligencias artificiales y estas se van a encargar cada una de aprender una cosa diferente y competir entre ellas, de ahí el nombre de red generativas adversarias. Una de estas redes, la red Generativa o *Generator*, es encargarda de aprender a generar datos completamente nuevos, a partir de los datos que tengamos de entrada.

Por otro lado, tenemos otra red neuronal, la Discriminadora o *Discriminator*, que se encarga de observar los datos generados por la red Generadora y aprender a discernir si esos datos creados por la red son reales o 'falsos'. De esta manera, tenemos una red que genera datos nuevos o 'falsos' y otra red neuronal que se va encargar de aprender a detectar si estos datos son 'falsos' o no.

Durante su entrenamiento, la red Discriminadora mejora progresivamente su capacidad para distinguir entre datos reales y generados. Como resultado, la red Generadora también se perfecciona, logrando producir datos cada vez más realistas. Este proceso puede entenderse como un juego de optimización en el que ambas redes se entrenan simultáneamente, impulsándose mutuamente a mejorar. Al final del entrenamiento, la red Generadora alcanza una fidelidad muy cercana a la de los datos originales. Un aspecto particularmente interesante es que la red Generadora nunca tiene acceso directo a los datos reales; su único punto de referencia proviene del aprendizaje adquirido por la red Discriminadora.

Mediante la derivación de señales de retropropagación a través de un proceso competitivo entre dos redes, las GANs son capaces de aprender representaciones complejas y detalladas. Estas representaciones pueden aplicarse en diversas áreas, como la síntesis y generación de imágenes realistas, la edición semántica de imágenes, la transferencia de estilos, la superresolución de imágenes y la mejora en tareas de clasificación, demostrando su versatilidad y potencial en el campo del aprendizaje profundo.

3.4. Reconstrucción de superficies y generación de ortoimágenes

Como es sabido, la fotogrametría, se basa en el procesamiento de imágenes, siendo sus principales productos: Modelos Digitales de Terrenos DTMs (del inglés: *Digital Terrain Models*), Modelos Digitales de Superficies DSMs (del inglés: *Digital Surface Models*), ortoimágenes, reconstrucción y clasificación 2D y 3D de objetos para aplicaciones cartográficas y visualización (mapas, vistas 3D, animación y simulación).

La fotogrametría es una técnica que permite obtener las propiedades geométricas de un objeto o una superficie a partir de múltiples imágenes con información redundante. Para lograr una reconstrucción fiel del objeto, este debe aparecer en un número suficiente de imágenes, lo que garantiza la extracción precisa de su estructura. Este proceso se basa en la superposición o solapamiento (*overlap*) entre imágenes consecutivas, permitiendo así la obtención de una representación tridimensional precisa y detallada.

El porcentaje de *overlapping* entre imágenes suele variar entre el 60 % y el 90 % y, generalmente, es determinado por un software de planificación de vuelos. Estos programas calculan la secuencia óptima de captura de imágenes en función de la posición esperada del dron, su altura y el nivel de solapamiento deseado.

El software identifica correspondencias entre imágenes y determina las posibles posiciones de un mismo elemento desde diferentes puntos de vista, lo que permite fusionar imágenes con una proporción significativa de información en común a través de un proceso conocido como *stitching* o alineación de imágenes (Brown and Lowe, 2007; Sazonov et al., 2006; Szeliski et al., 2007). Este método, también denominado mosaico de imágenes, combina múltiples capturas con áreas superpuestas para generar vistas panorámicas de alta resolución, facilitando la reconstrucción precisa del entorno tridimensional.

En los últimos años, esta técnica ha experimentado un desarrollo significativo, consolidándose como una rama esencial del procesamiento de imágenes digitales con diversas aplicaciones. Se han propuesto numerosos métodos para la implementación del *stitching*, los cuales incluyen técnicas como el registro de imágenes y la eliminación de uniones, entre otras estrategias avanzadas que optimizan la calidad y exactitud del resultado final (Fu et al., 2023; Lin et al., 2015; Sun et al., 2021; Wang and Yang, 2020). Los algoritmos de unión de imágenes crean mosaicos fotográficos de alta resolución que se utilizan para producir los mapas digitales y las fotografías satelitales actuales.

En este mismo contexto, un mosaico de imágenes aéreas se construye a partir de la unión de imágenes, de tal manera que sean superpuestas una con otra y con ello formar una sola imagen de mayor tamaño de manera que sea posible amplíar el campo de visión que proporcionan las cámaras que se encuentran a bordo de un avión o vehículo aéreo no tripulado UAV (del inglés: *Unmanned Aerial Vehicle*). El aumento de este campo de visión permite una vista de las características de interés durante más tiempo sin aumentar la complejidad del sistema, lo que representa una ventaja para el operador.

Los mosaicos se construyen alineando e integrando todos los frames de una secuencia de imágenes proveniente de un video o de la captura de una cámara. Dado que las imágenes sucesivas de una escena habitualmente presentan una importante superposición, usualmente los mosaicos de imágenes proporcionan una considerable reducción en la cantidad de información visual contenida en la secuencia.

El proceso de construcción de un mosaico de imágenes tradicional, generalmente implica la realización de dos pasos; el proceso de alinear las imágenes de un secuencia y la integración de estas imágenes en una sola. Para realizar el proceso de alineación de las imágenes respecto a un sistema de referencia común es preciso definir; cuál va a ser este sistema de referencia y que tipo de transformación o modelo de movimiento se va a usar durante el proceso de registro que permita realizar el alineamiento. Los métodos utilizados habitualmente en aplicaciones de construcción de mosaicos pueden clasificarse en dos grandes grupos: métodos basados en establecimientos de correspondencias y métodos de registro basado en escala de grises.

Sin embargo, para lograr una alineación precisa de imágenes, es fundamental determinar un modelo matemático adecuado que relacione las coordenadas de los píxeles en una imagen $\mathbf{A}$ con las coordenadas correspondientes en otra imagen $\mathbf{B}$. Para manejar imágenes con diferentes resoluciones de manera eficiente, es común adoptar un sistema de coordenadas normalizadas del dispositivo.

En una imagen rectangular típica o en un cuadro de video, se establece que las coordenadas de los píxeles oscilen entre $[-1, 1]$ a lo largo del eje más largo y entre $[-a, a]$ a lo largo del eje más corto, donde a es la inversa de la relación de aspecto. De este modo, para una imagen con ancho W y alto H, la transformación que asigna las coordenadas de píxeles enteros $x = (x, y)$ a las coordenadas normalizadas del dispositivo $x' = (x', y')$ se expresa mediante las siguientes ecuaciones:

$$x' = \frac{2\bar{x} - W}{S} \tag{3.7}$$

$$y' = \frac{2\bar{y} - H}{S} \tag{3.8}$$

$$S = max(W, H) \tag{3.9}$$

Donde S es una constante de escala usada para normalizar las coordenadas de píxeles de la imagen. Este enfoque permite estandarizar la representación espacial de las imágenes, facilitando su alineación y posterior procesamiento en aplicaciones de reconstrucción tridimensional y visión artificial.

Teniendo en cuenta que si se trabaja con imágenes en una estructura pirámidal, es necesario reducir a la mitad el valor S después de cada paso de diezmado en lugar de volver a calcularlo desde $max(W, H)$, ya que los valores (W, H) pueden redondearse o truncarse de forma impredecible. Habiendo definido un sistema de coordenadas, ahora podemos describir cómo se transforman las coordenadas. Las transformaciones más simples ocurren en el plano $2D$ y se ilustran en la Tabla 3.2.

Una vez que hemos elegido un modelo de transformación adecuado para describir la alineación entre un par de imágenes y a su vez poder emparejarlas, es necesario algún método para la estimación de los parámetros. Un enfoque es desplazar o distorsionar las imágenes entre sí y ver cuántos píxeles se ajustan unos con otros. Los enfoques que utilizan la coincidencia de píxel a píxel a menudo se denominan métodos directos, mientras que los otros métodos son basados en la extracción de características, lo que implica hacer coincidir estas características para establecer una correspondencia global y luego estimar la transformación geométrica necesaria entre las imágenes y con ello realizar el *stitching* de las imágenes.

Para usar un método directo, primero se debe elegir una métrica de error adecuada para comparar las imágenes. Una vez que esto se ha establecido, se debe idear una técnica de búsqueda adecuada. La técnica más sencilla es probar exhaustivamente todas las alineaciones posibles, es decir, hacer una búsqueda completa de todas las combinaciones posibles. En la práctica, esto puede ser demasiado lento; sin embargo se han desarrollado técnicas jerárquicas de *grueso-a-fino* basadas en una estructura pirámidal de imágenes. Para obtener una exactitud de subpíxel en la alineación, a menudo se utilizan métodos incrementales basados en una expansión de la serie de Taylor de la función de imagen. Estos también se pueden aplicar a modelos de movimiento paramétrico que se describen en la siguiente sección.

Por otro lado, los métodos basados en correspondencias dependen de la detección y emparejamiento de puntos característicos entre ambas imágenes. A partir de este conjunto de correspondencias, es posible estimar la homografía que transforma los puntos de una imagen en los de la otra mediante algoritmos específicos que filtran valores atípicos.

Este método de registro basado en escala de grises hace uso directo de las funciones de la imagen para obtener los parámetros de la homografía, formulando la minimización del indice de error, frecuentemente un error cuadrático de intensidades de la imagen.

En general, para alinear dos imágenes de la misma escena, las imágenes deben mapearse en un modelo de la superficie 3D de interés. Sin embargo, para alinear imágenes capturadas desde plataformas aéreas, como drones o satélites, se puede suponer en muchos casos que el terreno es aproximadamente plano. Esta suposición es válida cuando las variaciones de elevación en la escena son pequeñas comparadas con la altura de observación, lo que permite modelar la geometría mediante una única homografía 2D sin necesidad de una reconstrucción tridimensional completa.

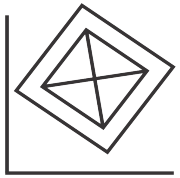
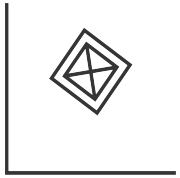
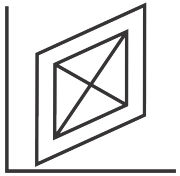
Esto permite emplear un modelo mucho más simple, en el cual dos imágenes de una superficie plana capturadas por una cámara están relacionadas mediante una transformación de perspectiva. En particular, si ambas imágenes representan el mismo plano, entonces existe una homografía, representada por una matriz 3×3, denotada como H, que satisface la siguiente relación para cualquier punto x en la primera imagen y su correspondiente x' en la segunda imagen (ambos expresados en coordenadas homogéneas):

$$Hx = x' \tag{3.10}$$

Al establecer una relación entre un plano del mundo real y una imagen, también sera posible establecer una relación entre dos imágenes diferentes del mismo plano real, esto se conoce como homografía, o transformación proyectiva, esta es la mas general de las transformaciones que mapea línea a línea. La homografía generaliza otras transformaciones más restrictivas como las transformaciones afines y de similitud, y permite modelar adecuadamente la geometría entre dos vistas de un plano observado desde distintas perspectivas.

Para muchas aplicaciones de creación de mosaicos, es apropiado un modelo más simple: para una cámara giratoria en una ubicación fija (todos los puntos están en el plano en el infinito), una simple traslación $2D$ y una rotación $1D$ son suficientes para alinear dos imágenes. Para una cámara que apunta estrictamente hacia abajo en una plataforma aérea, una transformación de similitud (rotación, traslación y escala) es suficiente. Para una cámara con un campo de visión pequeño, la transformación de perspectiva puede aproximarse con una transformación afín: $Ax + t = x'$ para una matriz de transformación (matriz 2×2) A y traslación $2D$ t.

Tabla 3.2: Clasificación de Modelos de Transformación

Transformación	Grados de Libertad	Modelo de Transformación	Ejemplo
Euclidiana	Traslaciones + Rotaciones	$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\varphi & \pm\sin\varphi \\ \mp\sin\varphi & \cos\varphi \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$	
Rígida	Euclidiana + Factor de Escala Uniforme	$\begin{bmatrix} x' \\ y' \end{bmatrix} = k \begin{bmatrix} \cos\varphi & \pm\sin\varphi \\ \mp\sin\varphi & \cos\varphi \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$	
Afin	Rigida + Factor de Escala No Uniforme	$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} k_x\cos\varphi & \pm S_x\sin\varphi \\ \mp S_y\sin\varphi & k_y\cos\varphi \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$	
Proyectiva	Afin + Proyección Perspectiva	$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} u/w \\ v/w \end{bmatrix} \Lambda \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$	

Tomada de Rodríguez Santiago (2013).

Capítulo 4

Metodología para la reconstrucción utilizando un modelo CNN

La generación de un ortomosaico u ortofotografía requiere la integración de técnicas de visión artificial y fotogrametría de manera conjunta para producir una única imagen panorámica de las zonas de interés sobrevoladas por una aeronave. Esta imagen permite obtener información detallada del área estudiada.

Como se mencionó anteriormente, la obtención de dicha fotografía se basa en el uso de técnicas de Procesamiento Digital de Imágenes y Visión Artificial. Estas técnicas permiten generar un modelo bidimensional a través del stitching de múltiples imágenes, así como un modelo tridimensional mediante la reconstrucción de nubes de puntos obtenidas a partir de las imágenes capturadas durante el vuelo del dron.

Con base en lo anterior, la ortofotografía puede obtenerse mediante una metodología tradicional (ver Figura 4.1a), la cual emplea técnicas clásicas de Procesamiento Digital de Imágenes y Visión Artificial. Sin embargo, en este trabajo se ha desarrollado una metodología alternativa (ver Figura 4.1b) que, aunque mantiene un enfoque simple, incorpora técnicas de aprendizaje profundo (*Deep Learning*) para el procesamiento de la información visual. Esto permite no solo la generación de una ortofotografía (*stitching*) o modelo bidimensional, sino también la reconstrucción tridimensional mediante nubes de puntos a partir de las imágenes aéreas capturadas.

Ambas metodologías comparten una estructura de tres etapas: (1) la adquisición de datos mediante imágenes aéreas, (2) el procesamiento de dichas imágenes y (3) la reconstrucción bidimensional o tridimensional de la zona de interés sobrevolada por el dron. La metodología basada en *Deep Learning* optimiza este proceso al mejorar la exactitud en la correlación de puntos de coincidencia y la generación de modelos más detallados.

Según la metodología tradicional propuesta (ver Figura 4.1a), tras la adquisición de la información se lleva a cabo el procesamiento de las imágenes mediante técnicas de Procesamiento Digital de Imágenes. Esta fase consta de tres etapas principales: (1) la extracción de características (Feature Extraction), (2) la correlación y corrección de características, y (3) la eliminación de características redundantes en el conjunto de coincidencias entre pares de imágenes en áreas superpuestas, extraídas en la etapa previa (Escalante Torrado et al., 2016; Lingua et al., 2009b).

Los algoritmos empleados en estas etapas se basan en la detección y manejo de características densas radiométricas presentes en las imágenes, como puntos, bordes y esquinas. Estas características permiten lograr invariancia a la rotación y mejorar la eficiencia del proceso, ya que pueden identificarse de manera consistente en imágenes adyacentes capturadas en condiciones normales.

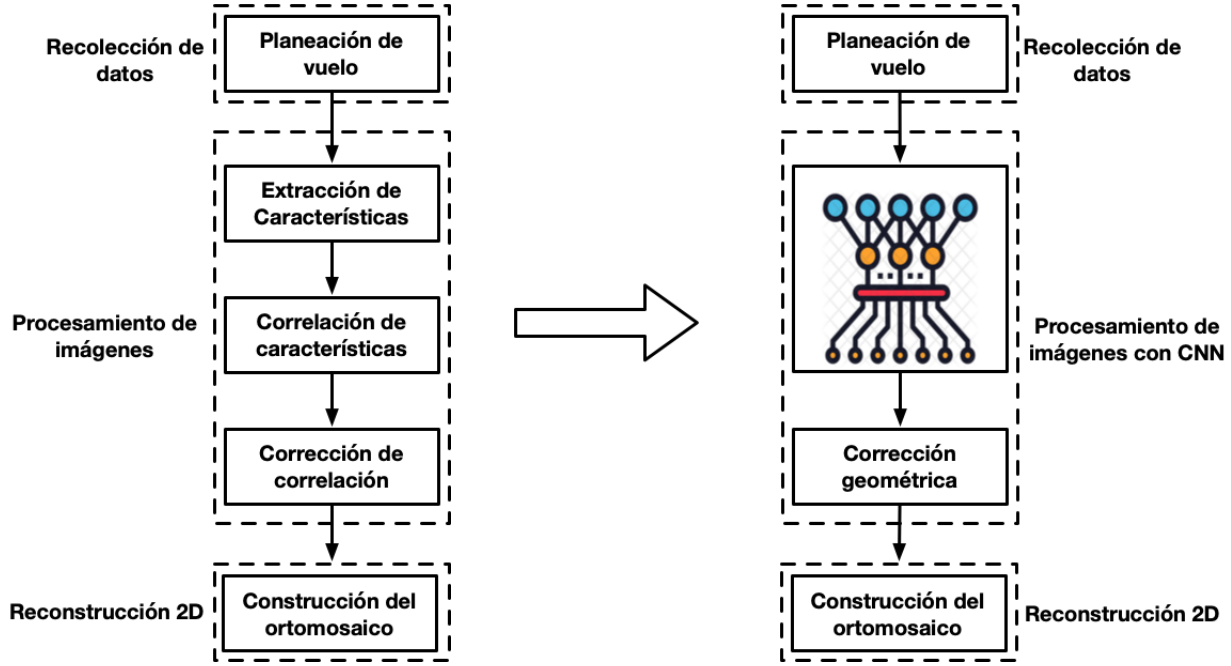


Figura 4.1: La metodología tradicional para la obtención de un ortomosaico se muestra a la izquierda (Figura 4.1a). Consta de cinco etapas, de las cuales las que representan mayor complejidad son las correspondientes al procesamiento digital de imágenes. En la figura de la derecha (Figura 4.1b), se muestra la metodología propuesta, la cual hace uso de un modelo CNN que reemplaza las etapas más complejas del procesamiento digital de imágenes.

No obstante, estas estrategias presentan limitaciones en términos de robustez ante variaciones en inclinación, rotación, escala o iluminación. Para abordar estos desafíos, se emplean algoritmos de visión por computadora con mayor capacidad de adaptación y resistencia a estos cambios. Entre las técnicas más avanzadas destacan SIFT (del inglés: *Scale-Invariant Feature Transform*) (Lowe, 1999, 2004), SURF (del inglés: *Speeded-Up Robust Features*) (Bay et al., 2008), *KAZE* (palabra japonesa que significa viento)(Alcantarilla et al., 2012; Alcantarilla and Solutions, 2011) y ORB (del inglés: *Oriented FAST and Rotated BRIEF*) (Leutenegger et al., 2011; Rublee et al., 2011), las cuales han demostrado un desempeño superior en la detección y correspondencia de características en imágenes con condiciones variables.

Una vez extraídas un conjunto relevante de características, el siguiente paso consiste en establecer la correlación o correspondencia de datos. Esta fase implica la comparación de descriptores vectoriales entre pares de imágenes que comparten información en común. Para ello, se pueden emplear diversos métodos, como la búsqueda cuadrática o el uso de estructuras de datos basadas en árboles k-dimensionales (k-d *trees*), entre otros. Las correspondencias erróneas, o valores atípicos, detectados durante este proceso se eliminan mediante la estimación de la matriz fundamental o, en caso de conocerse los parámetros internos de la cámara, mediante mediante una matriz esencial (Arya et al., 1998; Barazzetti et al., 2010).

Este proceso es particularmente complejo, ya que los parámetros internos de la cámara suelen ser desconocidos, y el presente estudio no es una excepción. No obstante, la literatura ofrece diversas estrategias para abordar este desafío, entre las que se incluyen *LMS* (*Least-Median-Square*) y *MAPSAC* (*Maximum A Posterior Estimation Sample Consensus*). Sin embargo, una de las técnicas más utilizadas es *RANSAC* (*RANdom Sample Consensus*), un algoritmo iterativo ampliamente empleado para la estimación robusta de la matriz fundamental (Fischler and Bolles, 1981; Li et al., 2005; Martínez-Otzeta et al., 2023).

RANSAC permite identificar y eliminar correspondencias erróneas entre puntos clave en las imágenes, seleccionando subconjuntos de datos que mejor se ajustan a un modelo geométrico. Su capacidad para manejar datos con alto porcentaje de ruido lo convierte en una herramienta esencial en la alineación de imágenes y en la reconstrucción tridimensional.

Esta técnica es ampliamente recomendada para ajustar correspondencias y eliminar características que no cumplen con un valor de referencia. *RANSAC* utiliza un conjunto de datos inicial lo más pequeño posible y lo amplía gradualmente, estableciendo datos consistentes cuando es viable. Por ejemplo, al ajustar un arco a un círculo a partir de un conjunto de puntos bidimensionales, *RANSAC* selecciona tres puntos (ya que con solo tres puntos es posible determinar si efectivamente forman un círculo). A partir de esto, calcula el centro y el radio del círculo, cuantificando el número de puntos que se encuentran lo suficientemente cercanos a la circunferencia para evaluar su compatibilidad, es decir, si sus desviaciones son lo suficientemente pequeñas como para ser consideradas errores de medición. Si se encuentran suficientes puntos compatibles, *RANSAC* emplea una técnica de suavizado, como los mínimos cuadrados, para calcular una estimación mejorada de los parámetros del círculo, ahora que se ha identificado un conjunto de puntos mutuamente consistentes.

La etapa final de la metodología consiste en la construcción del ortomosaico a partir de la información obtenida durante las fases previas. En esta última etapa, las imágenes capturadas, que presentan áreas superpuestas, se combinan en una única imagen panorámica. El objetivo es lograr que la unión de las imágenes sea lo más natural posible, lo cual puede ser subjetivo en cuanto a su interpretación visual, ya que la percepción de la "naturalidad"de la transición entre las imágenes puede variar dependiendo de diversos factores, como la calidad de las imágenes, el contraste y la iluminación.

En este proceso de combinación de imágenes, se utilizan técnicas avanzadas de visión por computadora que tienen en cuenta la compleja interacción de las escenas tridimensionales (3D) en la unión de imágenes bidimensionales (2D). Estas técnicas permiten estimar las transformaciones geométricas necesarias para combinar las imágenes de manera coherente, como las transformaciones de homografía. La homografía es esencialmente una transformación proyectiva que mapea un plano 3D a un plano 2D, permitiendo que las imágenes tomadas desde diferentes puntos de vista sean alineadas correctamente.

Para lograr una unión precisa y fluida de las imágenes, se estiman los parámetros de la homografía utilizando los puntos correspondientes en las regiones superpuestas de las imágenes. Estos puntos, también conocidos como "puntos de correspondencia", son cruciales para calcular la transformación que permite la alineación adecuada de las imágenes. Es importante destacar que la exactitud de estos puntos de correspondencia tiene un impacto directo en la calidad de la estimación de la homografía: cuanto más precisos sean los puntos de estimación, menor será el error en la alineación de las imágenes y, por lo tanto, se reducirá el coste computacional asociado al cálculo de la homografía.

Además, el proceso de estimación de la homografía puede beneficiarse de técnicas de optimización, como el método de los mínimos cuadrados, que ajustan los parámetros de la transformación para minimizar las discrepancias entre las imágenes superpuestas. Una estimación precisa de la homografía no solo mejora la calidad visual del ortomosaico, sino que también optimiza el tiempo de procesamiento en la generación tanto del modelo bidimensional como del modelo tridimensional. Este proceso de estimación y ajuste de la homografía, junto con la cuidadosa selección y correspondencia de puntos, es fundamental para la creación de un ortomosaico que refleje con exactitud las características de la zona de interés, lo que a su vez facilita la construcción de modelos tridimensionales precisos y eficientes.

Como se mencionó anteriormente, la tarea de extracción y correlación de características es un proceso complejo que conlleva un alto coste computacional y una considerable abstracción algorítmica. Este desafío ha sido ampliamente estudiado en la literatura, y ha motivado la búsqueda de enfoques más eficientes para el procesamiento de imágenes digitales. En este contexto, las redes neuronales convolucionales (CNN, por sus siglas en inglés: *Convolutional Neural Networks*) han demostrado ser altamente eficaces para abordar tareas similares, tales como la clasificación, segmentación y coincidencia de características en imágenes (Arandjelovic et al., 2016; Gordo et al., 2016; Radenović et al., 2016). La capacidad de las CNN para aprender representaciones jerárquicas de características a partir de grandes cantidades de datos las hace especialmente adecuadas para tareas de procesamiento de imágenes complejas que requieren un alto nivel de abstracción.

En vista de estas ventajas, para la generación del ortomosaico en esta investigación se propone la implementación de un enfoque híbrido que combine las técnicas tradicionales de procesamiento digital de imágenes con el poder de las arquitecturas profundas, específicamente las redes neuronales convolucionales. Este enfoque permite superar las limitaciones computacionales y mejorar la exactitud en la correlación de características entre las imágenes, al mismo tiempo que aprovecha las capacidades de aprendizaje automático de las CNN para adaptarse y optimizar el proceso de combinación de imágenes.

Tanto en la metodología tradicional como en la que integra redes neuronales profundas, la recolección de datos constituye una etapa fundamental. Particularmente, cuando se emplea una arquitectura basada en redes neuronales convolucionales (CNN), resulta indispensable contar con un conjunto de datos(dataset) debidamente estructurado que permita entrenar, validar y evaluar el modelo de aprendizaje automático.

La calidad, diversidad y volumen del conjunto de datos son factores determinantes para garantizar el desempeño del modelo, ya que un dataset bien diseñado no solo incrementa la capacidad del modelo para aprender representaciones significativas, sino que también mejora su robustez y capacidad de generalización frente a escenarios variables. Esto repercute directamente en la exactitud de los ortomosaicos generados, así como en la fidelidad de la reconstrucción tridimensional posterior.

En este sentido, la presente investigación da prioridad a la construcción de una base de datos sólida y representativa, que abarque una amplia gama de condiciones de iluminación, puntos de vista, alturas de captura y características texturales del terreno. Este enfoque busca asegurar que el modelo basado en CNN pueda generalizar de manera efectiva, enfrentando con éxito las variaciones inherentes a las imágenes adquiridas desde diferentes ángulos, distancias y configuraciones de captura.

4.1. Generación de la base de datos o Dataset

En lo que respecta a la recolección de datos o adquisición de imágenes del área seleccionada, uno de los enfoques más comunes es el uso de aplicaciones móviles de terceros, diseñadas específicamente para la planificación de vuelos autónomos de aeronaves no tripuladas. Estas aplicaciones permiten configurar la aeronave con las especificaciones necesarias para obtener información precisa y detallada de la zona de interés. Entre las aplicaciones comerciales más utilizadas se encuentran *Pix4D*, *DroneDeploy* y *DJIGO4*, que, además de permitir la planificación de vuelos autónomos, ofrecen herramientas de procesamiento de imágenes y generación de ortomosaicos.

Estas plataformas se integran con mapas georreferenciados, lo que posibilita la colocación de marcadores de posición en el área de estudio. A través de estas herramientas, el usuario puede configurar varios parámetros cruciales, tales como la altura de vuelo de la aeronave y el porcentaje de superposición entre las imágenes capturadas durante el vuelo. La superposición es esencial para garantizar que las imágenes adquiridas se puedan combinar de manera adecuada en el proceso posterior de generación de ortomosaicos, evitando errores de interpretación y facilitando la reconstrucción precisa del área.

Además de sus versiones comerciales, algunas de estas aplicaciones ofrecen versiones gratuitas con funciones limitadas, pero suficientes para realizar la tarea de adquisición de imágenes de forma autónoma, eficiente y confiable. Estas versiones gratuitas permiten llevar a cabo vuelos planificados con una configuración básica, lo que resulta ideal para proyectos de investigación que no requieren una infraestructura de alto costo, pero que aún así requieren resultados precisos para la reconstrucción digital de áreas geográficas.

El uso de estas herramientas contribuye significativamente a la optimización del proceso de adquisición de imágenes, garantizando que la información recolectada cumpla con los requisitos necesarios para los siguientes pasos del análisis y procesamiento de datos. Además, su integración con sistemas *GPS* y otros sensores de la aeronave permite un alto grado de exactitud en la georreferenciación de las imágenes, lo cual es fundamental para la construcción precisa del ortomosaico y la generación de modelos tridimensionales.

Para este estudio en particular, se utilizó el dron **DJI Mavic Pro** (ver Figura 4.2), perteneciente a los laboratorios de robótica de la UTM. Este dron, dentro de sus múltiples características, se destaca por contar con una cámara integrada de alta resolución y un estabilizador compacto, que le permite grabar video en calidad $4K$ a una tasa de hasta 30 fotogramas por segundo, así como capturar fotografías con una resolución de hasta 12 *megapíxeles*.

El DJI Mavic Pro, gracias a su tecnología avanzada y diseño compacto, es ideal para la captura de imágenes aéreas de alta calidad en entornos complejos, como los que se requieren en estudios de ortofotografía y generación de modelos tridimensionales.

El control del vuelo de la aeronave y la captura de imágenes se realizaron de manera autónoma mediante el uso de la aplicación móvil *Pix4DCapture* y el software comercial *Pix4DMapper*. Estas herramientas permitieron planificar y ejecutar vuelos programados, garantizando una cobertura eficiente y consistente del área de estudio. Con esta configuración, fue posible generar una base de datos que abarca todo el campus universitario, compuesto por aproximadamente $3,000$ imágenes aéreas en resolución $4K$.

La utilización de este sistema integrado de adquisición de imágenes, combinado con las capacidades del DJI Mavic Pro y las herramientas de software asociadas, proporcionó una plataforma robusta y confiable para la recolección de datos necesarios para el análisis y la posterior creación de ortomosaicos y modelos tridimensionales. Cabe destacar que, debido a las condiciones del terreno del campus universitario, la trayectoria de vuelo de la aeronave se planificó de forma poligonal (ver Figura 4.3a), definiendo tres alturas seguras de vuelo para el dron: 50 metros, 100 metros y 150 metros. Para garantizar una cobertura adecuada y la calidad de las imágenes obtenidas, se configuraron dos posibles valores para los porcentajes de traslape u overlapping: $30\,\%$ y $50\,\%$, tanto de forma longitudinal como transversal (ver Figura 4.3b).

Con estos parámetros configurados, la aplicación Pix4DCapture determinó automáticamente el número de imágenes a capturar, el tiempo de vuelo necesario y, en consecuencia, el número de baterías requeridas para completar la ruta de vuelo sobre la zona de interés. Esta planificación permitió una recolección eficiente de los datos sin interrupciones, optimizando el uso de la aeronave.

Como resultado, se generó una base de datos de imágenes aéreas del campus universitario con un total de $3,000$ imágenes, cada una con una resolución de $3,000x4,000$ *píxeles*. Estas imágenes fueron organizadas y almacenadas según las especificaciones detalladas en la Tabla 4.1.

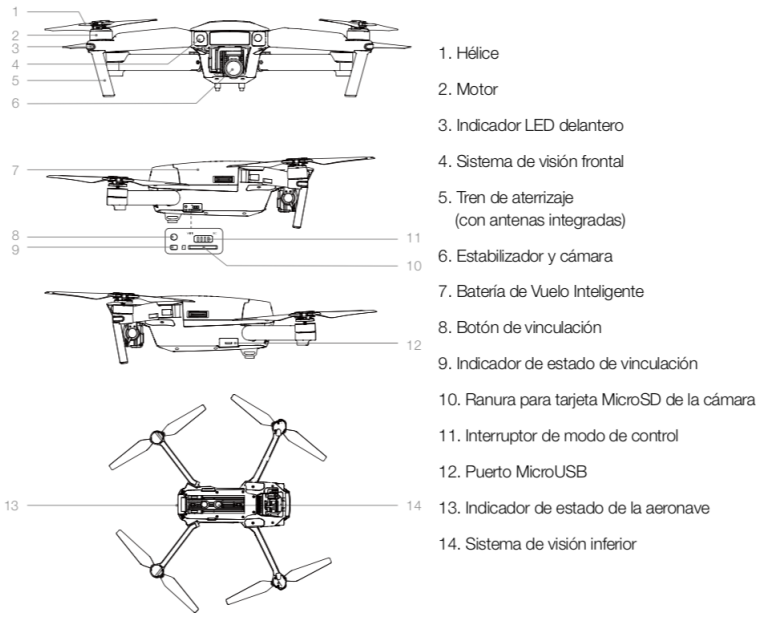
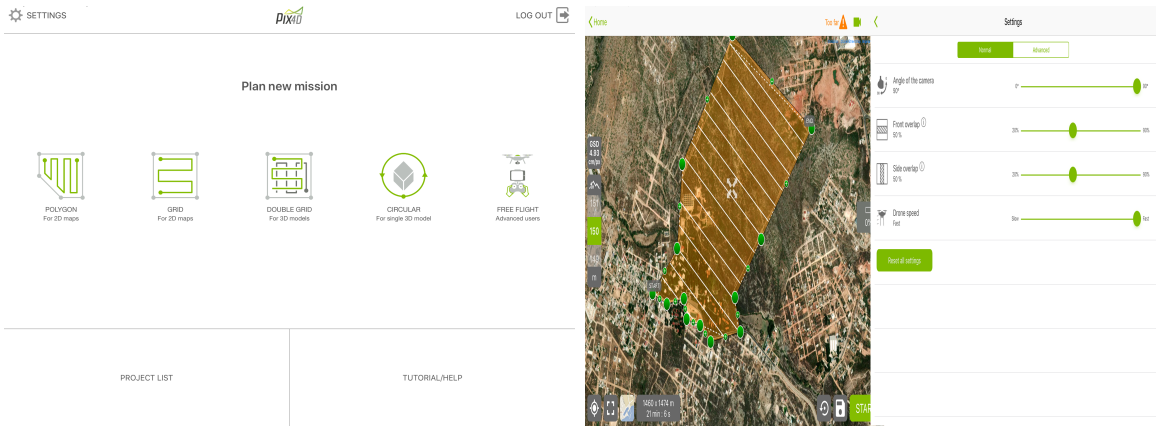


Figura 4.2: Características del DJI Mavic Pro. Se destaca la portabilidad de la aeronave y cámara de alta definición capaz de grabar video en $4K$ y capturar imágenes de 12 Megapíxeles.



(a) (b)

Figura 4.3: Configuración de la aplicación para la captura de las imágenes aéreas de la UTM de forma autónoma. Se muestran los tipos de trayectoria para el vuelo del dron (Figura 4.3a), asi como la configuración de los parámetros de vuelo y la ruta seguida por el dron en un mapa georreferenciado de la zona de interés (Figura 4.3b).

Tabla 4.1: Conjunto de datos de imágenes del campus de la UTM. Esta es la forma en que se han organizado las imágenes aéreas del terreno de la Universidad con las cuales el modelo CNN se ajustará de manera que se tenga conocimiento de estos datos y pueda trabajar con ellos.

Altura\Overlapping	**30 %x30 %**	**50 %x50 %**
50*m*.	600	1,200
100*m*.	300	600
150*m*.	100	200
Total	1,000	2,000

Capítulo 5

Reconstrucción 2D con Deep Learning

5.1. Metodología para el procesamiento de imágenes con una arquitectura CNN

Con base en la metodología propuesta en el capítulo anterior (ver Figura 4.1), la tarea siguiente después de la generación de la base de datos es el procesamiento de la información, en este caso imágenes aéreas, de manera que sea posible obtener un modelo bidimensional en una única ortofotografía panorámica de la zona sobrevolada por el dron. Para esto, según la metodología clásica, se debe realizar una extracción de un mapa de características (*feature maps*) y la correlación de las mismas de un par de imágenes con cierto porcentaje de superposición entre ellas. Esta superposición permite realizar un matching de estos mapa de características entre las imágenes, facilitando la unión de los pares de imágenes y, de forma iterativa, la formación de una única imagen panorámica que abarque toda la base de datos de la zona de interés.

Este proceso implica la identificación de puntos clave y su descripción, lo que permitirá una correcta alineación de las imágenes para generar una ortofotografía precisa. La calidad y eficiencia de este proceso dependen en gran medida de la exactitud en la extracción de las características y de la capacidad del algoritmo para gestionar las correspondencias entre las imágenes, minimizando los errores y mejorando la calidad final del mosaico generado.

Con base en lo anterior, en primer lugar, se busca diseñar una arquitectura de red neuronal profunda con la capacidad de extraer las características válidas entre un par de imágenes con cierto porcentaje de superposición entre ellas para poder realizar el matching de las mismas y, a su vez, permitir registrar este par de imágenes en una sola, es decir, realizar el *stitching*.

Se ha diseñado una arquitectura siamesa, es decir, una red neuronal compuesta por dos ramas simétricas que comparten pesos y están destinadas a realizar tareas de comparación de características entre pares de imágenes. Cada una de estas ramas se basa en una arquitectura residual profunda, siendo una de las más reconocidas y robustas la ResNet50 (He et al., 2016), ampliamente utilizada en tareas de extracción eficiente de mapas de características.

En particular, se toma como punto de partida el modelo original de ResNet50, empleando la salida del cuarto bloque convolucional como representación intermedia, con el fin de capturar características discriminativas de nivel medio y alto. Esta elección proporciona un balance adecuado entre la resolución espacial y la abstracción semántica de las características extraídas.

Posteriormente, a esta salida se le añaden una capa completamente conectada (*fully connected*) y una capa de recuperación de imágenes (*Image Retrieval Layer*), cuya función es identificar los puntos clave (*keypoints*) en ambas imágenes y establecer las correspondencias geométricas entre estos puntos. Este proceso es esencial para tareas posteriores como la estimación de homografías o la reconstrucción de escenas. La Figura 5.1 ilustra la arquitectura propuesta, mostrando el flujo de procesamiento desde la entrada de las imágenes hasta la obtención de correspondencias clave.

Esta arquitectura se propone con el objetivo de mejorar la exactitud y eficiencia en la extracción y alineación de características clave entre imágenes, lo que es fundamental para lograr una fusión adecuada de las imágenes y la construcción precisa de la ortofotografía final.

Finalmente, para la corrección geométrica, se agregaron dos capas más a la arquitectura de la red: una capa de max-pooling y una capa completamente conectada (*fully connected*). Cabe destacar que los bloques de la red han sido previamente entrenados con la base de datos ImageNet (Russakovsky et al., 2015), por lo que en este estudio se realizó un proceso de fine-tuning y transfer learning sobre los bloques del modelo original de la red ResNet50. Como parte de las modificaciones, se cambió la función de activación original, ReLU, por Parametric ReLU ($PReLU$), lo que contribuye a una mejor capacidad de adaptación y aprendizaje en tareas de visión por computadora.

Para el entrenamiento de la red, se utilizaron $2,000$ imágenes para el conjunto de entrenamiento y $1,000$ imágenes para el conjunto de validación. Para el entrenamiento, se exploraron múltiples configuraciones de entrenamiento, variando parámetros como el número de épocas, el tamaño del lote (*batch size*), la tasa de aprendizaje y la estrategia de optimización. Estas pruebas permitieron evaluar el comportamiento del modelo bajo distintas condiciones y niveles de exigencia computacional. No obstante, la combinación de 50 épocas con un *batch-size* de 40, y el uso de 2,000 imágenes para entrenamiento y 1,000 para validación fue la que ofreció los resultados más favorables en términos de desempeño y generalización, logrando una convergencia estable de la función de pérdida y un alto nivel de exactitud en la validación.

Este proceso se llevó a cabo en una estación de trabajo equipada con dos tarjetas gráficas NVIDIA RTX 2080Ti y 32 GB de memoria RAM. Tras aproximadamente doce horas de entrenamiento, se obtuvieron los siguientes resultados: una función de costo (Loss) de 0.5911 en entrenamiento y 0.1714 en validación (ver Figura 5.2a), con un Accuracy de 78.464 % en el conjunto de entrenamiento y 96.875 % en el conjunto de validación (ver Figura 5.2b).

Estos resultados indican que el modelo ha logrado una buena generalización y exactitud en la tarea de extracción de características y correspondencias, lo cual es crucial para el posterior proceso de *stitching* y generación del ortomosaico.

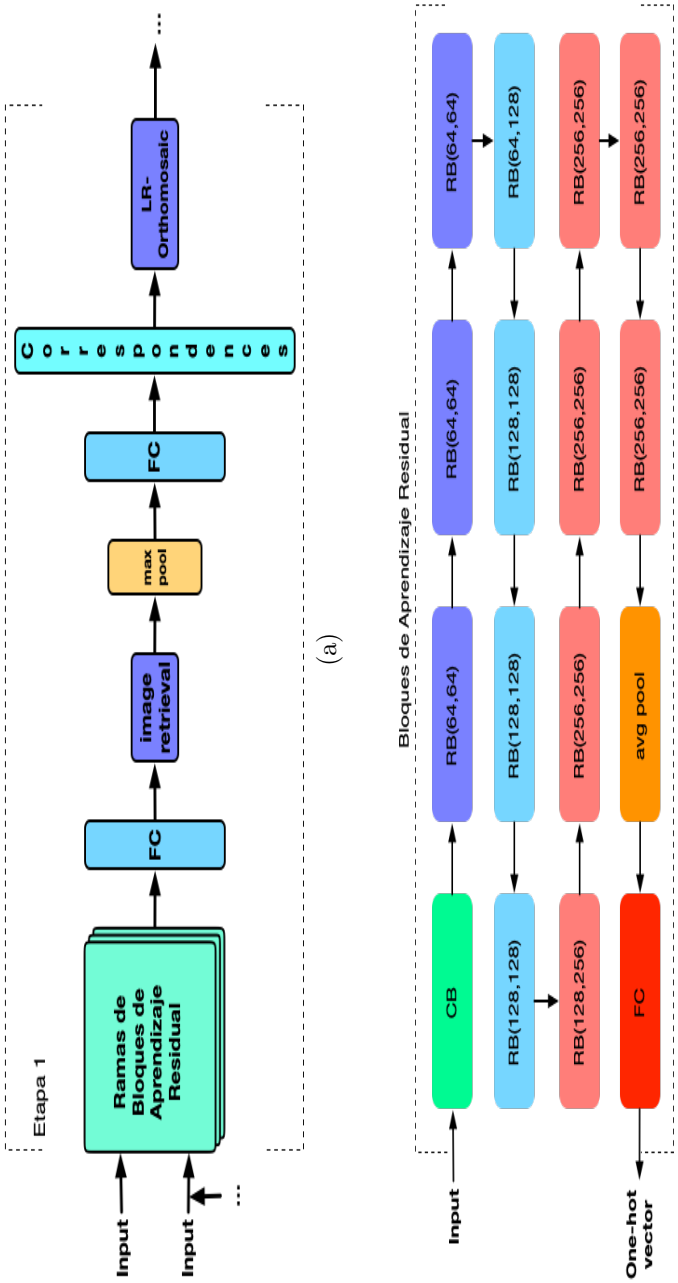
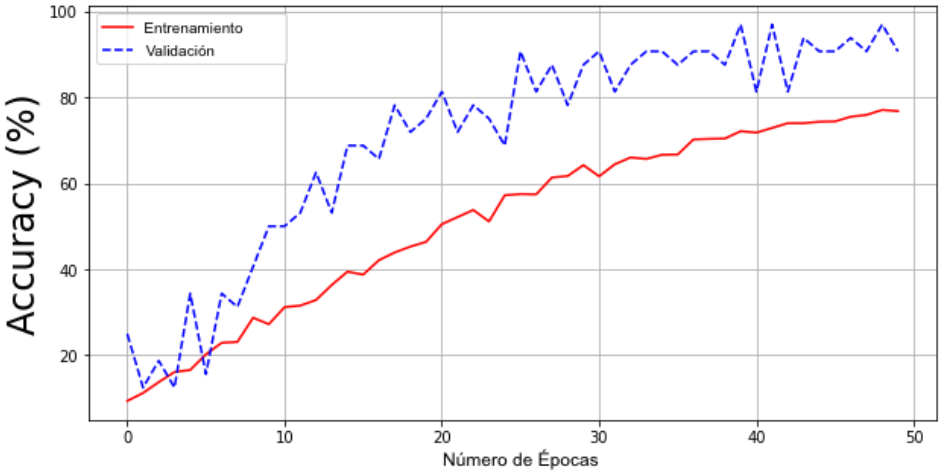
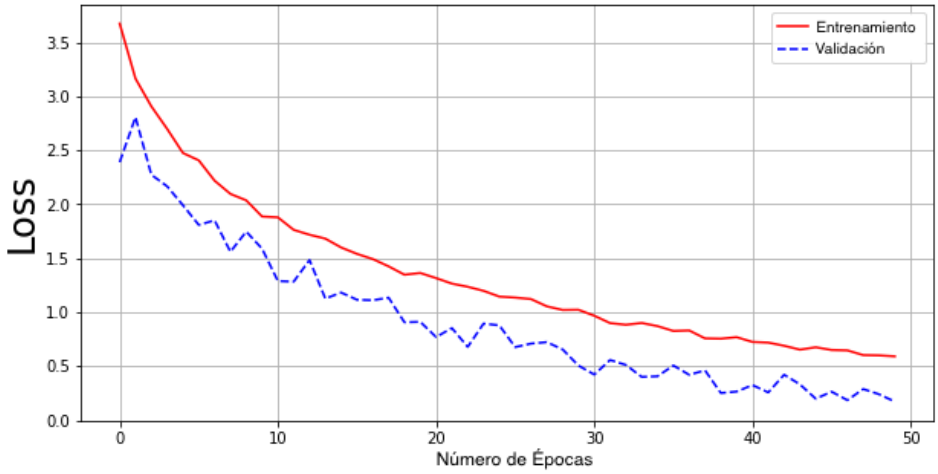


Figura 5.1: En la Figura 5.1a se muestra el esquema de la arquitectura de la red neuronal empleada para la generación de mapas de características. La red se basa en el modelo ***ResNet50***, con una estructura de doble rama que opera simultáneamente. Cada rama está compuesta por capas convolucionales, bloques residuales, capas de pooling y capas completamente conectadas, lo que permite el aprendizaje profundo y la extracción de características densas del par de imágenes de entrada. Mientras que en la Figura 5.1b se presenta la descripción detallada de la organización de los componentes de la red neuronal utilizada para la extracción de características. Se ilustran las diferentes capas que conforman la arquitectura, destacando el flujo de procesamiento desde los bloques residuales hasta la generación de los *feature maps* (tomada de Rodríguez-Santiago et al. (2021b)).



(a)



(b)

Figura 5.2: La Figura 5.2a ilustra la evolución del *Accuracy* durante el entrenamiento y validación de la primera etapa de la metodología propuesta. Se observa una tendencia creciente en la exactitud a lo largo de las épocas, lo que evidencia una mejora en el rendimiento del modelo. Por otro lado, la Figura 5.2b muestra la evolución de la función de costo (*Loss*) en el mismo proceso, donde la reducción progresiva de la pérdida sugiere una convergencia estable del modelo, reflejando una optimización efectiva de sus parámetros.

Al entrenar el modelo de CNN, la arquitectura propuesta es capaz de obtener un mapa de características y seleccionar las adecuadas para emparejar un par de imágenes. Los puntos de coincidencia obtenidos mediante esta red presentan un nivel de exactitud comparable al de algoritmos clásicos como SIFT *SIFT* (del inglés: *Scale-Invariant Feature Transform*). Este último es ampliamente reconocido por su robustez en la extracción de características y puntos clave en diversas condiciones de imagen, como el cambio de escala, rotación y ruido. No obstante, en el caso específico de imágenes aéreas, como las empleadas en este estudio, los métodos tradicionales pueden verse afectados por factores como variaciones de iluminación, distorsiones geométricas derivadas de la perspectiva, o la presencia de regiones con baja textura o características homogéneas. Estos factores pueden influir negativamente en el rendimiento de los algoritmos de coincidencia basados en características manuales. En contraste, el enfoque basado en CNN permite adaptar automáticamente la extracción de características a las propiedades del conjunto de datos utilizado, lo cual puede mejorar la capacidad del modelo para establecer correspondencias robustas en este tipo de imágenes.

Por lo tanto, se lleva a cabo una comparación entre los resultados obtenidos con *SIFT* por sí solo y los resultados cuando *SIFT* se combina con un algoritmo de *RANSAC* (del inglés: *Random Sample Consensus*). *RANSAC* se emplea para la corrección geométrica de puntos clave, ya que ayuda a eliminar las correspondencias erróneas (*outliers*) y mejora la exactitud de la estimación de las transformaciones geométricas entre las imágenes.

La combinación de *SIFT* con *RANSAC* tiene como objetivo optimizar la exactitud de las correspondencias de características, eliminando aquellos puntos que no se alinean correctamente debido a ruido o fallos en la extracción de características. Este proceso permite una mejor exactitud en la alineación y fusión de imágenes, lo cual es fundamental en la creación de un ortomosaico de alta calidad. La comparación de la arquitectura de la red neuronal profunda con estas soluciones clásicas permitirá determinar cuál de las estrategias es más efectiva en términos de exactitud y rendimiento computacional en la tarea específica de stitching de imágenes aéreas.

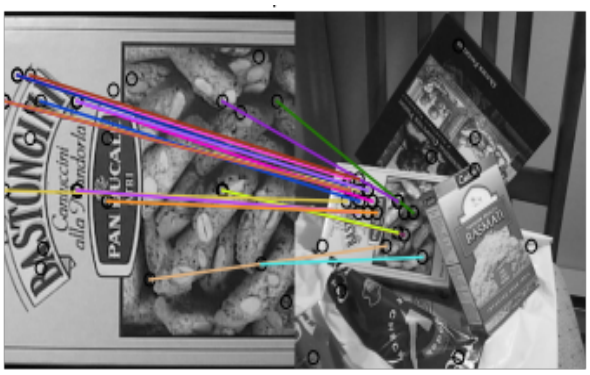
Los resultados obtenidos durante la comparación entre las correspondencias de características para el par de imágenes se presentan en las siguientes figuras. La Figura 5.3 muestra las correspondencias de características entre un par de imágenes, destacando cómo los puntos clave se alinean a lo largo de la imagen. La Figura 5.3a presenta el resultado obtenido mediante la metodología propuesta en este estudio, utilizando la red neuronal profunda para la extracción de características y la corrección geométrica.

Los puntos de coincidencia deberían mostrar una mayor exactitud y consistencia en comparación con los resultados clásicos, gracias a la capacidad de la red para aprender y adaptar los puntos clave en contextos complejos. Por otro lado, la Figura 5.3b muestra los resultados obtenidos utilizando el algoritmo clásico *SIFT*, que a pesar de ser robusto, podría mostrar algunas limitaciones en situaciones con imágenes aéreas, debido a su tendencia a generar puntos clave que no siempre se alinean correctamente debido a las variaciones en la escena, como cambios en la iluminación o patrones repetitivos.

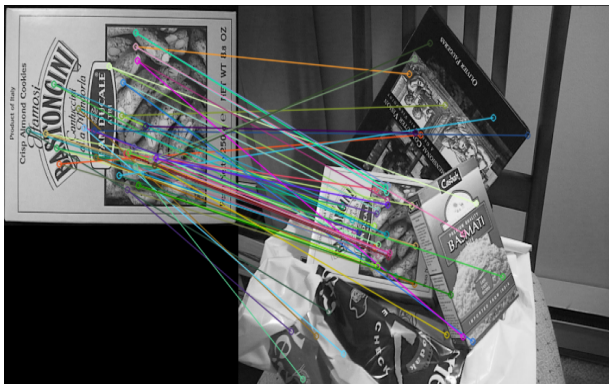
Estas figuras permiten visualizar cómo la red neuronal propuesta mejora la exactitud en la coincidencia de las imágenes, lo cual se refleja en una mayor exactitud en la creación de ortomosaicos. Además, permiten comparar los resultados obtenidos por la metodología de aprendizaje profundo con los generados por un algoritmo tradicional ampliamente utilizado como *SIFT*, y evaluar cuál de las dos técnicas produce mejores resultados en términos de alineación de las imágenes.

Los puntos característicos se combinan con éxito en la metodología propuesta; sin embargo, en un entorno desafiante como el campus de la UTM, que incluye cambios en el contraste, nitidez, brillo y rotaciones en las imágenes, la arquitectura de la red neuronal convolucional (CNN) propuesta también demuestra resultados favorables. Esto se refleja en la Figura 5.3c, que muestra los resultados obtenidos con la arquitectura de CNN en estas condiciones variables. La capacidad de la red para adaptarse a tales cambios en las condiciones de las imágenes resulta en una mayor robustez y exactitud al realizar el emparejamiento de las imágenes.

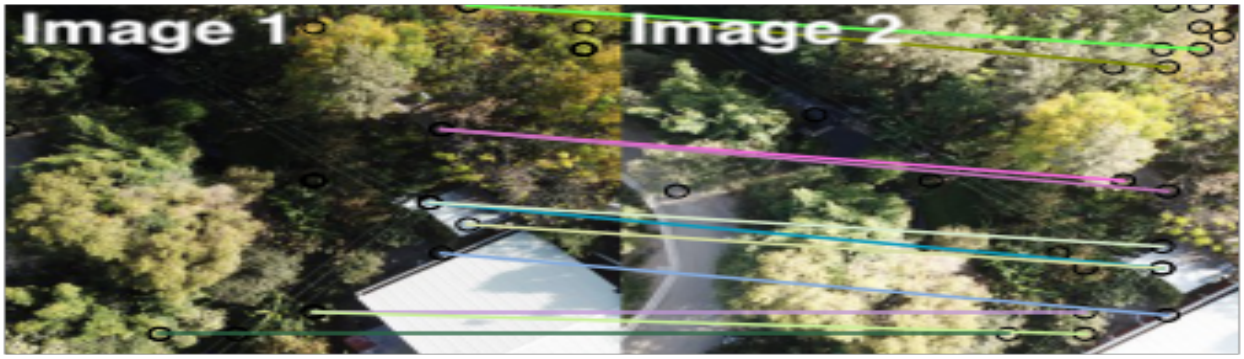
Por otro lado, en la Figura 5.3d se presenta el resultado obtenido al aplicar el algoritmo clásico *SIFT* en un entorno similar. Como se puede observar visualmente, *SIFT* genera una gran cantidad de características atípicas que podrían conducir a un emparejamiento de imágenes poco favorable o incluso erróneo. Esto se debe a que *SIFT*, aunque robusto en ciertas condiciones, tiene dificultades para manejar escenarios complejos donde existen variaciones significativas en las imágenes, como cambios de iluminación, rotaciones o desenfoques. Estos resultados destacan la ventaja de la metodología basada en CNN para tareas de correspondencia de imágenes en entornos dinámicos y desafiantes.



(a)



(b)



(c)

(d)

Figura 5.3: Comparación de resultados de extracción de mapas de características. Se muestran los resultados tanto de la aplicación de la red neuronal desarrollada (ver Figuras 5.3a y 5.3c) como del algoritmo SIFT (ver Figuras 5.3b y 5.3d).

5.2. Construcción del modelo bidimensional

Por otro lado, la red CNN propuesta presenta un emparejamiento (de puntos clave del mapa de características obtenido), adecuado para realizar las correspondencias. De esta manera, las imágenes se pueden unir de forma congruente y formar un ortomosaico visualmente aceptable y robusto, incluso en exteriores, como se muestra en la Figura 5.4. Este rendimiento es especialmente positivo, ya que la arquitectura demuestra ser eficaz en condiciones reales, donde los escenarios exteriores pueden presentar variaciones considerables en las imágenes.

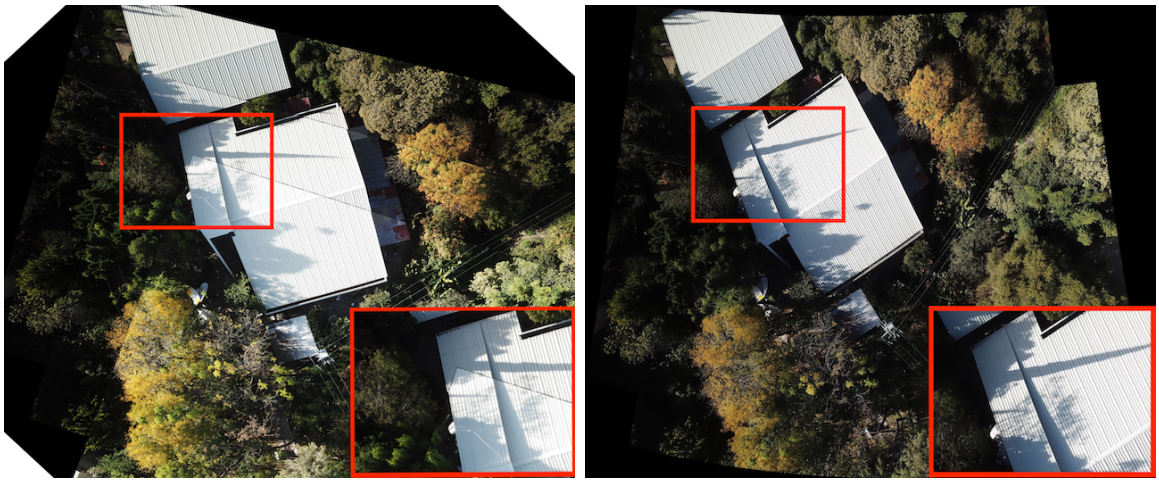
Sin embargo, al intentar emparejar múltiples imágenes, se observa una pequeña ambigüedad en la alineación, como se ilustra en la Figura 5.4a. Aunque esta ambigüedad es mínima en el emparejamiento de un par de imágenes, se propaga a medida que aumenta el número de imágenes a alinear. Es decir, conforme se agregan más imágenes al proceso de stitching, la ambigüedad de alineación aumenta, lo que puede provocar que se limite el área del stitching que representa la zona de interés o, en algunos casos, que el emparejamiento no se pueda realizar de manera exitosa.

Este desafío resalta una de las limitaciones del enfoque actual cuando se trata de grandes volúmenes de imágenes, y plantea la necesidad de desarrollar soluciones adicionales para mejorar la exactitud del emparejamiento a gran escala.

Desafortunadamente, esta ambigüedad no puede ser eliminada utilizando técnicas tradicionales de visión por computadora. Por lo tanto, se hace necesario emplear algún algoritmo especializado que permita cumplir con un consenso grupal de las correspondencias para eliminar los valores atípicos presentes y, de esta forma, mejorar el emparejamiento de pares de imágenes. Este tipo de estrategias son ampliamente utilizadas en sistemas de Localización y Mapeo Simultáneos (SLAM, por sus siglas en inglés: *Simultaneous Localization and Mapping*). Dichas estrategias se han adoptado de manera generalizada como una forma de restricción de consenso global, que facilita el proceso de cierre de bucles (*loop closing*) (He et al., 2016; Stachniss et al., 2004).

En el contexto de nuestra investigación, la aplicación de este tipo de estrategias ayudaría a cerrar el bucle de las características encontradas, permitiendo un emparejamiento más preciso y fiable. El uso de algoritmos especializados, como los implementados en sistemas SLAM, ofrece una herramienta robusta para manejar la ambigüedad en el emparejamiento de imágenes, asegurando que las correspondencias sean más confiables y se minimicen los errores que podrían surgir debido a valores atípicos, de esta manera, se lograría un *stitching* más efectivo, incluso en entornos con variaciones complejas y en escenarios que requieren la alineación de múltiples imágenes.

Por esta razón, se implementó el algoritmo Greedy Loop Closing (GLC) (Le and Li, 2019) para imponer restricciones de cierre de bucle o consenso global. Aunque su enfoque codicioso no garantiza una solución globalmente óptima, ofrece una alta eficiencia computacional al seleccionar emparejamientos localmente consistentes. Esto facilita la eliminación de ambigüedades y reduce significativamente los errores de alineación en la reconstrucción de ortomosaicos. La Figura 5.4a muestra cómo el uso del GLC mejora el proceso de alineación en entornos complejos.



(a) (b)

Figura 5.4: Aplicación del algoritmo GLC en el proceso de unión de imágenes. Al hacer zoom sobre una zona del resultado del *stitching*, se pudo observar, por un lado, la unión visible de las imágenes (ver Figura 5.4a) y la unión suavizada con la aplicación del GLC (ver Figura 5.4b)

Al realizar una inspección detallada con aumento (*zoom*) sobre una región del resultado del proceso de ensamblado de imágenes (stitching), se observa, por un lado, la unión directa entre las imágenes sin postprocesamiento (ver Figura 5.4a), y por otro, una transición suavizada entre las imágenes tras la aplicación del algoritmo Global Loop Closure (GLC) (ver Figura 5.4b).

En este caso particular, el algoritmo GLC se basa en la construcción de un grafo dirigido $G = V, E$, donde cada vértice V representa una imagen individual, y cada arista $e_k \in E$ denota una posible alineación (o emparejamiento) entre un par de imágenes. El subíndice k representa la k-ésima hipótesis de alineación entre las imágenes correspondientes. Por ejemplo, si A y B son dos imágenes candidatas a empalmarse, cada arista (A, B, k) en el grafo codifica una transformación de alineación potencial entre ellas.

La validez global de las transformaciones geométricas se evalúa mediante la restricción de cierre de bucle (loop closure), que se expresa de la siguiente manera:

$$\prod_{(A,B,k)\in l_t} T_{A,B,k} = I \tag{5.1}$$

Donde $T_{A,B,k}$ representa la transformación asociada a la alineación k-ésima entre las imágenes A y B, e I es la matriz identidad. Cuando el producto de las transformaciones a lo largo de un bucle l_t en el grafo es igual a la identidad, se considera que existe consistencia geométrica global, es decir, que las transformaciones locales entre imágenes no introducen errores acumulativos. Esto permite resolver ambigüedades, corregir errores de emparejamiento local y mejorar la calidad general del ortomosaico.

Los resultados presentados en la Figura 5.4b evidencian una mejora significativa en la calidad del stitching de las imágenes. Este proceso no solo incrementa la exactitud en la correspondencia de puntos clave entre las imágenes de entrada, sino que también garantiza una transición más fluida y coherente entre ellas, lo que se traduce en la generación de un ortomosaico más preciso y detallado.

La incorporación del algoritmo *GLC* para la aplicación de restricciones de cierre de bucle ha demostrado ser una estrategia eficaz para mitigar las ambigüedades en la alineación de múltiples imágenes. Este enfoque permite eliminar valores atípicos en el emparejamiento de características, mejorando sustancialmente la exactitud en la reconstrucción de las escenas. Dicha capacidad es particularmente relevante en entornos complejos y dinámicos, como el campus universitario, donde las variaciones en iluminación, perspectiva y calidad de las imágenes pueden dificultar la correcta alineación.

5.3. Modelo bidimensional de alta resolución

El procesamiento de imágenes mediante la red CNN implementada conlleva inevitablemente una pérdida de resolución en las imágenes de entrada. Como resultado, el ortomosaico generado presenta una calidad inferior, dado que la alineación se realiza a partir de imágenes con menor resolución en las que se han extraído los mapas de características. Esta limitación contradice el propósito fundamental de una ortofotografía, la cual debe proporcionar información visual precisa y detallada de una región de interés, con una validez comparable a la de un plano cartográfico. Por ello, es deseable obtener una imagen panorámica con la mayor resolución posible para garantizar la fidelidad y utilidad del ortomosaico en aplicaciones que requieren alta exactitud espacial.

Como respuesta a esta limitación, se desarrolla una segunda etapa en la arquitectura de la red neuronal, cuyo propósito es generar una imagen en alta resolución a partir de la ortofotografía de baja resolución obtenida en la fase previa. Para ello, se implementa un modelo de red neuronal basado en la arquitectura de Redes Generativas Adversarias (GANs, por sus siglas en inglés: *Generative Adversarial Networks*), el cual se acopla en serie a la etapa inicial de generación de la ortofotografía en baja resolución. Esta segunda fase permite mejorar la calidad de la imagen final, preservando la coherencia estructural y espacial de la escena, al tiempo que optimiza la fidelidad de los detalles en la reconstrucción de la imagen de alta resolución. Por otra parte, se ha mencionado previamente que es fundamental que la ortofotografía resultante abarque la mayor extensión posible de la zona de interés. Esto implica el procesamiento de cientos de imágenes aéreas de alta resolución. Para lograrlo, una vez completadas y acopladas en serie ambas etapas de la red neuronal, se implementa una estrategia de retroalimentación en lazo cerrado (*closed-loop feedback*).Esta arquitectura de red neuronal se presenta en la Figura 5.5.

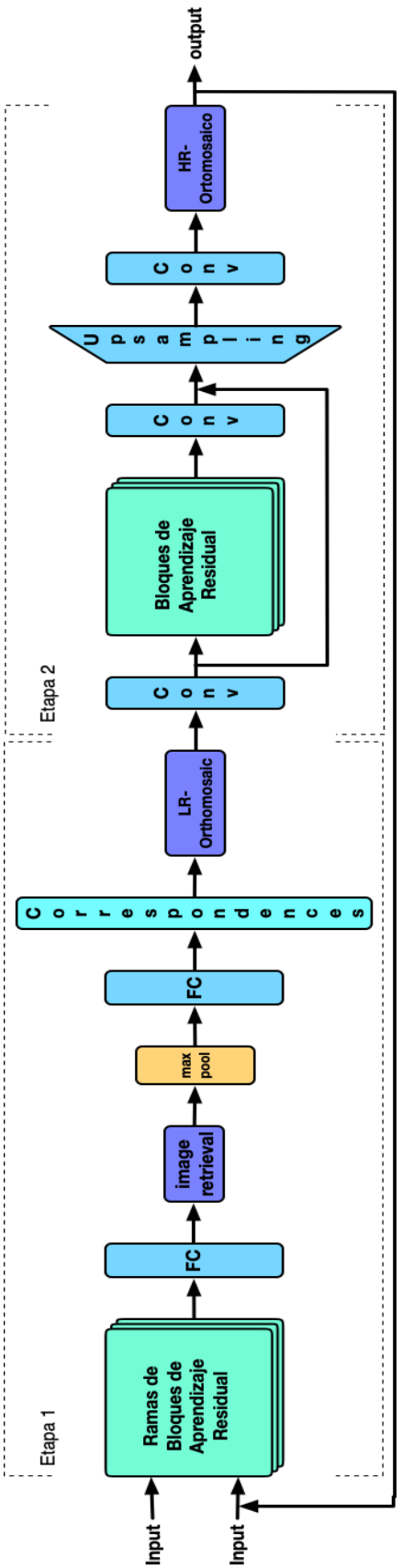


Figura 5.5: Estructura general del modelo de red neuronal convolucional propuesto. El modelo consta de dos etapas principales para la generación de un ortomosaico en Alta resolución o HR(del inglés: *High Resolution*). La primera etapa se encarga de procesar y realizar el emparejamiento de las imágenes de entrada generando un primer ortomosaico de baja resolución o LR (del inglés: *Low Resolution*). La segunda etapa se encarga de generar una imagen de alta resolución a partir de la imagen de la etapa anterior. Este resultado también se utiliza como retroalimentación para optimizar el procesamiento en la generación de un ortomosaico en HR (tomada de Rodríguez-Santiago et al. (2021b)).

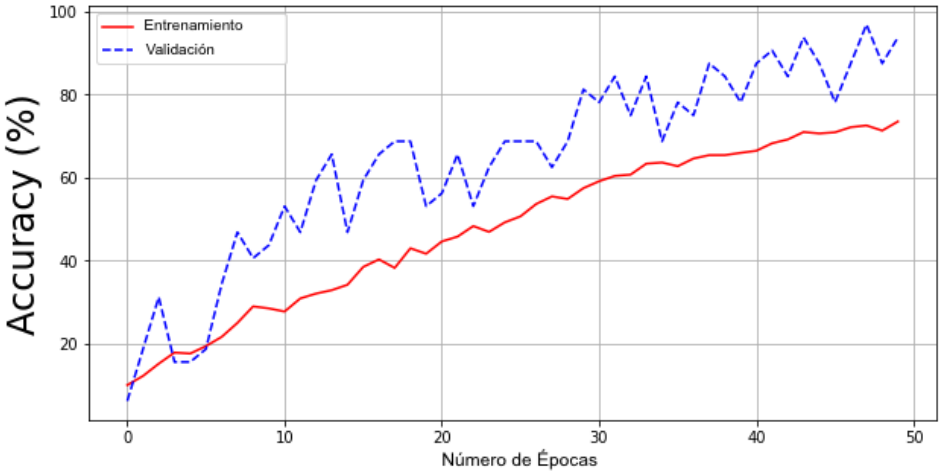
El modelo *GAN* implica el diseño e implementación de dos redes neuronales interconectadas: un Generador y un Discriminador, los cuales compiten entre sí en un proceso de entrenamiento iterativo. El Generador tiene como objetivo sintetizar imágenes que sean indistinguibles de las imágenes reales, mientras que el Discriminador busca diferenciar entre imágenes sintéticas y reales. A medida que el entrenamiento avanza, el Generador mejora progresivamente hasta alcanzar un punto en el que es capaz de "engañar" al Discriminador, generando imágenes que pueden considerarse realistas o altamente similares a las originales.

Por lo tanto, en este estudio, se adopta como base el modelo *SRGAN* (Ledig et al., 2017), diseñado para la generación de imágenes en súper resolución SR (del inglés: *Súper Resolution*), permitiendo escalar de manera eficiente una imagen de baja resolución LR (del inglés: *Low Resolution*) con un factor de aumento de hasta $X4$. Para ello, el Generador se estructura con 10 bloques residuales, incorporando conexiones de salto (*skip connections*), similares a las utilizadas en la primera etapa de la metodología.

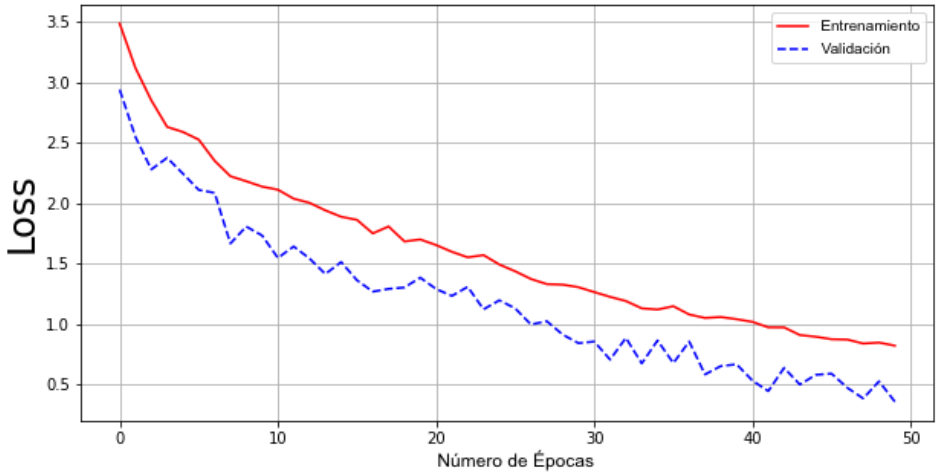
Además, se incorpora una capa de *Upsampling*, basada en la propuesta de Shi et al. (2016), con el propósito de aumentar la resolución de la imagen de entrada. Esta capa se compone de dos bloques, cada uno de los cuales incluye una capa convolucional, dos capas *PixelShuffler*, una función de activación *Parametric ReLU (PReLU)* y una capa de normalización por lotes (*batch normalization*). Cabe destacar que la capa de *Upsampling* es un elemento fundamental dentro del modelo, ya que permite la reconstrucción de imágenes de mayor resolución a partir de representaciones de menor tamaño, optimizando así la calidad visual del ortomosaico generado.

Por otra parte, el Discriminador se diseña incorporando una arquitectura compuesta por una capa convolucional inicial, seguida de una función de activación *LeakyReLU* con un coeficiente de fuga $\alpha = 0.1$. Posteriormente, se integran 10 bloques residuales, similares a los empleados en la etapa de extracción de características, los cuales están acompañados por una capa de normalización por lotes (*batch normalization*) y una función de activación *LeakyReLU*. Finalmente, se añade una capa densa, seguida de una función de activación *sigmoide*, cuya finalidad es realizar la clasificación entre imágenes reales y generadas por el modelo.

Este Discriminador desempeña un papel fundamental en el proceso de entrenamiento del Generador, ya que le proporciona una retroalimentación constante, permitiéndole mejorar progresivamente la calidad de las imágenes generadas. A medida que avanza el entrenamiento, el Generador es capaz de producir imágenes cada vez más realistas, hasta el punto en que resultan indistinguibles de las imágenes reales según la evaluación del Discriminador. Por lo tanto, con el objetivo de mejorar los resultados obtenidos, se implementa un proceso de *transfer learning* y *fine-tuning*, utilizando un conjunto de datos compuesto por $2,000$ imágenes para el entrenamiento y $1,000$ imágenes para la validación. Como resultado, se logra una función de costo (*Loss*) de 0.8203 durante la fase de entrenamiento y de 0.3574 en la fase de validación (ver Figura 5.6a). Asimismo, se obtiene un Accuracy del 73.53 % en entrenamiento y del 93.75 % en validación (ver Figura 5.6b), lo que evidencia una mejora significativa en la capacidad del modelo para generalizar sobre nuevas imágenes.



(a)



(b)

Figura 5.6: Gráfica de *Accuracy* del Discriminador en la segunda etapa de la metodología (Figura 5.6a). Se observa un incremento progresivo en la exactitud tanto en el conjunto de entrenamiento (línea roja) como en el conjunto de validación (línea azul), alcanzando un Accuracy del 93.75 % en validación, lo que indica un buen desempeño del modelo. La gráfica de la Figura 5.6b representa la función de costo (*Loss*) del Discriminador en la segunda etapa de la metodología. La función de pérdida disminuye de manera consistente tanto en el conjunto de entrenamiento (línea roja) como en el de validación (línea azul), alcanzando valores de 0.8203 y 0.3574, respectivamente. Esto sugiere que el modelo ha logrado una optimización efectiva sin presentar sobreajuste.

Gracias al entrenamiento realizado, el modelo es capaz de generar texturas detalladas y coherentes para reconstruir imágenes de alta resolución (*HR*), manteniendo la fidelidad estructural y mejorando significativamente la calidad visual en comparación con las imágenes de baja resolución (*LR*). Este proceso permite recuperar detalles finos y preservar características esenciales de la escena original, lo que resulta fundamental para aplicaciones que requieren exactitud en la reconstrucción visual.

En la Figura 5.7 se presentan los resultados obtenidos tras la aplicación del modelo en distintos tipos de imágenes. La primera fila ilustra el desempeño del modelo en imágenes sintéticas, destacando una región específica con el propósito de analizar en detalle la generación de texturas y la reconstrucción de información visual. Se observa que la red generativa adversarial (GAN) logra sintetizar detalles finos de manera convincente, lo que sugiere una mejora sustancial en comparación con la imagen original de baja resolución.

Por otro lado, la segunda fila ilustra los resultados obtenidos en imágenes provenientes de nuestra base de datos, capturadas en escenarios reales. Estos resultados evidencian que el modelo no solo es capaz de generar texturas adecuadas en entornos controlados, sino que también logra reconstrucciones satisfactorias en condiciones desafiantes, como aquellas presentes en diversas áreas del campus universitario, donde factores como variaciones de iluminación, sombras, cambios en la perspectiva y presencia de ruido pueden afectar la calidad de las imágenes de entrada.

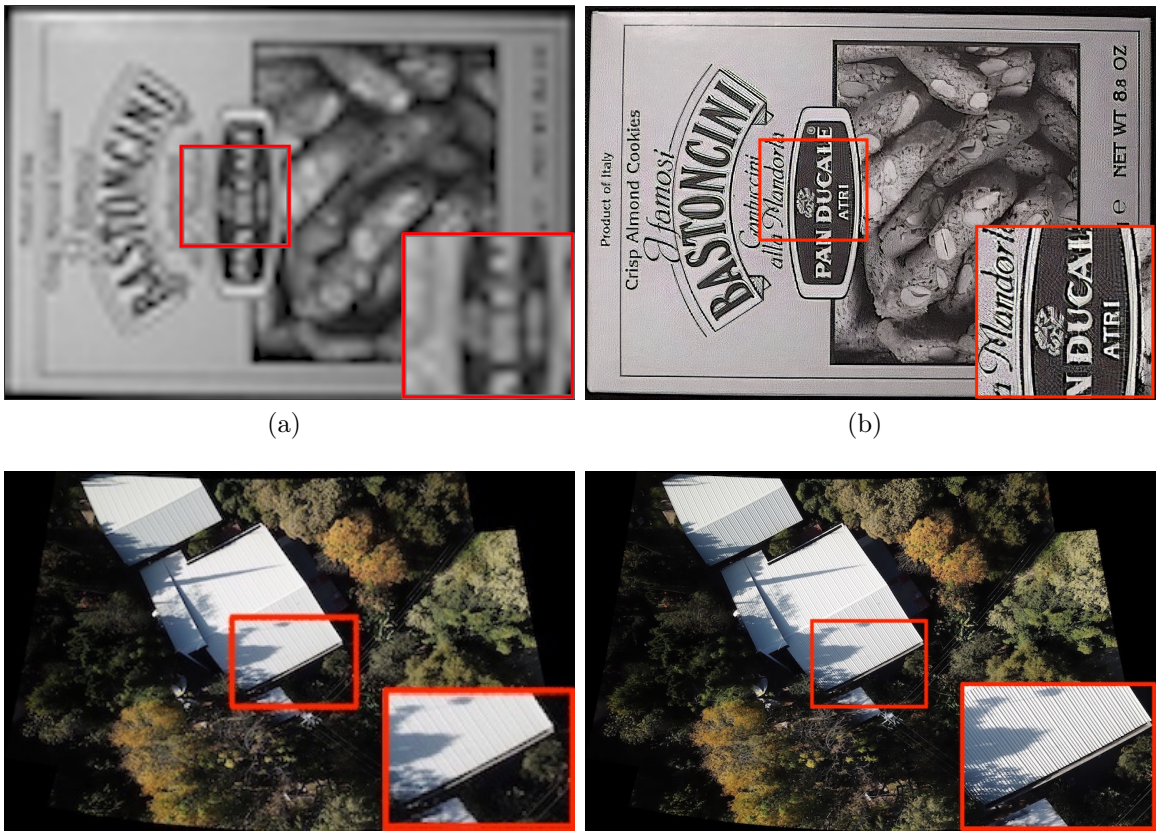
El análisis cualitativo de los resultados sugiere que el modelo ha logrado una mejora considerable en la reconstrucción de detalles de alta frecuencia, lo que se traduce en imágenes más nítidas y con un alto nivel de realismo. Esto demuestra que la combinación de *transfer learning* y *fine-tuning* ha permitido optimizar el desempeño del modelo, logrando que la red *GAN* sea capaz de generar imágenes con características visuales cercanas a las originales de alta resolución.

Por otra parte, resulta esencial que la ortofotografía generada cubra la máxima extensión posible de la zona de interés, lo que implica procesar cientos de imágenes aéreas de alta resolución. Para ello, una vez completadas y acopladas en serie ambas etapas de la red neuronal, se implementa un mecanismo de retroalimentación en lazo cerrado (*closed-loop feedback*).

Este mecanismo posibilita la integración progresiva y continua de las imágenes, de modo que cada nuevo empalme (image stitching) se realiza de forma iterativa entre una imagen de entrada y la composición parcial obtenida en el paso anterior. Gracias a esta arquitectura iterativa, el sistema escala a volúmenes significativamente mayores de imágenes, sin sacrificar la coherencia geométrica ni la continuidad visual del ortomosaico.

Como resultado, se obtiene un ortomosaico de alta resolución que cubre integralmente el área sobrevolada, con tiempos de procesamiento optimizados y un desempeño robusto frente a las variaciones de la escena. Este enfoque demuestra su aplicabilidad en entornos reales, especialmente cuando el número de imágenes es elevado y se requiere una construcción visual precisa y eficiente.

En las pruebas de validación se trabajó sobre zonas de aproximadamente $22,500m^2$. Las imágenes fuerón capturadas a $100m$ con resolución $4K$ lo que proporciona un alto nivel de detalle. Durante la integración se detectó que la primera etapa presentaba limitaciones al procesar más de 100 imágenes, debido al incremento en el coste computacional y el riesgo de errores acumulativos. El lazo cerrado permitió superar este cuello de botella, garantizando eficiencia y calidad en todo el conjunto de datos.



(a) (b)

(c) (d)

Figura 5.7: Generación de imágenes de alta resolución (HR) a partir de imágenes de baja resolución (LR). Los resultados presentados evidencian la capacidad del modelo propuesto para reconstruir imágenes de alta resolución (HR) a partir de imágenes de baja resolución (LR). En las Figuras 5.7a y 5.7c se muestran las imágenes LR utilizadas como entrada, correspondientes a imágenes sintéticas y reales, respectivamente. Por su parte, las Figuras 5.7b y 5.7d presentan las imágenes HR generadas por el modelo. Se observa una mejora significativa en la definición de detalles y texturas, lo que resalta la efectividad del enfoque basado en aprendizaje por transferencia (*transfer learning*) y ajuste fino (*fine-tuning*) para la reconstrucción de imágenes de alta resolución.

Capítulo 6

Generación del modelo tridimensional

Hasta este punto, la metodología basada en redes neuronales profundas ha demostrado ser capaz de generar ortomosaicos de alta resolución a partir de una secuencia de imágenes aéreas capturadas en extensas áreas geográficas. La arquitectura propuesta para la reconstrucción 3D sigue el diseño de un Autoencoder estructurado en tres etapas principales: el Encoder, el Bottleneck y el Decoder (véase Figura 6.1).

A diferencia de los Autoencoders convencionales basados en bloques convolucionales, en la presente propuesta se han implementado bloques residuales, lo que permite mejorar la propagación del gradiente y optimizar el proceso de extracción de características. La etapa del Encoder está conformada por una red de cuatro bloques residuales, diseñados específicamente para extraer mapas de características a partir de imágenes aéreas de entornos exteriores.

Por otro lado, la etapa del Decoder se ha concebido como una red generativa adversarial (GAN), denominada GAN-Decoder, la cual ha sido entrenada para la generación de nubes de puntos, permitiendo así una reconstrucción más precisa y detallada del terreno representado en las imágenes de entrada.

La arquitectura de red desarrollada utiliza como entrada una secuencia de imágenes aéreas bidimensionales (2D). En la etapa del Encoder, se lleva a cabo la extracción de un vector de características que representa de manera descriptiva cada imagen de entrada. Posteriormente, en la etapa del GAN-Decoder, se genera una nube de puntos a partir de la información obtenida en la fase previa.

El uso de una secuencia de fotogramas con cierto porcentaje de superposición entre imágenes consecutivas permite estimar con exactitud la ubicación espacial de cada punto generado, facilitando así la reconstrucción de la geometría del entorno. Los experimentos realizados demuestran que la metodología propuesta permite generar representaciones tridimensionales precisas de áreas sobrevoladas por el dron. En particular, las nubes de puntos generadas han demostrado ser eficaces en la reconstrucción de paisajes urbanos bajo condiciones desafiantes, como las que presentan los terrenos de la UTM.

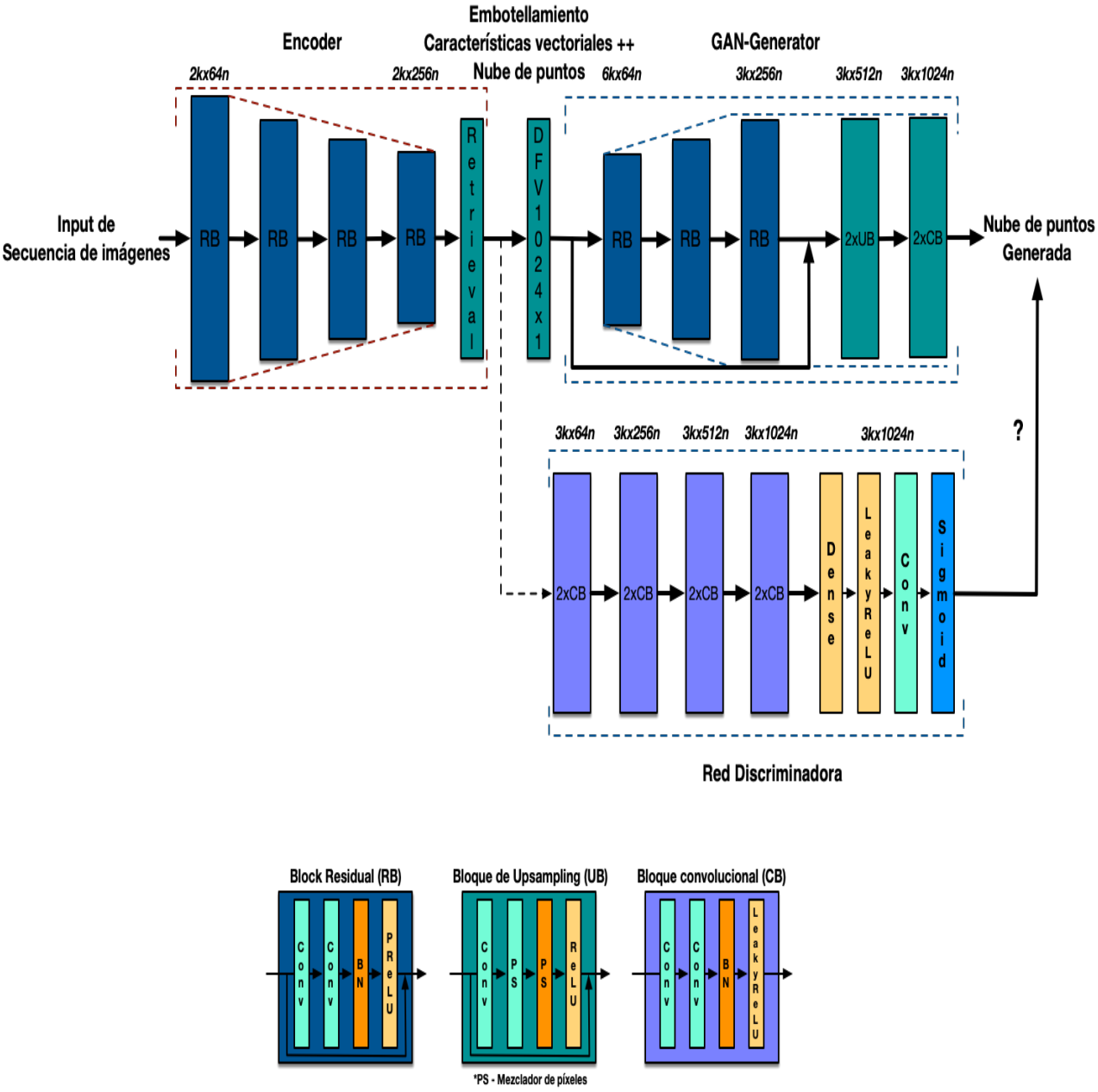


Figura 6.1: Se muestra en detalle la configuración general del modelo de red neuronal profunda, propuesto para la generación de modelos tridimensionales. La configuración de las capas internas de cada etapa de la arquitectura, tanto el Encoder como el Decoder, el cual hemos denominando GAN-Decoder. Los bloques especiales utilizados, como el bloque residual (RB) y el bloque de muestreo superior (UB), y el bloque convolucional (CB) se muestran en detalle en la parte inferior (tomada de Rodríguez-Santiago et al. (2021a)).

6.1. Encoder

La etapa del Encoder se configura mediante cuatro bloques residuales, cada uno de los cuales ha sido diseñado con dos capas convolucionales sucesivas. Estas capas están seguidas por una capa de *batch normalization* para mejorar la estabilidad y la convergencia del entrenamiento, y una capa de activación *Parametric ReLU* (*PReLU*) (He et al., 2015a), que permite una mayor flexibilidad en la función de activación al aprender los parámetros de pendiente negativa durante el entrenamiento.

Cada bloque residual sigue la arquitectura típica de los *ResNet*, donde las salidas de las capas convolucionales se suman directamente con las entradas del bloque, lo que permite un mejor flujo de gradientes y mitiga el problema del desvanecimiento del gradiente en redes profundas. Esto favorece la extracción eficiente de características a diferentes niveles de abstracción, preservando las propiedades de las imágenes a medida que pasan a través de las capas de la red. La función de activación *PReLU* es preferida sobre la tradicional *ReLU* debido a su capacidad para adaptarse a características más complejas y no lineales en los datos, permitiendo una mayor capacidad de generalización en tareas como la reconstrucción de mapas de características. A través de estas capas convolucionales, se extraen mapas de características densas que capturan información relevante sobre los patrones espaciales y texturales presentes en las imágenes aéreas de entrada.

La configuración específica de estas capas, como se muestra en la Figura 6.1, facilita la extracción de características de alta calidad, lo cual es crucial para el posterior procesamiento en la etapa del GAN-Decoder y la generación precisa de la nube de puntos tridimensional.

Para generar correctamente la geometría tridimensional del entorno, se añade una capa completamente conectada, FC(del inglés: *Fully Connected*), cuya función es integrar la información extraída a través de las etapas previas y proporcionar una representación compacta y coherente de las características. Adicionalmente, se incorpora una capa de recuperación de imágenes (*Image Retrieval*), cuyo propósito es mejorar la correspondencia entre las imágenes de entrada mediante la búsqueda de características similares dentro de la base de datos de imágenes, facilitando así la generación precisa de la nube de puntos.

Además, para la identificación de puntos clave y la obtención de correspondencias de puntos entre cada fotograma de la secuencia de imágenes de entrada, se añade una capa de *max-pooling*, la cual realiza una operación de reducción espacial para obtener una representación más abstracta y robusta de las características visuales. Esto permite identificar de manera más efectiva los puntos relevantes en las imágenes que serán utilizados para la posterior alineación y reconstrucción geométrica. Finalmente, se incorpora una capa *Fully Connected* (FC) adicional, que aplica una corrección geométrica para alinear y ajustar los puntos clave y las correspondencias a una representación coherente, asegurando la exactitud y consistencia de la reconstrucción tridimensional.

El Encoder ha sido diseñado con capas residuales que, en comparación con las capas convolucionales tradicionales, permiten una extracción más eficiente de características densas a partir de las imágenes aéreas. Las capas convolucionales residuales son especialmente útiles para describir de manera más detallada y precisa los objetos presentes en la escena objetivo, al facilitar un mejor flujo de gradientes a través de las capas, lo cual optimiza la capacidad de aprendizaje de la red.

El vector de características extraídas por el Encoder, denominado *Dense Feature Vector* (DFV), tiene un tamaño de $1 \times 1 \times 1024$. Este vector, que encapsula las características relevantes de la imagen, se concatena con una nube de puntos, formando una representación más completa y precisa del entorno. Esta concatenación se utiliza en la etapa de entrenamiento para proporcionar a la red una mayor cantidad de información sobre las relaciones espaciales entre los puntos clave, mejorando la capacidad de la red para generar la reconstrucción tridimensional de la escena.

6.2. Decodificador GAN

La red generadora propuesta se basa en una arquitectura de red generativa adversarial (GAN), que consta de dos componentes principales: una red generadora (*Generator Network*) y una red discriminadora (*Discriminator Network*). Durante el proceso de entrenamiento, un vector de características extraído en la etapa del Encoder se concatena con cada punto de una nube de puntos inicial uniformemente espaciados. Esta operación genera un nuevo vector de características de tamaño 1024×1. La Figura 6.1, ilustra el proceso de mejora y concatenación.

Este nuevo vector se utiliza como entrada para la red GAN-Generator. El generador consiste en tres capas completamente conectadas (*Fully Connected, FC*), seguidas de una capa de activación *ReLU*, lo que permite modelar relaciones no lineales y aumentar la capacidad de aprendizaje de la red. La arquitectura del generador culmina con una capa completamente conectada y una capa de *max-pooling*, que predice la nube de puntos final con una forma de 1024×3, representando la geometría 3D de la escena.

Al utilizar una nube de puntos inicial acompañada de mapas de características reales extraídos a partir de imágenes aéreas, el GAN-Decoder (decodificador GAN) es capaz de realizar inferencias más precisas sobre la nube de puntos final. Esta configuración permite que el decodificador GAN realice una generación más robusta y realista de nubes de puntos, mejorando la reconstrucción tridimensional del entorno.

6.2.1. Red Generadora

El núcleo de la red generadora propuesta, como se ilustra en la Figura 6.1, emplea bloques residuales idénticos a los utilizados en la etapa del Encoder. Estos bloques residuales permiten una extracción más eficiente y precisa de características espaciales, favoreciendo la generación de nubes de puntos de alta calidad. Para mejorar la exactitud del vector de nube de puntos generado, se incorporan dos capas completamente conectadas (*Fully Connected, FC*) y dos capas de *max-pooling* en el bloque de convolución de subpíxeles, lo que facilita una mejor integración y reducción de la información espacial. Estas capas son entrenadas conforme al modelo propuesto por Shi et al. (2016), lo cual contribuye a la mejora de la resolución y coherencia de los puntos generados.

Además, en este enfoque, se utiliza una base de datos de nubes de puntos para incrementar el conocimiento del modelo propuesto, permitiendo que la red generadora aprenda patrones y relaciones espaciales de manera más robusta, lo que optimiza su capacidad para generar nubes de puntos más precisas y realistas en el contexto de reconstrucción tridimensional a partir de imágenes aéreas.

6.2.2. Red Discriminadora

La *Discriminator Network* tiene como objetivo discriminar entre nubes de puntos reales y generadas. Para ello, sigue las pautas arquitectónicas descritas por Ledig et al. (2017) y Goodfellow et al. (2014), adaptadas a las características del problema específico. La red emplea una función de activación *LeakyReLU* con un parámetro $\alpha = 0.3$, lo cual permite un flujo de gradientes más efectivo durante el entrenamiento al evitar la saturación de la función de activación. A diferencia de otros enfoques, se evita el uso de *max-pooling* a lo largo de toda la red, lo que contribuye a preservar la resolución espacial de las nubes de puntos a medida que la red discrimina entre las muestras reales y generadas.

Finalmente, se utiliza la función sigmoidea para normalizar la salida del discriminador, produciendo una probabilidad que indica si la nube de puntos es real o generada. Esta configuración no solo reduce la complejidad computacional del modelo, sino que también mejora el tiempo de procesamiento, permitiendo una discriminación más rápida y eficiente de las nubes de puntos durante el entrenamiento del modelo GAN.

6.3. Detalles de implementación

El desarrollo del modelo propuesto, incluyendo el proceso de *fine-tuning* y *transfer learning*, se llevó a cabo utilizando las bibliotecas *Keras* y *TensorFlow* (Abadi et al., 2016). El Encoder fue entrenado utilizando un conjunto de datos previamente generado, el cual incluye un total de $2,000$ imágenes aéreas distribuidas de acuerdo con el esquema detallado en la Tabla 6.1. Es importante señalar que las imágenes aéreas de esta base de datos fueron capturadas a lo largo de una trayectoria circular (*Circular Mission*) alrededor del área objetivo. Este enfoque de captura proporciona múltiples puntos de vista desde diferentes ángulos del área sobrevolada por el dron, lo que permite obtener una perspectiva de 360° del entorno de interés, optimizando la cobertura espacial y mejorando la exactitud en la reconstrucción tridimensional del área.

La función *Circular Mission*, disponible en la aplicación DJI GO 4 para el DJI Mavic Pro, facilita la ejecución de vuelos autónomos alrededor de un punto de interés a lo largo de una trayectoria circular, lo cual es particularmente útil para capturar imágenes aéreas con un amplio ángulo de visión. Esta funcionalidad es clave ya que permite obtener una cobertura completa del área desde diversas perspectivas. Durante la misión, el dron mantiene una altitud y un radio preestablecidos, lo que asegura que las imágenes capturadas provengan de diferentes ángulos pero manteniendo una perspectiva coherente del área sobrevolada.

La utilización de múltiples puntos de vista obtenidos a partir de una trayectoria circular del área sobrevolada por el dron contribuye significativamente a la calidad y exactitud de la reconstrucción tridimensional. Como se ha demostrado en estudios previos, la recopilación de imágenes con una cobertura completa mejora la fidelidad de los modelos generados, especialmente cuando se combinan con técnicas avanzadas de procesamiento de imágenes y redes neuronales (Fei et al., 2022; Guo et al., 2020; Mohammadi et al., 2019; Schlosser et al., 2020).

Tabla 6.1: Número de imágenes capturadas por configuración experimental. Se muestra la cantidad de imágenes obtenidas para cada combinación de parámetros empleados durante los vuelos de adquisición. Las imágenes fueron capturadas siguiendo trayectorias circulares bajo dos configuraciones distintas de altura de vuelo y porcentaje de superposición. Estas combinaciones se aplicaron en diversas zonas de prueba dentro del área experimental, con el propósito de evaluar el desempeño del sistema de reconstrucción en condiciones variables. Cada conjunto de datos corresponde a un escenario independiente, y su aplicación en múltiples sectores dentro del entorno experimental contribuye a una validación más amplia y a una evaluación robusta del modelo propuesto.

Altura\Overlapping	**30 %x30 %**	**50 %x50 %**
100*m*.	300	700
150*m*.	300	700
Total	600	1,400

De esta forma, es posible realizar una reconstrucción más detallada del área de interés, incluso en escenarios que incluyen múltiples objetos y paisajes complejos. El modelo fue entrenado hasta que la exactitud de validación dejó de mejorar, lo que indicó la convergencia del proceso de entrenamiento. Para realizar el *fine-tuning* y *transfer learning*, se utilizaron 1,500 imágenes para el entrenamiento y 500 para la validación. El entrenamiento se ejecutó durante 50 épocas, con un *batch size* de 20.

El proceso de entrenamiento se llevó a cabo en una estación de trabajo equipada con dos tarjetas gráficas NVIDIA RTX 2080Ti, lo que permitió una mayor capacidad de procesamiento paralelo, fundamental para el manejo de grandes volúmenes de datos y la complejidad de las redes neuronales involucradas. El sistema operativo utilizado fue *Ubuntu 19.04*, y la estación contaba con 32 GB de memoria RAM, lo que garantizó un rendimiento adecuado durante todo el proceso de entrenamiento. Este entorno de trabajo proporcionó los recursos necesarios para alcanzar la exactitud deseada en la validación del modelo.

El entrenamiento del modelo se realizó en tres etapas. En la primera etapa, se entrenó el Encoder durante aproximadamente 24 horas, obteniendo una función de costo *Loss* de entrenamiento y validación de 0.623 y 0.219, respectivamente (ver Figura 6.2a). En cuanto a la *Accuracy*, se alcanzaron valores del 80.25 % en entrenamiento y 93.75 % en validación (ver Figura 6.2b). Estos resultados indican una correcta convergencia del modelo en la etapa inicial y un buen desempeño en la generalización a datos no vistos.

En la segunda etapa, se entrenó el GAN-Decoder, utilizando el *optimizador Adam* (Kingma and Ba, 2015). Durante esta fase, se actualizó alternativamente la red del Generador y la del Discriminador. Dado que el Generador emplea bloques convolucionales con *skip connections*, similares a los utilizados en el modelo *ResNet* e idénticos a los implementados en la etapa del Encoder, se optó por utilizar los bloques y sus correspondientes pesos obtenidos después del entrenamiento del Encoder. Esta decisión permitió aprovechar el conocimiento ya aprendido por el Encoder, optimizando el proceso de entrenamiento del GAN-Decoder y mejorando la calidad de la reconstrucción de las nubes de puntos generadas.

Adicionalmente, siguiendo el trabajo presentado en Goodfellow et al. (2020), las Redes Discriminadoras se entrenan utilizando los puntos clave obtenidos de las imágenes aéreas generadas por la etapa del Encoder. Este entrenamiento se lleva a cabo empleando la función de maximización que se describe en la Ecuación 6.1, lo que permite optimizar el proceso de discriminación entre las nubes de puntos reales y las generadas, mejorando así la exactitud del modelo en la etapa de inferencia.

$$\begin{aligned} min_G max_D E(D,G) = E_{P^{out} \sim p_{train}(P^{out})}[log D_{\theta_D}(P^{out})] + \\ E_{P^{input} \sim p_G(P^{input})}[log(1 - D_{\theta_D})(G_{\theta_G}(P^{input}))] \end{aligned} \tag{6.1}$$

Donde p_G es la distribución del generador sobre los datos de entrada P, y el generador G_{θ_G} se parametriza por sus pesos y sesgos específicos, indicados por θ_G. Por otro lado, D representa el discriminador, con una distribución D_{θ_D} que representa la probabilidad de que un punto provenga de los datos generados por G en lugar de los datos reales p_G. El discriminador D se entrena para maximizar la probabilidad de clasificar correctamente las muestras reales, mientras que minimiza la probabilidad de clasificar incorrectamente las muestras generadas por G.

Además, para restringir el rango de salida del discriminador y mejorar la estabilidad del entrenamiento, proponemos utilizar una activación *sigmoide* al final del discriminador. Esta técnica ha demostrado ser útil para estabilizar el entrenamiento en nuestros experimentos, especialmente en el contexto de los puntos de entrada residuales y la nube de puntos generada, como se reporta en estudios previos (Hui et al., 2019; Ni et al., 2018; Wang et al., 2020; Wu et al., 2019). Con las configuraciones previamente descritas, se logra una reducción en la complejidad computacional del modelo y una mejora significativa en el tiempo de procesamiento. Una vez finalizado el entrenamiento del discriminador dentro del módulo GAN-Decoder, los resultados obtenidos muestran una pérdida final de 0.647 en el conjunto de entrenamiento y 0.338 en el conjunto de validación (ver Figura 6.2c). Asimismo, la exactitud alcanzada fue del 78.04 % en entrenamiento y del 90.63 % en validación (ver Figura 6.2d).

Estos valores indican que el modelo propuesto es capaz de generar nubes de puntos con alta exactitud, diferenciando de manera efectiva entre las representaciones reales y las generadas. La estabilidad del entrenamiento y la capacidad de generalización del modelo se ven favorecidas por el uso de arquitecturas residuales en el Encoder y por la implementación de la red GAN en la etapa de reconstrucción.

Finalmente, en la tercera y última etapa del proceso de entrenamiento, se llevó a cabo la optimización de la arquitectura completa del modelo, integrando tanto el Encoder como el GAN-Decoder. Tras el ajuste de hiperparámetros y la ejecución del entrenamiento, se obtuvo una pérdida de 0.673 en el conjunto de entrenamiento y 0.237 en el conjunto de validación (ver Figura 6.2e).

En cuanto a la exactitud, los resultados finales muestran un desempeño del 76.68 % en el conjunto de entrenamiento y del 96.88 % en el conjunto de validación (ver Figura 6.2f). Estos valores reflejan una mejora significativa en la capacidad del modelo para reconstruir nubes de puntos de alta fidelidad a partir de imágenes aéreas, destacando la robustez del enfoque propuesto y su capacidad de generalización en distintos escenarios de prueba.

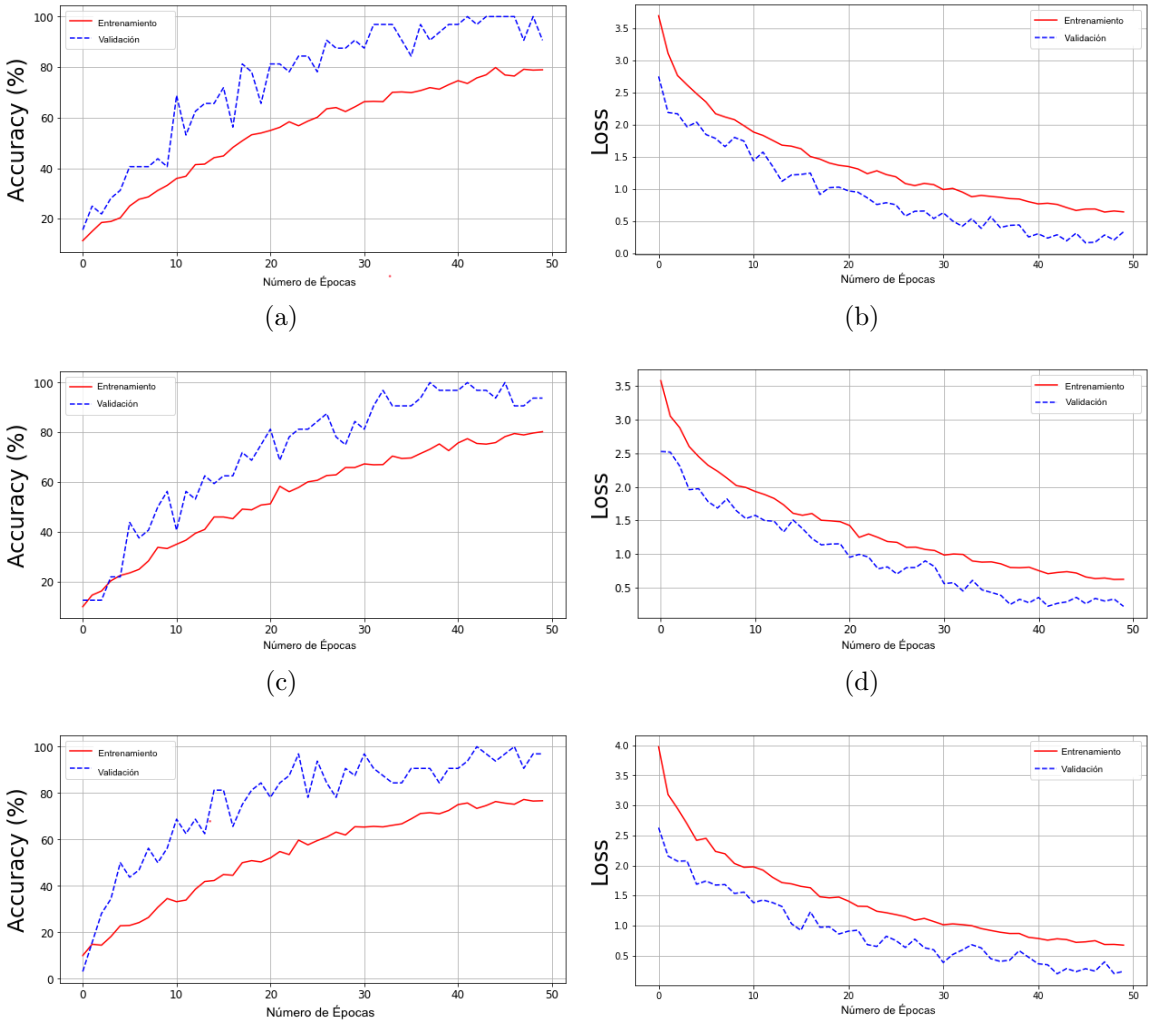


Figura 6.2: Gráficas que muestran la evolución de la *Accuracy* y la pérdida (*Loss*) durante el entrenamiento y la validación de cada etapa de la arquitectura propuesta. En la Figura 6.2a y la Figura 6.2b, se presentan los resultados obtenidos en la fase de entrenamiento del Encoder. La Figura 6.2c y la Figura 6.2d ilustran el desempeño del GAN-Decoder, mientras que la Figura 6.2e y la Figura 6.2f muestran los resultados finales al entrenar la arquitectura completa, evidenciando la convergencia del modelo y la mejora progresiva en su desempeño.

Capítulo 7

Resultados

Con el objetivo de evaluar la efectividad de nuestra propuesta, se llevó a cabo un análisis exhaustivo tanto cualitativo como cuantitativo de los resultados obtenidos en cada una de las etapas de la metodología, abarcando tanto la generación del ortomosaico como la reconstrucción tridimensional. Este enfoque permitió una valoración integral y detallada de los procedimientos implementados en ambas fases del estudio.

Para las pruebas relacionadas con la generación del ortomosaico, se utilizaron imágenes aéreas capturadas en diversas áreas de la universidad. Las imágenes fueron obtenidas con una resolución de $4K$, a una altitud constante de 100 metros, con un porcentaje de superposición del 50 % tanto en dirección longitudinal como transversal. Esta configuración permitió una alta exactitud en la visualización de los detalles en las áreas de interés. El área total cubierta por las imágenes fue aproximadamente de $22,500m^2$.

En lo que respecta a la generación del modelo tridimensional, se emplearon también imágenes con resolución $4K$, pero con configuraciones variables de altitud, que abarcaron desde los 100 metros hasta los 500 metros. Los porcentajes de superposición entre las imágenes de las secuencias fueron del 30 %, 50 % y 80 %, tanto en el eje longitudinal como transversal. Para este conjunto de pruebas, se utilizaron dos trayectorias de vuelo específicas: *GridMission* y *CircularMission*. En consecuencia, en el presente capítulo se detallan los procedimientos seguidos en cada una de las pruebas realizadas, así como los resultados obtenidos en función de las diferentes configuraciones experimentales.

7.1. Generación del Ortomosaico

La evaluación de la eficiencia de la configuración de la red neuronal propuesta para la generación del ortomosaico se presenta de manera cualitativa a través de la obtención de un ortomosaico de baja resolución (Figura 5.4a), utilizando mapas de características y su correlación correspondiente (Figura 5.3). Asimismo, se incluyó la corrección geométrica de la correlación de datos, llevada a cabo mediante el algoritmo GLC, como se muestra en la Figura 5.4b.

Adicionalmente, los resultados obtenidos en la generación del ortomosaico de alta resolución fueron verificados mediante la configuración de una Red Generativa Adversaria (GAN), denominada GAN-Decoder. Esta red fue entrenada mediante técnicas de *fine-tuning* y *transfer learning*, utilizando como base de datos la colección de imágenes previamente generada en el contexto de la universidad.

Finalmente, se realizó una evaluación cuantitativa de los resultados obtenidos mediante la comparación de la similitud entre el ortomosaico generado y dos ortomosaicos de referencia. El primer ortomosaico de referencia fue una reconstrucción manual llevada a cabo por un técnico especializado en ortofotografía, quien empleó métodos tradicionales de procesamiento y alineación de imágenes. El segundo ortomosaico fue generado utilizando el software comercial ***Pix4DMapper***, una herramienta ampliamente reconocida en el ámbito de la fotogrametría para la creación de ortomosaicos de alta exactitud. La comparación entre estos ortomosaicos permitió evaluar la exactitud y la calidad del ortomosaico generado, considerando tanto la exactitud geométrica como la fidelidad visual en las áreas de interés.

7.1.1. Resultados Cualitativos

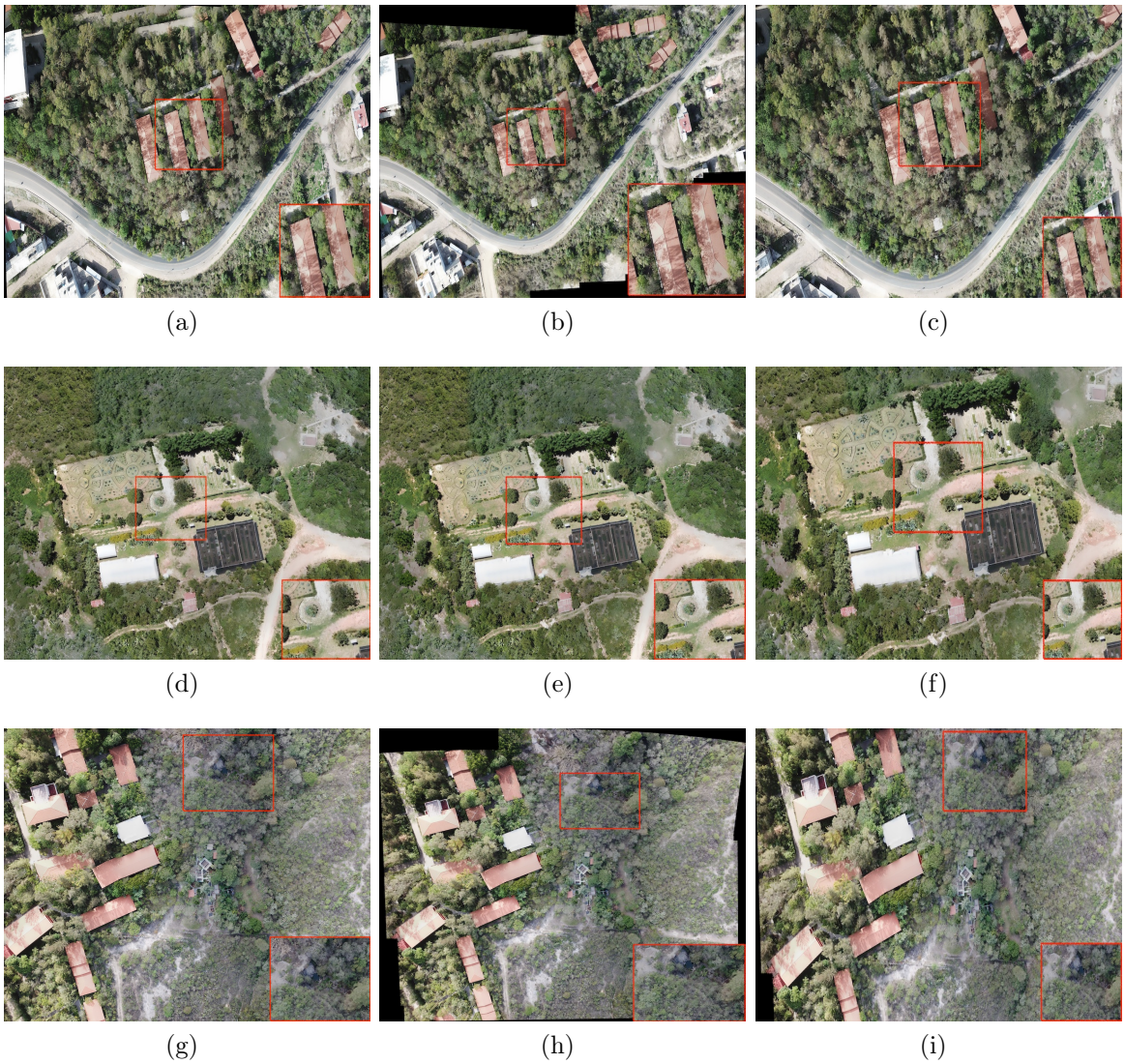
A través de la metodología propuesta, se obtuvieron ortomosaicos de calidad aceptable tanto en baja resolución (Figura 5.4) como en alta resolución (Figura 5.7). A pesar de que cada etapa del proceso ha sido correctamente configurada y ambas etapas pueden trabajar en conjunto para la generación de ortomosaicos, se observó que la primera etapa presenta ciertas limitaciones, particularmente debido a que solo maneja menos de 100 imágenes aéreas. Esta restricción impacta directamente en el proceso de emparejamiento de las imágenes, lo que requiere una simplificación adicional para mejorar la eficiencia.

Para abordar esta limitación, se implementó un mecanismo de retroalimentación de lazo cerrado o *closed-loop feedback*, que conecta la salida de la segunda etapa con la entrada de la primera (ver Figura 5.5). Este enfoque permite la integración de las imágenes del conjunto de la base de datos previamente generada con las imágenes de salida de la segunda etapa (ortomosaico de alta resolución). Gracias a este proceso, se logra expandir la capacidad de la red, permitiendo que sea capaz de manejar más de 100 imágenes aéreas en alta definición (HD). Además, esta estrategia contribuye significativamente a la mejora de los tiempos de procesamiento, optimizando tanto la exactitud como la eficiencia del sistema.

Los resultados obtenidos se consideran suficientemente válidos, dado que muestran una gran cantidad de detalles fácilmente distinguibles en el ortomosaico generado, lo cual indica una alta exactitud en el proceso de creación. Además, para corroborar la fiabilidad de los resultados, se realizó una validación mediante la comparación con dos ortomosaicos de referencia: el primero, una reconstrucción manual efectuada por un técnico especializado en ortofotografía, y el segundo, un ortomosaico generado utilizando el software comercial *Pix4DMapper*. La comparación entre los ortomosaicos generados por nuestra metodología y los de referencia permitió verificar la exactitud y la calidad del ortomosaico generado, asegurando que los resultados cumplen con los estándares de exactitud requeridos.

Las reconstrucciones manuales se llevaron a cabo utilizando imágenes de alta resolución; sin embargo, los resultados obtenidos a partir de estas reconstrucciones muestran una calidad inferior en comparación con los ortomosaicos generados mediante nuestra propuesta. Esta diferencia de calidad es evidente, especialmente al observar la capacidad de nuestra metodología para capturar detalles más precisos y fieles a la realidad.

Por otro lado, los ortomosaicos generados con el software comercial *Pix4DMapper* (última columna de la Figura 7.1) también son de alta resolución. No obstante, se observó que *Pix4DMapper* solo logró generar un ortomosaico que cubre el 80 % de la información del área total seleccionada, lo que limita la exactitud y el alcance de los resultados obtenidos. Además, las imágenes utilizadas por este software requieren de características especiales, como un adecuado solapamiento y una correcta disposición de los puntos de control, para garantizar su correcto funcionamiento y una reconstrucción precisa.



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Figura 7.1: Comparación de Resultados: En la primera columna se presentan los ortomosaicos obtenidos mediante una reconstrucción manual realizada por un experto en ortofotografía. La segunda columna muestra el ortomosaico generado con la metodología propuesta en este trabajo. Finalmente, la última columna presenta la reconstrucción obtenida utilizando el software comercial *Pix4DMapper*

7.1.2. Resultados Cuantitativos

Para evaluar la similitud entre los tres ortomosaicos generados, se utilizó la distancia euclidiana, una métrica ampliamente empleada en el análisis de similitud entre imágenes debido a su simplicidad y efectividad. La distancia euclidiana, definida por la ecuación 7.1, indica que a menor valor de la distancia, mayor es la similitud entre las imágenes comparadas (Chen and Chu, 2005; Wang et al., 2005). Esta métrica es particularmente útil en la evaluación cuantitativa de la exactitud en la reconstrucción de ortomosaicos, ya que permite una comparación directa y objetiva de las diferencias entre los resultados obtenidos.

$$d_E^2(x,y) = \sum_{k=1}^{MN}(x^k - y^k)^2 \tag{7.1}$$

Por otro lado, el error cuadrático medio (MSE, por sus siglas en inglés: Mean Squared Error) y la Proporción Máxima de Señal a Ruido (PSNR, por sus siglas en inglés: *Peak Signal-to-Noise Ratio*) son métricas de evaluación comúnmente empleadas para comparar imágenes generadas de alta resolución con imágenes reales. Estos métodos de evaluación de calidad de imagen se basan en comparaciones cuantitativas que utilizan criterios numéricos explícitos, los cuales se expresan en términos de parámetros estadísticos y pruebas específicas (Hore and Ziou, 2010). Ambas métricas son fundamentales para determinar la fidelidad de la imagen generada en relación con la imagen original, permitiendo una evaluación objetiva de la exactitud de las reconstrucciones.

La Proporción Máxima de Señal a Ruido (PSNR, por sus siglas en inglés: Peak Signal-to-Noise Ratio) es una de las métricas más comunes para evaluar la calidad de las imágenes. El PSNR, medido en decibelios (dB), se utiliza para cuantificar la relación entre la señal máxima posible de una imagen y la cantidad de ruido presente en la misma. Esta métrica se define entre dos imágenes $g(x,y)$ y $h(x,y)$ de la siguiente manera:

$$PSNR = 10log_{10}\frac{S^2}{MSE} \tag{7.2}$$

Donde S es el valor máximo de píxel en la imagen, y el error cuadrático medio (MSEor sus siglas en inglés: *Mean Squared Error*) se define como:

$$MSE = \frac{1}{MN}\sum_{x=1}^{M}\sum_{y=1}^{N}[g(x,y) - h(x,y)]^2 \tag{7.3}$$

En esta ecuación, $g(x,y)$ y $h(x,y)$ son los valores de píxel de las imágenes comparadas en las coordenadas (x,y), y M y N representan las dimensiones de la imagen (en términos de filas y columnas, respectivamente). El MSE mide el promedio de los cuadrados de las diferencias entre los píxeles de las dos imágenes, siendo una métrica que cuantifica el error global entre las imágenes comparadas.

Sin embargo, la capacidad de la distancia euclidiana (*MSE*, ecuación 7.3) y la Proporción Máxima de Señal a Ruido (*PSNR*, ecuación 7.2) para distinguir las diferencias perceptualmente relevantes, como los detalles de alta textura, es bastante limitada, ya que ambas métricas se definen en términos de las diferencias de imagen por píxel Gupta et al. (2011); Wang et al. (2004); Yang et al. (2014). Aunque un valor alto de *PSNR* indica una mayor similitud numérica entre las imágenes, no necesariamente refleja un resultado perceptualmente mejor en términos de la calidad visual de una reconstrucción en alta resolución. De hecho, una imagen con un PSNR elevado puede no ser fotorrealista o no capturar adecuadamente los detalles finos, como las texturas de alta resolución. Es importante destacar que el objetivo de aplicar estas métricas es evaluar los resultados obtenidos por un algoritmo diseñado para generar imágenes de superresolución, o en este caso, la arquitectura de redes neuronales utilizada para generar texturas de alta calidad en imágenes de alta resolución (HR).

Los resultados de la evaluación de los ortomosaicos generados se presentan en la Tabla 7.1, donde se observa que un valor alto de PSNR se correlaciona directamente con una baja distancia euclidiana. Este comportamiento nos permite concluir que los ortomosaicos generados presentan texturas de alta calidad a nivel de píxel, muy similares a las de las imágenes originales. Además, los resultados obtenidos demuestran que la metodología propuesta supera al software comercial en varios aspectos clave, lo que respalda su efectividad. Para asegurar la validez de los resultados, estos fueron contrastados con una reconstrucción realizada por un experto en fotogrametría aérea, lo que refuerza la fiabilidad y exactitud de la propuesta.

Tabla 7.1: Comparación de ortomosaicos. Esta tabla muestra las distancias euclidianas, la Proporción Máxima de Señal a Ruido (PSNR) y el tiempo de procesamiento entre el ortomosaico generado con el método propuesto, una reconstrucción manual elaborado por un experto y un ortomosaico generado con el software *Pix4DMapper*.

Ortomosaico	**Método Propuesto**	**Manual**	**Pix4DMapper**
Distancia Euclidiana	-	7.29	20.14
PSNR	-	28.17 dB	24.92 dB
Tiempo de Procesamiento	60 min	1500 min	120 min
Resolución	HR	HR	HR

7.2. Generación del modelo tridimensional

La propuesta fue evaluada utilizando tanto medidas cuantitativas como cualitativas, lo que permitió demostrar la efectividad del modelo en la reconstrucción tridimensional (3D) de paisajes urbanos, con un enfoque particular en el uso exclusivo de imágenes aéreas. Esta metodología presenta una ventaja significativa al reducir la dependencia de múltiples fuentes de información, algo que caracteriza a la mayoría de los enfoques existentes. En la literatura actual, se pueden encontrar diversas arquitecturas de Deep Learning aplicadas a la reconstrucción 3D. Sin embargo, estos modelos generalmente no están orientados a resolver el problema de la reconstrucción tridimensional utilizando únicamente imágenes aéreas, que son más fácilmente accesibles y de bajo costo en comparación con otras fuentes de datos.

La mayoría de los trabajos y modelos de Deep Learning en la literatura están centrados en el uso de imágenes estéreo, nubes de puntos generadas por sensores LiDAR o incluso en la combinación de estos datos para generar modelos 3D más precisos (Kurenkov et al., 2018; Lu et al., 2019a,1; Mandikal et al., 2018; Wang et al., 2018). Si bien estos enfoques han mostrado resultados satisfactorios en la generación de modelos 3D, su principal limitación radica en la necesidad de integrar múltiples fuentes de información, lo que no solo aumenta la complejidad del proceso, sino que también incrementa los costos y la dificultad técnica. En contraste, la metodología propuesta en este trabajo permite realizar la reconstrucción tridimensional de paisajes urbanos a partir de imágenes aéreas de alta resolución, lo que simplifica considerablemente el proceso y ofrece una solución más accesible y eficiente.

En nuestro caso, no contamos con fuentes de información adicionales más allá de las imágenes aéreas 2D obtenidas por un drón. Esto presentó un desafío, ya que muchos enfoques tradicionales para la reconstrucción tridimensional dependen de datos adicionales, como nubes de puntos generadas por sensores LiDAR o imágenes estéreo. Como resultado, fue necesario desarrollar una solución que permitiera generar nubes de puntos 3D a partir exclusivamente de este tipo de información bidimensional. A pesar de que existen enfoques en la literatura que abordan problemas similares, como el uso de imágenes 2D para generar puntos 3D (Afifi et al., 2020; Zhang et al., 2019a), estos modelos a menudo requieren configuraciones específicas o la incorporación de técnicas adicionales que van más allá de los métodos tradicionales de fotogrametría aérea. La metodología propuesta busca resolver este desafío de manera eficiente, sin recurrir a fuentes de datos adicionales, ofreciendo así una solución práctica para la reconstrucción 3D basada únicamente en imágenes aéreas de alta resolución.

Sin embargo, los modelos mencionados en la literatura están principalmente enfocados en la reconstrucción tridimensional de objetos específicos, lo cual los hace inaplicables para nuestro enfoque, que se orienta a la reconstrucción de entornos exteriores en paisajes urbanos, que incluyen vegetación y edificaciones. Este es un aspecto crucial, ya que las arquitecturas diseñadas para reconstruir objetos tienen limitaciones cuando se trata de modelar entornos complejos y dinámicos, como los que se encuentran en áreas urbanas, donde la variabilidad de los elementos naturales y construidos requiere una mayor capacidad para manejar detalles y texturas a gran escala.

Por lo tanto, no sería posible utilizar ni comparar directamente nuestro modelo con este tipo de arquitecturas, dado que la naturaleza de los datos y los objetivos de reconstrucción son significativamente diferentes. Nuestro enfoque busca capturar la complejidad de estos paisajes urbanos mediante imágenes aéreas, lo que presenta desafíos adicionales en términos de la exactitud y calidad de los modelos generados.

Para este tipo de problemas, normalmente se utilizan software comerciales como por ejemplo *Pix4DMapper*, *DroneDeploy*, *Agisoft*, entre otros, los cuales son ampliamente utilizados en la fotogrametría aérea y la reconstrucción 3D. Estos programas suelen emplear técnicas como *Structure from Motion* (SfM) para generar modelos tridimensionales a partir de imágenes 2D. Dado que la universidad cuenta con una licencia comercial de *Pix4DMapper*, decidimos comparar nuestros resultados con los obtenidos utilizando este software. Esto nos permite realizar múltiples pruebas comparativas entre los resultados obtenidos mediante *Deep Learning* y aquellos generados por el software comercial, brindando una base sólida para evaluar el rendimiento y la exactitud de nuestra propuesta en comparación con una solución ampliamente utilizada y validada en la industria. La comparación entre ambos enfoques nos proporciona una visión integral de las ventajas y limitaciones de nuestro modelo en contextos reales y de uso práctico.

7.2.1. Evaluación cualitativa del modelo tridimensional

Teniendo en cuenta que la mayoría de los software comerciales, como *Pix4DMapper*, *DroneDeploy* y *DJIGO4*, tienen requisitos similares para generar reconstrucciones tridimensionales válidas, decidimos realizar experimentos comparativos utilizando diferentes configuraciones de vuelo y parámetros de captura. Estos parámetros fueron seleccionados para simular diversas condiciones y evaluar el rendimiento del modelo en distintos escenarios. La primera configuración emplea una ruta de vuelo circular alrededor de la zona objetivo, conocida como *CircularMission*, con un 80 % de superposición entre las imágenes. La segunda configuración utiliza una ruta de vuelo en cuadrícula (*GridMission*) con un 50 % de superposición. Ambas configuraciones se realizaron utilizando imágenes en resolución $4K$ tomadas a alturas que varían entre los 150 metros y los 500 metros. La elección de estas configuraciones permitió explorar cómo las variaciones en la superposición y la altura afectan la calidad de las reconstrucciones generadas, proporcionando una base para evaluar el rendimiento de nuestra propuesta frente a los métodos tradicionales.

Los resultados cualitativos del primer experimento se muestran en la Figura 7.2, donde la primera columna presenta los resultados obtenidos con *Pix4DMapper* y la segunda columna muestra los resultados generados utilizando la metodología propuesta. Cada fila de la figura corresponde a las reconstrucciones de diferentes áreas objetivo, lo que permite observar las variaciones en la exactitud de las reconstrucciones dependiendo del área de interés. Las nubes de puntos obtenidas a partir de ambas configuraciones muestran resultados en general similares, con la metodología propuesta capturando los principales detalles de las estructuras y el terreno. Sin embargo, el software *Pix4DMapper* genera nubes de puntos más precisas en comparación con nuestra propuesta en este caso, especialmente al trabajar con configuraciones de superposición más altas, lo que mejora la exactitud en la localización de los puntos y la definición de las estructuras en las áreas evaluadas.

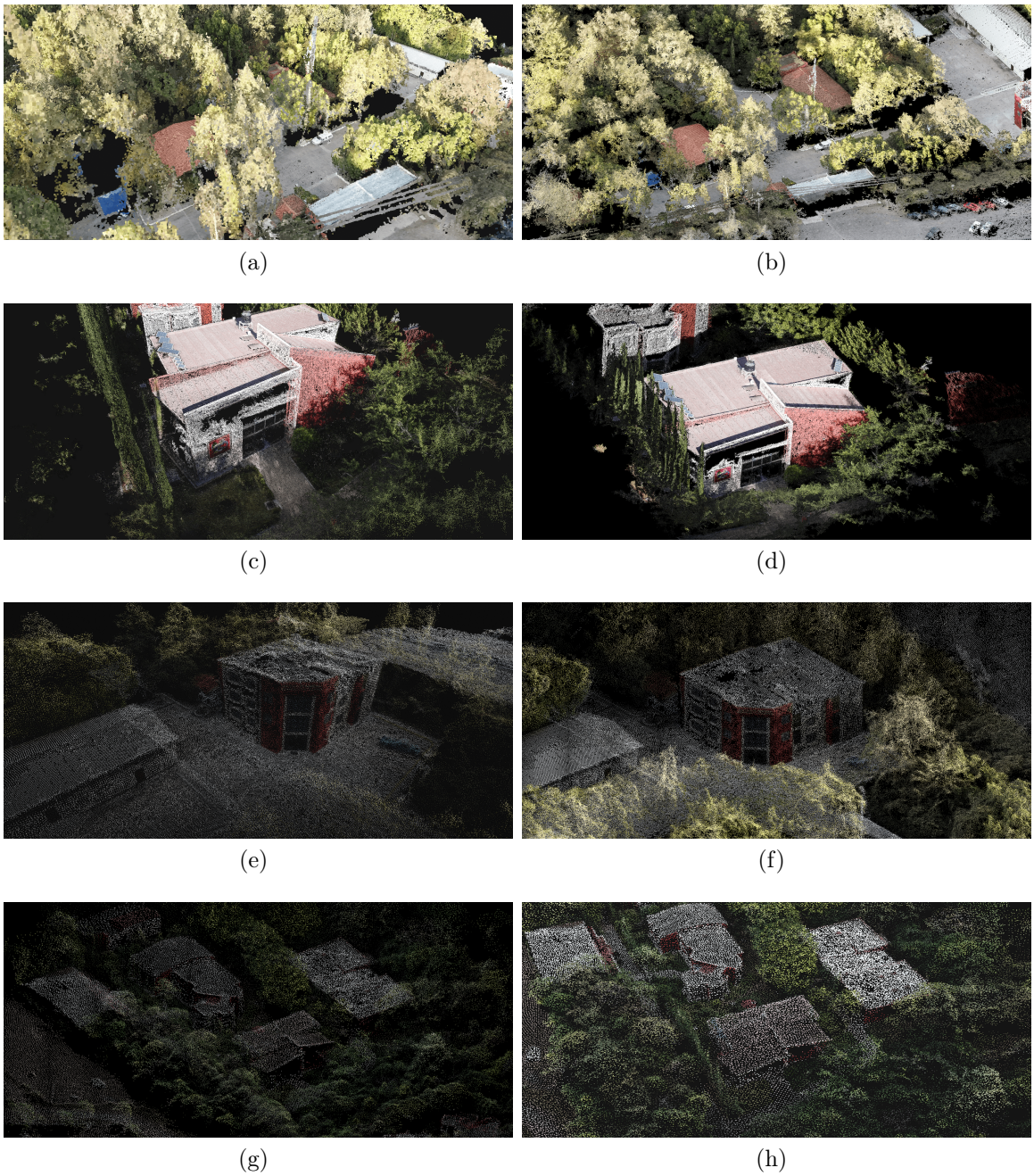
Al analizar los resultados obtenidos, es posible observar que el modelo propuesto genera una nube de puntos que no solo captura las formas y geometrías de los objetos presentes en el entorno, sino también las texturas de elementos como árboles, caminos, vehículos y edificios. La nube de puntos generada se distribuye de manera uniforme en el espacio, lo que permite una representación fiel de la topografía y estructura tridimensional del área seleccionada. Además, la distribución homogénea de los puntos asegura que tanto los detalles finos de los objetos pequeños, como las grandes estructuras (por ejemplo, edificios y caminos), sean capturados con exactitud.

La generación de la nube de puntos cubre una parte significativa del área seleccionada, garantizando que se incluya una amplia variedad de objetos presentes en el paisaje urbano y natural, lo que contribuye a una reconstrucción integral y representativa del entorno. Esta capacidad de capturar tanto la geometría como la textura de los objetos es crucial para la creación de modelos tridimensionales útiles en aplicaciones como la planificación urbana, el análisis geoespacial o el monitoreo de cambios en el paisaje. En conjunto, estos resultados indican que el modelo propuesto es eficaz en la reconstrucción de paisajes urbanos complejos, superando las limitaciones de otros enfoques que requieren múltiples fuentes de información.

7.2.2. Evaluación cuantitativa del modelo tridimensional

Los resultados obtenidos a partir de la metodología propuesta han demostrado que es posible reconocer con claridad las características geométricas y texturales de los objetos presentes en la escena. Elementos como edificios, árboles, caminos y vehículos son fácilmente distinguibles en las nubes de puntos generadas. No obstante, se ha observado que existen espacios vacíos o áreas interrumpidas que separan ciertos objetos o grupos de objetos en la reconstrucción tridimensional. Estos vacíos pueden ser relevantes dependiendo de la aplicación, ya que pueden afectar la continuidad visual o la exactitud espacial en algunas áreas específicas. Sin embargo, en general, los resultados son visualmente aceptables y muestran una representación bastante precisa del entorno urbano. Cuando se comparan con los resultados obtenidos por el software comercial *Pix4DMapper*, la reconstrucción generada por nuestra propuesta es igualmente válida y adecuada. A pesar de los espacios vacíos en algunas áreas, los resultados generales son suficientemente robustos y pueden utilizarse con confianza para aplicaciones que requieren modelos tridimensionales de paisajes urbanos, análisis geoespacial y planificación de infraestructuras.

Además, se realizó un análisis cuantitativo de la nube de puntos generada mediante el uso de dos métricas de similitud comúnmente empleadas en la evaluación de reconstrucciones tridimensionales: la distancia Chamfer (Nacken, 1994; Navaneet et al., 2019) y la distancia EMD (del inglés: *Earth Mover's Distance*) (Mandikal and Radhakrishnan, 2019; Zhang et al., 2018). La distancia Chamfer se calcula mediante la ecuación (7.4) y es una métrica que mide la discrepancia entre dos conjuntos de puntos, proporcionando una evaluación de la proximidad de las nubes de puntos generadas, sin tener en cuenta el orden de los puntos, lo que la convierte en una métrica eficiente para evaluar la similitud general entre dos nubes de puntos.



(a) (b)

(c) (d)

(e) (f)

(g) (h)

Figura 7.2: Reconstrucciones 3D. Comparación de los resultados cualitativos obtenidos con el software comercial Pix4DMapper y la arquitectura propuesta en las diferentes áreas del campus de la UTM. De izquierda a derecha: Pix4DMapper o ground-truth y la nube de puntos generada a partir del modelo propuesto.

Por otro lado, la distancia EMD se calcula a través de la ecuación (7.5) y evalúa la cantidad de "trabajo" necesario para transformar una nube de puntos en otra, teniendo en cuenta las distribuciones espaciales y las distancias geométricas entre los puntos. La EMD es especialmente útil para capturar diferencias más sutiles en la forma y la estructura de las nubes de puntos, y proporciona una medición más precisa de las similitudes y diferencias entre las reconstrucciones.

El análisis de estas métricas de similitud permite cuantificar la proximidad entre las reconstrucciones generadas por nuestra metodología y el software comercial Pix4DMapper, proporcionando un valor numérico que describe la calidad de la reconstrucción tridimensional mediante la distancia entre los puntos correspondientes. Este tipo de evaluación es crucial para validar y comparar objetivamente los resultados obtenidos, permitiendo una medición precisa de la efectividad de la metodología propuesta.

En las Ecuaciones (7.4) y (7.5), $\hat{P}$ representa la nube de puntos generada con nuestra propuesta y P representa la nube de puntos generada utilizando el software comercial *Pix4DMapper*.

$$d_{Chamfer}(\hat{P}, P) = \sum_{x \in \hat{P}} min_{y \in P} \|x - y\|_2^2 + \sum_{y \in \hat{P}} min_{x \in P} \|x - y\|_2^2 \tag{7.4}$$

$$d_{EMD}(\hat{P}, P) = min_{\phi : \hat{P} \to P} \sum_{x \in \hat{P}} \|x - \phi(x)\|_2 \tag{7.5}$$

Para obtener la distancia $d_{Chamfer}$, se calcula la suma de las distancias al cuadrado entre cada punto de la nube de puntos y su vecino más cercano. Esta métrica, $d_{Chamfer}$, es suave y continua en muchas partes, y el proceso de búsqueda es independiente para cada punto. Cuanto menor sea el valor de $d_{Chamfer}$, mayor será la exactitud y similitud entre las dos nubes de puntos.

En el caso de la distancia EMD (del inglés: *Earth Mover's Distance*), se utiliza una biyección $\phi : \hat{P} \to P$, donde cada punto en $\hat{P}$ se asigna a un único punto en P. Este proceso impone una asignación punto a punto entre las dos nubes de puntos, lo que permite una comparación más precisa.

En la Tabla 7.2 se muestran los resultados de la distancia entre la nube de puntos generada con la metodología propuesta y la generada utilizando el software comercial. Los resultados con un bajo valor de $d_{Chamfer}$ indican una estrecha similitud entre las nubes de puntos comparadas.

Los resultados de la reconstrucción tridimensional obtenidos con *Pix4DMapper* y los obtenidos mediante nuestra propuesta se presentan en la Tabla 7.2. Para evaluar la similitud entre las nubes de puntos generadas por ambos métodos, se utilizaron dos métricas de distancia: $d_{Chamfer}$ y d_{EMD}. Estas métricas permiten comparar las distancias entre las nubes de puntos generadas por el software comercial y nuestra metodología.

En los experimentos, se emplearon un total de 1024 puntos de muestra, con diferentes configuraciones de imágenes aéreas: 1400, 700 y 300 imágenes, con porcentajes de superposición del 80 %, 50 % y 30 %, respectivamente. Las imágenes fueron capturadas a una altura de 150 metros. Estos parámetros permitieron realizar una evaluación robusta de la similitud entre las nubes de puntos generadas en ambos casos.

Los resultados obtenidos indican que la similitud entre las nubes de puntos generadas por *Pix4DMapper* y nuestra propuesta es muy cercana cuando se utilizan porcentajes de superposición del 80 %, que es el valor mínimo requerido por el software comercial para garantizar reconstrucciones tridimensionales precisas. Sin embargo, al reducir el porcentaje de superposición, particularmente en los casos de 50 % y 30 %, el software comercial no logra generar una reconstrucción tridimensional válida. En contraste, nuestra metodología sigue siendo capaz de generar una nube de puntos coherente, incluso con estos porcentajes de superposición más bajos.

Como resultado, las métricas de similitud, como las distancias $d_{Chamfer}$ yd_{EMD}, aumentan a medida que el porcentaje de superposición disminuye, lo que refleja la mayor flexibilidad y capacidad de la propuesta frente a los limitantes del software comercial en estas configuraciones. Esta tendencia se puede observar claramente en la Figura 7.3, donde se muestran las variaciones en las métricas de similitud en función del porcentaje de superposición y las respectivas distancias.

Además, se observa una mejora significativa en el tiempo de procesamiento utilizando la metodología propuesta ($t_{proposal}$) en comparación con el software comercial ($t_{Pix4DMapper}$). La propuesta permite generar una nube de puntos con un tiempo de cálculo considerablemente menor, lo que no solo optimiza el flujo de trabajo, sino que también hace que el proceso sea más eficiente y accesible. A pesar de la reducción en el tiempo de procesamiento, la calidad de la reconstrucción generada es excepcional, manteniendo la alta resolución en texturas, mallas y estructuras volumétricas presentes en los diversos objetos de los paisajes urbanos. Los resultados obtenidos con la metodología propuesta son muy similares a los producidos por *Pix4DMapper*, pero con la ventaja añadida de una mayor eficiencia en los tiempos de computación.

En la segunda configuración, se emplearon las mismas áreas objetivo que en la configuración anterior. En este caso, se utilizó la ruta de vuelo *Grid Mission*, y los resultados obtenidos se muestran en la Figura 7.3. La primera columna de la figura muestra los resultados generados con *Pix4DMapper*, mientras que la segunda columna presenta los resultados obtenidos mediante la metodología propuesta. La comparación entre ambos métodos permite evaluar la exactitud y la calidad de las nubes de puntos generadas, observando la capacidad de ambos enfoques para representar con exactitud las características y estructuras del paisaje urbano.

A diferencia de la configuración anterior, los resultados obtenidos en este experimento muestran una clara mejora en la reconstrucción 3D utilizando nuestra metodología propuesta. Mientras que *Pix4DMapper* y otros software comerciales requieren configuraciones especiales, tales como la ruta de vuelo específica y ajustes detallados de la cámara, nuestra propuesta demuestra ser más flexible, permitiendo obtener resultados de alta calidad sin la necesidad de estos requisitos estrictos. Este aspecto proporciona una ventaja significativa al simplificar el proceso de captura y generación de reconstrucciones tridimensionales de paisajes urbanos.

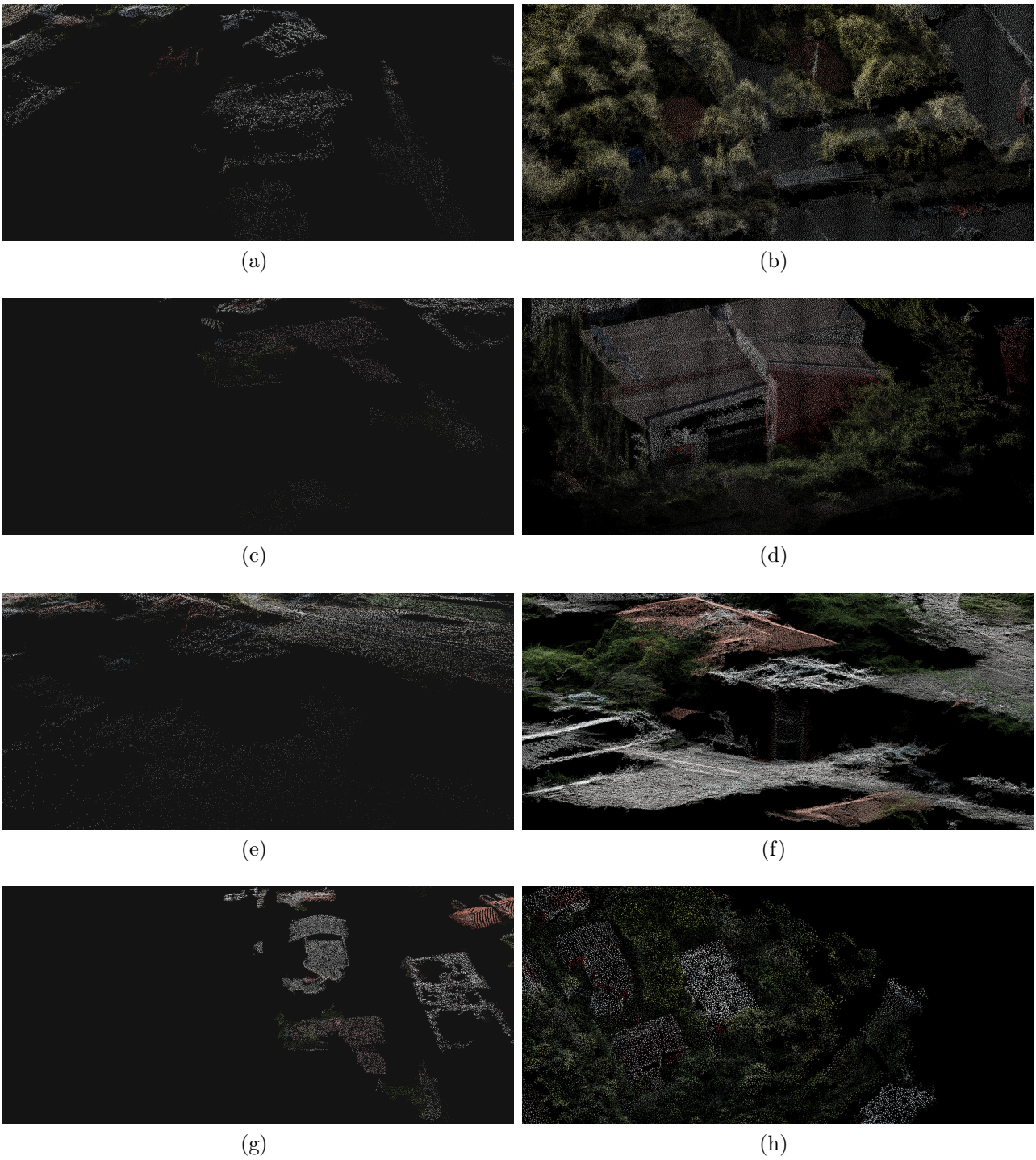
Si las configuraciones se modifican ligeramente, como se muestra en nuestras pruebas, los resultados obtenidos con *Pix4DMapper* y otros software comerciales se vuelven desfavorables, ya que dependen en gran medida de parámetros como la ruta de vuelo, el tipo de cámara utilizada y el ángulo de captura. En comparación, la metodología presentada en este documento es robusta ante este tipo de variaciones. Nuestra propuesta no depende de configuraciones especiales ni de factores externos para generar nubes de puntos claras y legibles. Esto hace que sea una solución más flexible y accesible para la reconstrucción 3D de paisajes urbanos, sin comprometer la calidad de los resultados.

Es importante destacar que la propuesta presentada es capaz de reconstruir áreas que no estaban incluidas en el conjunto de datos de entrenamiento, como por ejemplo, regiones fuera del campus universitario. La Figura 7.4 ilustra este desempeño, mostrando las reconstrucciones tridimensionales obtenidas a partir de nubes de puntos. En la primera columna se presentan los resultados generados por Pix4DMapper, mientras que en la segunda columna se observan los resultados obtenidos mediante la metodología propuesta. Este hallazgo pone de manifiesto la capacidad de nuestra metodología para generalizarse y adaptarse a entornos no previamente modelados, lo que evidencia la robustez y flexibilidad del enfoque frente a escenarios desconocidos.

Por otro lado, se realizaron pruebas utilizando imágenes tomadas a diferentes alturas, que variaron entre los 300 metros y los 500 metros. Los resultados obtenidos demuestran que, para lograr reconstrucciones tridimensionales precisas a alturas superiores a los 300 metros, es necesario ajustar y mejorar la arquitectura del modelo. En particular, la Figura 7.5 presenta los resultados de las reconstrucciones de un área objetivo a dos alturas distintas: 150 metros (Figura 7.5a) y 400 metros (Figura 7.5d). En el caso de las imágenes capturadas a 150 metros, el modelo es capaz de generar una reconstrucción detallada y precisa, con una cantidad adecuada de puntos para una representación tridimensional confiable del área. Sin embargo, cuando las imágenes se capturan a 400 metros, la cantidad de puntos generados por el modelo es insuficiente, lo que impide obtener una reconstrucción 3D precisa de la zona objetivo. Este comportamiento sugiere que, a alturas superiores a los 300 metros, el modelo enfrenta dificultades para generar nubes de puntos densas y detalladas, lo que afecta negativamente la calidad de la reconstrucción tridimensional. Estos resultados resaltan la necesidad de revisar y ajustar tanto la estrategia de captación de imágenes como los parámetros de la arquitectura propuesta, para garantizar una mayor efectividad y exactitud en alturas elevadas, permitiendo obtener una cobertura más amplia y una representación tridimensional más detallada del entorno.

Tabla 7.2: Comparación entre los resultados de reconstrucción obtenidos con Pix4DMapper y los resultados obtenidos con nuestra propuesta. Las métricas se calculan sobre 1024 puntos. Además, los resultados se calculan utilizando 1400, 700 y 300 imágenes aéreas con un porcentaje de superposición del 80 %, 50 %, and 30 % respectivamente. Las distancias sin valor indican un desbordamiento de los datos.

Área seleccionada	$d_{Chamfer}$	d_{EMD}	$t_{propuesta}$	$t_{Pix4DMapper}$
Superposición = 80 %				
Plaza principal	11.99	18.51	50 min	90 min
Edificios de laboratorrios	18.56	20.15	98 min	180 min
Edificios de Institutos	21.56	32.15	120 min	180 min
Salones de clases	18.79	20.23	80 min	120 min
Superposición = 50 %				
Plaza principal	41.99	48.51	30 min	50 min
Edificios de laboratorrios	58.56	50.15	60 min	80 min
Edificios de Institutos	41.56	62.15	80 min	70 min
Salones de clases	58.79	40.23	50 min	70 min
Superposición = 30 %				
Plaza principal	51.99	45.51	30 min	60 min
Edificios de laboratorrios	–	–	30 min	60 min
Edificios de Institutos	61.56	62.15	40 min	60 min
Salones de clases	–	–	30 min	60 min



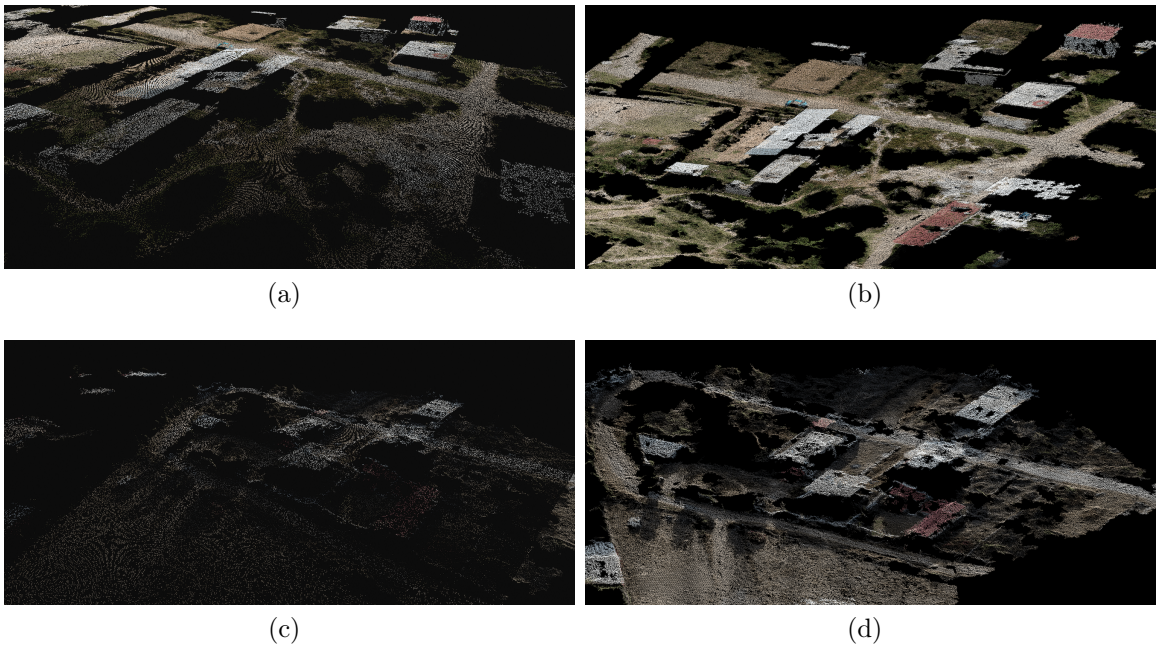
(a) (b)

(c) (d)

(e) (f)

(g) (h)

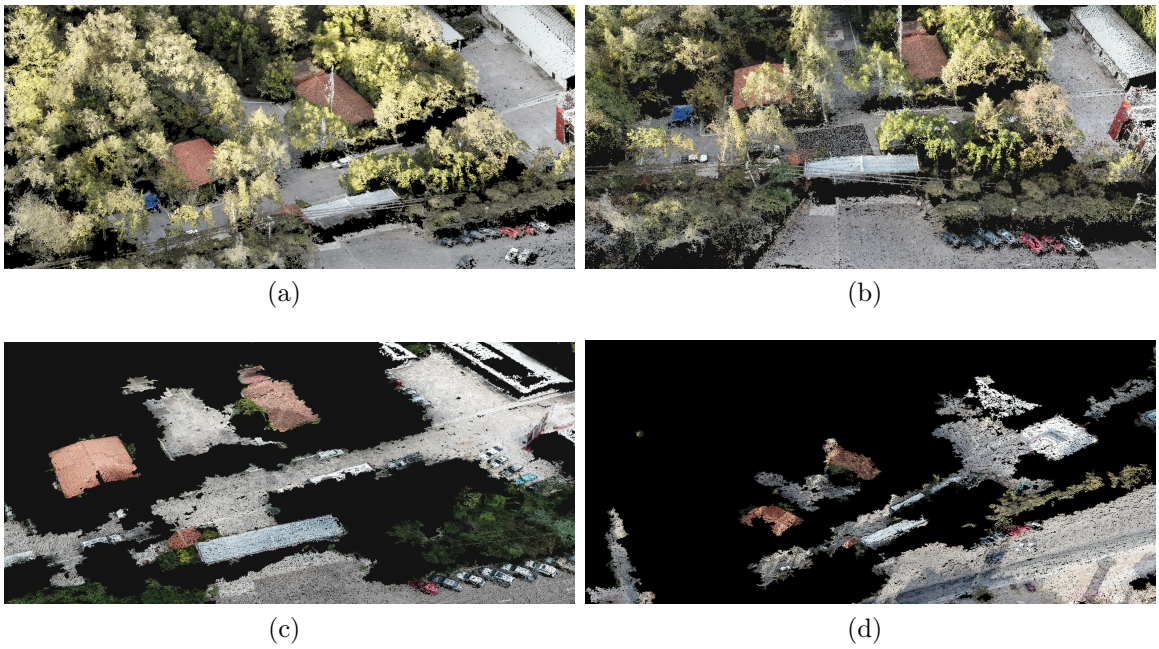
Figura 7.3: Comparación de los resultados de reconstrucciones tridimensionales de diferentes áreas de la universidad. Los resultados se obtienen tras procesar imágenes con un 50 % de superposición, tomadas en una ruta Grid Mission. La primera columna muestra los resultados obtenidos con el software comercial Pix4DMapper y la segunda columna muestra los resultados obtenidos con nuestra arquitectura propuesta.



(a) (b)

(c) (d)

Figura 7.4: Reconstrucciones tridimensionales de áreas fuera del campus de la UTM. Los resultados obtenidos con nuestra metodología (Figura 7.4b,7.4d) se comparan con software comercial (Figura 7.4a,7.4c).



(a) (b)

(c) (d)

Figura 7.5: Reconstrucciones tridimensionales utilizando imágenes aéreas tomadas a una altura de 150 metros (Figura 7.5a), 200 metros (Figura 7.5b), 300 metros (Figura 7.5c) y con 400 metros (Figura 7.5d). Estos resultados muestran que nuestra metodología realiza reconstrucciones 3D a diferentes alturas. Sin embargo, para altitudes mayores o iguales a 300 metros presenta algunas dificultades para realizar reconstrucciones tridimensionales.

7.3. Análisis de resultados

Los resultados obtenidos mediante la metodología propuesta evidencian un rendimiento competitivo, e incluso superior, en comparación con reconstrucciones manuales y herramientas comerciales ampliamente utilizadas como Pix4DMapper, así como frente a técnicas clásicas de reconstrucción 2D basadas en visión por computadora.

A diferencia de algoritmos no supervisados tradicionales como *SIFT*, que dependen de la extracción manual de características locales mediante técnicas supervisadas, nuestro enfoque basado en redes neuronales profundas ofrece ventajas sustanciales, especialmente en la detección y extracción automática de keypoints. Las redes profundas son capaces de aprender representaciones latentes más robustas y discriminativas directamente de los datos, lo cual mejora significativamente la exactitud y continuidad de la reconstrucción tridimensional, incluso en entornos con poca textura o variabilidad geométrica.

7.3.1. Generalización y robustez

Las reconstrucciones realizadas en zonas no incluidas en el conjunto de entrenamiento muestran resultados coherentes y sin deformaciones evidentes, lo que evidencia la capacidad de generalización del modelo propuesto. Esta propiedad se atribuye, en gran medida, al uso combinado de funciones de pérdida geométrica y métricas de error que permiten optimizar directamente la estructura espacial en 2D y 3D, incluso en contextos no vistos durante el entrenamiento.

La comparación directa con Pix4DMapper —software ampliamente reconocido en aplicaciones de fotogrametría aérea— demuestra que nuestra metodología no solo alcanza una calidad comparable o superior en la generación de nubes de puntos, sino que también ofrece importantes ventajas en términos de eficiencia computacional y robustez. En particular, se observa una reducción significativa en los tiempos de procesamiento y una mayor tolerancia frente a variaciones en los parámetros de adquisición, como el tipo de cámara, la altura de vuelo y el porcentaje de superposición entre imágenes.

Mientras Pix4DMapper requiere configuraciones específicas para lograr reconstrucciones óptimas (por ejemplo, una superposición frontal del 75–85 %, lateral del 60–85 %, y una calibración precisa de la cámara ajustada al terreno) Pix4D (2019), el enfoque propuesto demuestra una mayor flexibilidad operativa. De acuerdo con los resultados presentados en las Tablas 7.1 y 7.2, el modelo mantiene reconstrucciones consistentes incluso con superposiciones reducidas (∼50 %), condición en la que Pix4DMapper tiende a presentar pérdidas significativas de puntos y errores de alineación.

En cuanto al desempeño computacional, los tiempos de procesamiento se reducen hasta en un 50 % en las distintas regiones evaluadas. Esto se debe a la implementación de una arquitectura basada en redes generativas adversarias (GANs) optimizadas, así como a técnicas eficientes de inferencia que disminuyen la carga computacional sin comprometer la calidad del resultado.

Por último, los experimentos realizados con imágenes capturadas entre 150 m y 400 m de altura evidencian que el modelo mantiene un rendimiento sólido hasta aproximadamente los 300 m. A partir de esa altitud, la calidad de reconstrucción comienza a decrecer debido a la pérdida de resolución espacial, lo que limita la densidad y exactitud de los puntos generados.

7.3.2. Entrenamiento híbrido y GANs

La estrategia de entrenamiento híbrido adoptada en este trabajo combina enfoques supervisados y auto-supervisados, lo que permite al modelo aprender tanto de anotaciones explícitas como de la estructura inherente de los datos no etiquetados. Esta combinación resulta particularmente efectiva para mejorar la capacidad de generalización del sistema, permitiéndole adaptarse a escenarios no vistos durante el entrenamiento, así como a variaciones en condiciones de captura, tales como iluminación, ángulo o escala.

Además, la incorporación de arquitecturas basadas en Redes Generativas Adversarias (GANs) ha tenido un impacto significativo en la calidad de las reconstrucciones generadas. Las GANs, al incluir una red generadora y una red discriminadora en un esquema de competencia, permiten optimizar representaciones más realistas y coherentes, especialmente en regiones con baja textura o información visual limitada. Esta capacidad de las GANs para inferir estructuras plausibles en áreas ambiguas representa una ventaja importante respecto a métodos tradicionales de reconstrucción basados únicamente en correspondencias explícitas de píxeles o puntos clave.

A diferencia de los algoritmos clásicos de visión por computadora —que dependen de heurísticas rígidas y una buena visibilidad de las características locales—, la arquitectura propuesta logra inferencias robustas gracias al aprendizaje profundo de representaciones espaciales y contextuales. Este enfoque no solo supera las limitaciones de herramientas comerciales como Pix4DMapper, sino que también establece una solución escalable, flexible y de alta exactitud para tareas de reconstrucción 3D en entornos reales y no controlados.

Capítulo 8

Conclusiones y Trabajos Futuros

En este capítulo se presentan algunas conclusiones generales y específicas generadas a partir de los resultados obtenidos durante el desarrollo de la solución para el planeamiento del problema al inicio de esta investigación. De igual forma, también se presentan algunas propuestas como trabajos futuros para dar continuidad e incrementar el alcance de este trabajo en un futuro.

8.1. Conclusiones

El presente trabajo ha demostrado la viabilidad y eficacia de emplear arquitecturas de *Deep Learning* para la reconstrucción bidimensional y tridimensional de entornos exteriores a partir de imágenes aéreas capturadas con vehículos aéreos no tripulados (*UAVs*). La metodología propuesta, basada en una combinación de Autoencoders y Redes Generativas Adversarias (*GANs*), ha permitido la generación de ortomosaicos y modelos tridimensionales de alta resolución, con una eficiencia de procesamiento superior a la de algunas herramientas comerciales como Pix4DMapper. A través de una serie de experimentos, se ha validado que la arquitectura diseñada es capaz de reconstruir con alta exactitud diversas zonas del campus de la Universidad Tecnológica de la Mixteca, así como áreas con características geográficas diversas.

Con este trabajo se buscaba principalmente la reconstrucción $2D$ de las áreas recorridas por un dron, utilizando una arquitectura de redes neuronales profundas que permitiera el procesamiento eficiente de la información visual. Para ello, se desarrolló una metodología simple que posibilitó la construcción de ortomosaicos exclusivamente con imágenes aéreas y la generación de modelos tridimensionales de la misma zona de interés.

Uno de los principales aportes de este trabajo es la generación de una base de datos de imágenes aéreas, compuesta por $3,000$ imágenes de alta definición (ver Tabla 4.1), organizadas sistemáticamente para facilitar el ajuste del conocimiento previo de la red neuronal CNN seleccionada. Mediante un proceso de reentrenamiento (*fine tuning*), la arquitectura profunda adquirió la capacidad de trabajar con nuestro conjunto de datos, extrayendo mapas de características y correlacionando puntos coincidentes en pares de imágenes.

En comparación con trabajos previos, como el de Li et al. (2019), que utiliza múltiples fuentes de datos y redes convolucionales para la reconstrucción urbana en 3D, nuestra metodología presenta la ventaja de operar exclusivamente con imágenes aéreas sin necesidad de datos complementarios, reduciendo la complejidad del procesamiento. Asimismo, en contraste con la propuesta de Ghamisi and Yokoya (2018), basada en GAN para la simulación de Modelos Digitales de Superficie (*DSM*), nuestro enfoque logra una mayor exactitud en la generación de nubes de puntos sin necesidad de preprocesamiento adicional de los datos de entrada. Por otro lado, si bien enfoques como el de Tang et al. (2018) han demostrado ser efectivos en la detección de puntos característicos para la reconstrucción 3D, nuestra metodología mejora la continuidad en las reconstrucciones tridimensionales y presenta una mayor robustez en entornos con variaciones de iluminación y texturas homogéneas.

Los resultados fueron evaluados utilizando la distancia euclidiana como medida de similitud y la Proporción Máxima de Señal a Ruido (*PSNR*). El ortomosaico obtenido se comparó con una reconstrucción manual y con una obtenida mediante software comercial, demostrando que nuestra metodología proporciona resultados similares a los de una reconstrucción manual, pero con detalles de alta definición. Además, se presentó una novedosa arquitectura de red neuronal profunda para la generación de nubes de puntos a partir de imágenes aéreas de paisajes urbanos y naturales. Se realizaron ajustes clave en el *Autoencoder*, combinando una red residual en la etapa *Encoder* con una red *GAN* en la etapa Decoder, denominada ***GAN-Decoder***. Utilizando esta arquitectura, se obtuvieron resultados comparables e incluso superiores a los generados con software comercial.

La metodología propuesta es robusta ante variaciones en la configuración del vuelo para la adquisición de imágenes. No depende de un trayecto específico del *UAV* y permite obtener resultados precisos con un menor porcentaje de superposición de imágenes y en tiempos de procesamiento significativamente menores a los requeridos por software comercial. Estas ventajas fueron validadas mediante la comparación de nuestros resultados con los obtenidos con Pix4DMapper, un software con licencia completa que permitió una evaluación exhaustiva de la propuesta.

Sin embargo, se identificaron algunas limitaciones. Se observó que a alturas superiores a 300 metros, la densidad de puntos en las reconstrucciones tridimensionales disminuye significativamente, afectando la exactitud de los modelos generados. Asimismo, en áreas con texturas homogéneas o con variaciones mínimas en la estructura del terreno, el algoritmo presentó dificultades en la generación de detalles precisos. A pesar de estas limitaciones, los resultados obtenidos sientan un precedente sólido para la aplicación de estas técnicas en una amplia gama de escenarios, desde la cartografía digital hasta el monitoreo ambiental y la gestión de recursos naturales.

8.2. Trabajos Futuros

La generación de ortomosaicos en resoluciones superiores, como ***Full HD*** o $4K$, se vislumbra como un área de mejora clave para ampliar las aplicaciones del modelo en contextos que requieran una mayor exactitud visual. Además, integrar los algoritmos utilizados en sistemas ***SLAM*** permitirá la creación de mapas tridimensionales dinámicos en tiempo real, fortaleciendo su integración con la navegación autónoma.

Es fundamental continuar optimizando la red discriminante dentro de la arquitectura *GAN*, lo que permitirá mejorar la generación de datos tridimensionales y aumentar la exactitud en la reconstrucción de terrenos con alta complejidad geométrica. También es necesario explorar mejoras en la arquitectura para su uso en condiciones extremas, como la reconstrucción en altitudes superiores a 300 metros o en áreas con texturas homogéneas, lo que sigue representando un reto significativo.

En comparación con trabajos recientes, como los de Zheng et al. (2022), han demostrado que la combinación de aprendizaje profundo con técnicas de optimización geométrica puede mejorar la reconstrucción tridimensional en escenarios desafiantes. Integrar estos avances en el modelo propuesto podría ofrecer mejoras sustanciales en exactitud y eficiencia. Finalmente, la incorporación de modelos generativos más avanzados, como *StyleGAN* o *NeRF*, podría ampliar las capacidades del sistema en la reconstrucción de superficies con un nivel de detalle aún mayor, brindando nuevas oportunidades para aplicaciones en cartografía, conservación ambiental y monitoreo de infraestructuras.

De igual manera, para ampliar las capacidades del sistema en la reconstrucción de superficies con un mayor nivel de detalle, Charles et al. (2022) sugiere explorar modelos generativos avanzados como *StyleGAN* o *NeRF*. Por ejemplo, el trabajo "*Style2NeRF: An Unsupervised One-Shot NeRF for Semantic 3D Reconstruction*"presenta un modelo no supervisado para la recuperación de la pose 3D, forma y apariencia de objetos simétricos.

La validación en entornos reales, caracterizados por condiciones topográficas y climáticas extremas, será crucial para confirmar la robustez del modelo y su aplicabilidad en diversas disciplinas. Entornos como bosques densos o áreas urbanas de alta densidad ofrecerán la oportunidad de evaluar el desempeño del modelo en situaciones variadas y complejas. A medida que las técnicas de optimización y aprendizaje profundo continúan evolucionando, es probable que investigaciones futuras se orienten hacia la integración de modelos híbridos. Estos modelos podrían combinar visión por computadora, sensores multiespectrales y aprendizaje reforzado, ampliando las capacidades del sistema y su adaptabilidad.

En este sentido, la exploración de arquitecturas híbridas que integren redes convolucionales con modelos de atención podría potenciar significativamente la extracción de características y mejorar la exactitud en la reconstrucción de estructuras complejas. Tal enfoque permitiría generar modelos tridimensionales aún más detallados y realistas, ampliando las posibilidades de aplicación en sectores como la agricultura de exactitud, la gestión de desastres naturales y el mapeo de infraestructuras urbanas. Así, la presente investigación establece un marco sólido para el desarrollo de soluciones más sofisticadas, que prometen impactar significativamente en la mejora de estos campos con mayor exactitud y eficiencia.

Apéndice A

Publicaciones

- ✓ Rodríguez-Santiago, A. L., Arias-Aguilar, J. A., Petrilli-Barceló, A. E., & Miranda-Luna, R. (2020, June). A Simple Methodology for 2D Reconstruction Using a CNN Model. In Mexican Conference on Pattern Recognition (pp. 98-107). Springer, Cham.
- ✓ Rodríguez-Santiago, A. L., Arias-Aguilar, J. A., Hiroshi Takemura, & Petrilli-Barceló, A. E.. A feedback methodology for high-definition reconstructions using CNNs. Desarrollos Científicos y Tecnológicos en las Ciencias Computacionales. November 2020. Benemérita Universidad Autónoma de Puebla. ISBN: 978-607-8728-36-7, p. 50-57.
- ✓ Rodríguez-Santiago, A. L., Arias-Aguilar, J. A., Takemura, H., & Petrilli-Barcelo, A. E. (2021). High-Resolution Reconstructions of Aerial Images Based On Deep Learning. Computación y Sistemas, 25(4).
- ✓ Rodríguez-Santiago, A. L., Arias-Aguilar, J. A., Takemura, H., & Petrilli-Barceló, A. E. (2021). A Deep Learning Architecture For 3D Mapping Urban Landscapes. Applied Sciences, 11(23), 11551.

A Simple Methodology for 2D Reconstruction Using a CNN Model

Armando Levid Rodríguez-Santiago, José Anibal Arias-Aguilar, Alberto Elías Petrilli-Barceló(✉), and Rosebet Miranda-Luna

Graduate Studies Division, Universidad Tecnológica de la Mixteca, Km. 2.5 Carretera a Acatlima, 69000 Huajuapan de León, Oaxaca, Mexico
levid.rodriguez@gmail.com, {anibal,petrilli,rmiranda}@mixteco.utm.mx
http://www.utm.mx

Abstract. In recent years, Deep Learning research have demonstrated their effectiveness in digital image processing, mainly in areas with heavy computational load. Such is the case of aerial photogrammetry, where the principal objective is to generate a 2D map or a 3D model from a specific terrain. In these topics, high-efficiency in visual information processing is demanded. In this work we present a simple methodology to build an orthomosaic, our proposal is focused in replacing traditional digital imagen processing using instead a Convolutional Neuronal Network (CNN) model. The dataset of aerial images is generated from drone photographs of our university campus. The method described in this article uses a CNN model to detect matching points and RANSAC algorithm to correct feature's correlation. Experimental results show that feature maps and matching points obtained between pair of images through a CNN are comparable with those obtained in traditional artificial vision algorithms.

Keywords: Deep Learning · CNN · 2D reconstruction · Aerial images

1 Introduction

Image stitching produces a mosaic that corresponds to a set of images taken from one or several cameras which overlap and are joined in a single image [6]. In the generation of this mosaic several computer vision techniques are used. We worked with aerial images and computer vision strategies combined with photogrammetry techniques.

The stitching process is usually made with traditional computer vision methods as shown in Fig. 1a. It begins with a drone flight plan to image acquisition of a selected area. Then placeholders with georeferenced points are added over a map as well as flight height and overlapping percentage between each pair of acquired images. Usually a mobile application is configured with these specifications to acquire the information autonomously. Some popular free apps to help in this stage are Pix4D and DroneDeploy.

K. M. Figueroa Mora et al. (Eds.): MCPR 2020, LNCS 12088, pp. 98–107, 2020.
https://doi.org/10.1007/978-3-030-49076-8_10

Desarrollos Científicos y Tecnológicos en las Ciencias Computacionales

A Feedback Methodology for High-Definition Reconstructions Using CNNs

Armando Levid Rodríguez-Santiago[1,3], José Aníbal Arias-Aguilar[1,4], Hiroshi Takemura[2,5] and Alberto Elías Petrilli-Barceló[2,5]

[1] Universidad Tecnológica de la Mixteca, Km. 2.5 Carretera a Acatlima, 69000, Huajuapan de Léon, Oaxaca, México. Graduate Studies Division

[2] Tokyo University of Science, 2641 Yamazaki, Noda, Chiba 278-8510, Japan. Department of Mechanical Engineering Faculty of Science and Technology

[3] levid.rodriguez@gmail.com

[4] anibal@mixteco.utm.mx

[5] {takemura, petrilli}@rs.tus.ac.jp

Abstract. We present a methodology for the reconstruction of orthomosaics in high-definition resolution. Our proposal consists of two CNN networks connected and working with each other. Each one has been modified in their internal structure from known architectures and re-trained to have enough knowledge to calculate corresponding parameters from two images and perform image stitching to obtain an orthomosaic with high-resolution details. Results obtained show that our methodology provides similar results to those of a reconstruction performed by an expert in orthophotography but with high definition details.

Keywords: Deep Learning · CNN · 2D Reconstruction · Aerial images.

1 Introduction

In the process to generate an orthomosaic (orthophotography), techniques of aerial photogrammetry are used. Photogrammetry is a technique that determines geometric properties of the terrain and spatial relations from aerial photographic images [3].

The union of a set of images that are superimposed and joined in a single image produce an orthophotography. An orthophotography allows us to have current visual knowledge of an area of interest, as well as the visualization of a surface through a photograph where it is possible to appreciate all present elements in a land with validity similar to that of a cartographic plane.

However, it is necessary that the resolution of the generated orthophoto was as high as possible.

In the context of this work, we propose a novel methodology to generate high-resolution orthomosaics. Unlike previous works, we define a proposal that combines main elements of two deep neuronal network models joined with a closed-loop feedback that optimizes the process to generate results.

2 Related Works

Recent research in Deep Learning has included studies to obtain terrain models such as those presented in [2, 4, 8, 12] where they perform image pairing or 3D reconstructions using deep neuronal network techniques. Resulting maps or models need to have details in high-definition (HD), so it is necessary to work with aerial images with high-resolution. In

ISSN 2007-9737

High-Resolution Reconstructions of Aerial Images Based on Deep Learning

Armando Levid Rodríguez-Santiago[1], José Aníbal Arias-Aguilar[1], Hiroshi Takemura[2], Alberto Elías Petrilli-Barceló [2]

[1] Universidad Tecnológica de la Mixteca, Graduate Studies Division, México

[2] Tokyo University of Science, Faculty of Science and Technology, Department of Mechanical Engineering, Japan

levid.rodriguez@gmail.com, anibal@mixteco.utm.mx, {takemura, petrilli}@rs.tus.ac.jp

Abstract. We present a methodology for high-resolution orthomosaic reconstruction using aerial images. Our proposal consists a neural network with two main stages, one to obtain the correspondences necessary to perform a LR-orthomosaic and another one that uses these results to generate an HR- orthomosaic, and a feedback connection. The CNN are based on well known models and are trained to perform image stitching and obtain a high-resolution orthomosaic. The results obtained in this work show that our methodology provides similar results to those obtained by an expert in orthophotography, but in high-resolution.

Keywords. Deep learning, CNN, 2D reconstruction, aerial images, orthophotography, photogrammetry.

1 Introduction

To generate an orthomosaic (orthophotography), aerial-photogrammetry techniques are used. Photogrammetry is a technique that determines geometric properties and spatial relations of the terrain from aerial photographic images [3]. It is a very complex process in which the main objective is to convert two-dimensional data (flat images) into cartographic/three-dimensional data. This technique allows us to obtain the geometric properties of a surface based on information obtained from several images with redundant information. It is this repeated structure that allows the extraction of the object's structure through the overlap among consecutive images.

The pairing of a set of overlapping images that are joined in a single image produces an orthophotography. Orthophotography allows us to have current visual knowledge of an area of interest, with validity similar to that of a cartographic plane. Nevertheless, the resolution of the orthophoto needs to be as high as possible. For this, it is necessary to use a photogrammetry software that processes aerial images to generate 3D reconstructions or orthophotos. The software searches correspondences between images and determines the correct which are its probable positions, based on the different points of view of the same element, in a process called stitching. Commercial software offers different photogrammetry services, some base on geometry and pixel values of the images.

The current capabilities of photogrammetry and machine learning techniques have been integrated to revolutionize current workflows and allow many new ones. In this work, we propose a novel methodology to generate high-resolution

applied sciences

MDPI

Article

A Deep Learning Architecture For 3D Mapping Urban Landscapes

Armando Levid Rodríguez-Santiago [1], José Aníbal Arias-Aguilar [1,†], Hiroshi Takemura [2,†] and Alberto Elías Petrilli-Barceló [2,*,†]

[1] Graduate Studies Division, Universidad Tecnológica de la Mixteca, Km. 2.5 Carretera a Acatlima, Huajuapan de Léon 69000, Oaxaca, Mexico; levid.rodriguez@gmail.com (A.L.R.-S.); anibal@mixteco.utm.mx (J.A.A.-A.)

[2] Department of Mechanical Engineering, Faculty of Science and Technology, Tokyo University of Science, 2641 Yamazaki, Noda 278-8510, Chiba, Japan; takemura@rs.tus.ac.jp

* Correspondence: petrilli@rs.tus.ac.jp

† These authors contributed equally to this work.

Abstract: In this paper, an approach through a Deep Learning architecture for the three-dimensional reconstruction of outdoor environments in challenging terrain conditions is presented. The architecture proposed is configured as an Autoencoder. However, instead of the typical convolutional layers, some differences are proposed. The Encoder stage is set as a residual net with four residual blocks, which have been provided with the necessary knowledge to extract the feature maps from aerial images of outdoor environments. On the other hand, the Decoder stage is set as a Generative Adversarial Network (GAN) and called a GAN-Decoder. The proposed network architecture uses a sequence of the 2D aerial image as input. The Encoder stage works for the extraction of the vector of features that describe the input image, while the GAN-Decoder generates a point cloud based on the information obtained in the previous stage. By supplying a sequence of frames that a percentage of overlap between them, it is possible to determine the spatial location of each generated point. The experiments show that with this proposal it is possible to perform a 3D representation of an area flown over by a drone using the point cloud generated with a deep architecture that has a sequence of aerial 2D images as input. In comparison with other works, our proposed system is capable of performing three-dimensional reconstructions in challenging urban landscapes. Compared with the results obtained using commercial software, our proposal was able to generate reconstructions in less processing time, with less overlapping percentage between 2D images and is invariant to the type of flight path.

Keywords: deep learning; CNN; autoencoder; GAN; point cloud; 3D reconstruction; aerial images

Citation: Rodríguez-Santiago, A.L.; Arias-Aguilar, J.A.; Takemura, H.; Petrilli-Barceló, A.E. A Deep Learning Architecture For 3D Mapping Urban Landscapes. *Appl. Sci.* **2021**, *11*, 11551. https://doi.org/10.3390/app112311551

Academic Editor: Wonjoon Kim, Sekyoung Youm and Sungbum Jun

Received: 29 September 2021
Accepted: 30 November 2021
Published: 6 December 2021

1. Introduction

Three-dimensional reconstruction and visual representation is a broadly studied problem and can be used in many applications such as object recognition and scene understanding. State-of-the-art 3D reconstruction algorithms show important results and propose solutions to the Structure from Motion (SfM) and Simultaneous Localization And Mapping (SLAM) problems with important results and proposals that give a solution to these problems [1–11]. Techniques perform localization and mapping, and 3D reconstruction using active sensors (e.g., LiDAR scanners) and passive sensing (e.g., stereo cameras).

However, in 3D reconstruction, none of these methods perform well on practical scenarios, and given the ambiguous correspondences between pixels and 3D spatial points, projection from 2D to 3D remains remarkably difficult and intuitive, these models are typically incapable of producing reliable matches in regions with repetitive patterns, homogeneous appearance, or large illumination change, a typical problem in photogramatry [12–15]. The problem more challenging when working with aerial images of external environments.

Referencias

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al. (2016). Tensorflow: A system for large-scale machine learning. In *12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16)*, pages 265–283.

Afifi, A. J., Magnusson, J., Soomro, T. A., and Hellwich, O. (2020). Pixel2point: 3d object reconstruction from a single image using cnn and initial sphere. *IEEE Access*, 9:110–121.

Ahrens, S., Levine, D., Andrews, G., and How, J. P. (2009). Vision-based guidance and control of a hovering vehicle in unknown, gps-denied environments. In *2009 IEEE International Conference on Robotics and Automation*, pages 2643–2648. IEEE.

Alcantarilla, P. F., Bartoli, A., and Davison, A. J. (2012). Kaze features. In *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part VI 12*, pages 214–227. Springer.

Alcantarilla, P. F. and Solutions, T. (2011). Fast explicit diffusion for accelerated features in nonlinear scale spaces. *IEEE Trans. Patt. Anal. Mach. Intell*, 34(7):1281–1298.

Altwaijry, H., Veit, A., Belongie, S. J., and Tech, C. (2016). Learning to detect and match keypoints with deep architectures. In *BMVC*.

Amelie, B., Fei-Fei, L., Ranjay, K., and Danfei, X. (2020). Convolutional Neural Networks for Visual Recognition. https://cs231n.github.io/optimization-2/.

Amidi, A. (2019). Deep learning. https://stanford.edu/ shervine/.

Anitha, G. and Kumar, R. G. (2012). Vision based autonomous landing of an unmanned aerial vehicle. *Procedia Engineering*, 38:2250–2256.

Arandjelovic, R., Gronat, P., Torii, A., Pajdla, T., and Sivic, J. (2016). Netvlad: Cnn architecture for weakly supervised place recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5297–5307.

Arya, S., Mount, D. M., Netanyahu, N. S., Silverman, R., and Wu, A. Y. (1998). An optimal algorithm for approximate nearest neighbor searching fixed dimensions. *Journal of the ACM (JACM)*, 45(6):891–923.

AWS (2020). *AWS Foundations: Machine Learning Basics*. AWS.

Barazzetti, L., Remondino, F., and Scaioni, M. (2010). Extraction of accurate tie points for automated pose estimation of close-range blocks. In *ISPRS Technical Commission III Symposium on Photogrammetric Computer Vision and Image Analysis*.

Bay, H., Ess, A., Tuytelaars, T., and Van Gool, L. (2008). Speeded-up robust features (surf). *Computer vision and image understanding*, 110(3):346–359.

Baydin, A. G., Pearlmutter, B. A., Radul, A. A., and Siskind, J. M. (2017). Automatic differentiation in machine learning: a survey. *The Journal of Machine Learning Research*,

18(1):5595–5637.

Beardsley, P. A., Zisserman, A., and Murray, D. W. (1997). Sequential updating of projective and affine structure from motion. *International journal of computer vision*, 23(3):235–259.

Brown, M. and Lowe, D. G. (2007). Automatic panoramic image stitching using invariant features. *International journal of computer vision*, 74:59–73.

Bu, S., Zhao, Y., Wan, G., and Liu, Z. (2016). Map2dfusion: Real-time incremental uav image mosaicing based on monocular slam. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4564–4571. IEEE.

Cadena, C., Carlone, L., Carrillo, H., Latif, Y., Scaramuzza, D., Neira, J., Reid, I., and Leonard, J. J. (2016). Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Transactions on robotics*, 32(6):1309–1332.

Carlone, L., Tron, R., Daniilidis, K., and Dellaert, F. (2015). Initialization techniques for 3d slam: a survey on rotation estimation and its use in pose graph optimization. In *2015 IEEE international conference on robotics and automation (ICRA)*, pages 4597–4604. IEEE.

Charles, J., Abbeloos, W., Reino, D. O., and Cipolla, R. (2022). Style2nerf: An unsupervised one-shot nerf for semantic 3d reconstruction. In *BMVC*, page 104.

Chau, H. and Karol, R. (2014). Robust panoramic image stitching. *Department of Aeronautics and Astronautics Stanford University Stanford, CA, USA*.

Chen, C.-C. and Chu, H.-T. (2005). Similarity measurement between images. In *29th Annual International Computer Software and Applications Conference (COMPSAC'05)*, volume 2, pages 41–42. IEEE.

Chen, Y., Liu, L., Gong, Z., and Zhong, P. (2017). Learning CNN to pair UAV video image patches. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 10(12):5752–5768.

Cheng, X., Wang, P., Guan, C., and Yang, R. (2020). Cspn++: Learning context and resource aware convolutional spatial propagation networks for depth completion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 10615–10622.

Choe, J., Joo, K., Imtiaz, T., and Kweon, I. S. (2021). Volumetric propagation network: Stereo-lidar fusion for long-range depth estimation. *IEEE Robotics and Automation Letters*, 6(3):4672–4679.

Conte, G. and Doherty, P. (2008). An integrated uav navigation system based on aerial image matching. In *2008 IEEE Aerospace Conference*, pages 1–10. IEEE.

Creswell, A., White, T., Dumoulin, V., Arulkumaran, K., Sengupta, B., and Bharath, A. A. (2018). Generative adversarial networks: An overview. *IEEE Signal Processing Magazine*, 35(1):53–65.

Escalante Torrado, J. O., Porras Díaz, H., and et al. (2016). Ortomosaicos y modelos digitales de elevación generados a partir de imágenes tomadas con sistemas uav. *Tecnura*, 20(50):119–140.

Fei, B., Yang, W., Chen, W.-M., Li, Z., Li, Y., Ma, T., Hu, X., and Ma, L. (2022). Comprehensive review of deep learning-based 3d point cloud completion processing and analysis. *IEEE Transactions on Intelligent Transportation Systems*, 23(12):22862–22883.

Fischler, M. A. and Bolles, R. C. (1981). Random sample consensus: a paradigm for model

fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395.

Fu, M., Liang, H., Zhu, C., Dong, Z., Sun, R., Yue, Y., and Yang, Y. (2023). Image stitching techniques applied to plane or 3-d models: a review. *IEEE Sensors Journal*, 23(8):8060–8079.

Ghamisi, P. and Yokoya, N. (2018). Img2dsm: Height simulation from single imagery using conditional generative adversarial net. *IEEE Geoscience and Remote Sensing Letters*, 15(5):794–798.

Girshick, R. (2015). Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448.

Girshick, R., Donahue, J., Darrell, T., and Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587.

Golparvar-Fard, M., Pena-Mora, F., and Savarese, S. (2011). Monitoring changes of 3d building elements from unordered photo collections. In *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pages 249–256. IEEE.

Goodfellow, I., Bengio, Y., Courville, A., and Bengio, Y. (2016). *Deep learning*, volume 1. MIT press Cambridge.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets in: Advances in neural information processing systems (nips). *Adv. Neural Inf. Process. Syst. (NIPS)*, 27(1):2672–2680.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2020). Generative adversarial networks. *Communications of the ACM*, 63(11):139–144.

Gordo, A., Almazán, J., Revaud, J., and Larlus, D. (2016). Deep image retrieval: Learning global representations for image search. In *European conference on computer vision*, pages 241–257. Springer.

Guo, Y., Wang, H., Hu, Q., Liu, H., Liu, L., and Bennamoun, M. (2020). Deep learning for 3d point clouds: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 43(12):4338–4364.

Gupta, P., Srivastava, P., Bhardwaj, S., and Bhateja, V. (2011). A modified psnr metric based on hvs for quality assessment of color images. In *2011 International Conference on Communication and Industrial Application*, pages 1–4. IEEE.

Häming, K. and Peters, G. (2010). The structure-from-motion reconstruction pipeline–a survey with focus on short image sequences. *Kybernetika*, 46(5):926–937.

He, K. and Sun, J. (2015). Convolutional neural networks at constrained time cost. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5353–5360.

He, K., Zhang, X., Ren, S., and Sun, J. (2015a). Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034.

He, K., Zhang, X., Ren, S., and Sun, J. (2015b). Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 37(9):1904–1916.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Hore, A. and Ziou, D. (2010). Image quality metrics: Psnr vs. ssim. In *2010 20th international conference on pattern recognition*, pages 2366–2369. IEEE.

Hui, Z., Jin, S., Cheng, P., Ziggah, Y. Y., Wang, L., Wang, Y., Hu, H., and Hu, Y. (2019). An active learning method for dem extraction from airborne lidar point clouds. *IEEE Access*, 7:89366–89378.

Karen, S. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv: 1409.1556*.

Kingma, D. P. and Ba, J. (2015). Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations ICLR 2015, San Diego, CA, USA*.

Kurenkov, A., Ji, J., Garg, A., Mehta, V., Gwak, J., Choy, C., and Savarese, S. (2018). Deformnet: Free-form deformation network for 3d shape reconstruction from a single image. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 858–866. IEEE.

Le, C. and Li, X. (2019). Jigsawnet: Shredded image reassembly using convolutional neural network and loop-based composition. *IEEE Transactions on Image Processing*, 28(8):4000–4015.

LeCun, Y. e. a. (1989). Generalization and network design strategies. *Connectionism in perspective*, 19:143–155.

Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., and Wang, Z. (2017). Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4681–4690.

Leutenegger, S., Chli, M., and Siegwart, R. Y. (2011). Brisk: Binary robust invariant scalable keypoints. In *2011 International conference on computer vision*, pages 2548–2555. Ieee.

Li, J., Ai, M., Hu, Q., and Fu, D. (2014). A novel approach to generating dsm from high-resolution uav images. In *2014 22nd International Conference on Geoinformatics*, pages 1–5. IEEE.

Li, S., Zhu, Z., Wang, H., and Xu, F. (2019). 3d virtual urban scene reconstruction from a single optical remote sensing image. *IEEE Access*, 7:68305–68315.

Li, X., Liu, Y., Wang, Y., and Yan, D. (2005). Computing homography with ransac algorithm: a novel method of registration. In *Electronic Imaging and Multimedia Technology IV*, volume 5637, pages 109–112. International Society for Optics and Photonics.

Lin, C.-C., Pankanti, S. U., Natesan Ramamurthy, K., and Aravkin, A. Y. (2015). Adaptive as-natural-as-possible image stitching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1155–1163.

Lingua, A., Marenchino, D., and Nex, F. (2009a). Automatic digital surface model (dsm) generation procedure from images acquired by unmanned aerial systems (uass). *RevCAD: Journal of Geodesy and Cadastre*, 9:53–64.

Lingua, A., Marenchino, D., and Nex, F. (2009b). A comparison between "old and new " feature extraction and matching techniques in photogrammetry. *RevCAD: Journal of*

Geodesy and Cadastre, 9:43–52.

Long, J., Shelhamer, E., and Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440.

Lowe, D. G. (1999). Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pages 1150–1157. Ieee.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60:91–110.

Lu, Q., Lu, Y., Xiao, M., Yuan, X., and Jia, W. (2019a). 3d-fhnet: Three-dimensional fusion hierarchical reconstruction method for any number of views. *IEEE Access*, 7:172902–172912.

Lu, Q., Xiao, M., Lu, Y., Yuan, X., and Yu, Y. (2019b). Attention-based dense point cloud reconstruction from a single image. *IEEE Access*, 7:137420–137431.

Makhzani, A., Shlens, J., Jaitly, N., Goodfellow, I., and Frey, B. (2015). Adversarial Autoencoders. *arXiv e-prints*, page arXiv:1511.05644.

Mandikal, P., Navaneet, K. L., Agarwal, M., and Venkatesh Babu, R. (2018). 3D-LMNet: Latent Embedding Matching for Accurate and Diverse 3D Point Cloud Reconstruction from a Single Image. *arXiv e-prints*, page arXiv:1807.07796.

Mandikal, P. and Radhakrishnan, V. B. (2019). Dense 3d point cloud reconstruction using a deep pyramid network. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1052–1060. IEEE.

Martínez-Otzeta, J. M., Rodríguez-Moreno, I., Mendialdua, I., and Sierra, B. (2023). Ransac for robotic applications: A survey. *Sensors*, 23(1).

Metz, L., Poole, B., Pfau, D., and Sohl-Dickstein, J. (2016). Unrolled Generative Adversarial Networks. *arXiv e-prints*, page arXiv:1611.02163.

Mohammadi, M. E., Watson, D. P., and Wood, R. L. (2019). Deep learning-based damage detection from aerial sfm point clouds. *Drones*, 3(3):68.

Mouragnon, E., Lhuillier, M., Dhome, M., Dekeyser, F., and Sayd, P. (2009). Generic and real-time structure from motion using local bundle adjustment. *Image and Vision Computing*, 27(8):1178–1193.

Murphy, K. P. (2012). *Machine learning: a probabilistic perspective*. MIT press.

Nacken, P. F. (1994). Chamfer metrics in mathematical morphology. *Journal of Mathematical Imaging and Vision*, 4(3):233–253.

Navaneet, K., Mandikal, P., Agarwal, M., and Babu, R. V. (2019). Capnet: Continuous approximation projection for 3d point cloud reconstruction using 2d supervision. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8819–8826.

Ni, D., Nee, A., Ong, S., Li, H., Zhu, C., and Song, A. (2018). Point cloud augmented virtual reality environment with haptic constraints for teleoperation. *Transactions of the Institute of Measurement and Control*, 40(15):4091–4104.

Nikolakopoulos, K. G., Soura, K., Koukouvelas, I. K., and Argyropoulos, N. G. (2017). Uav vs classical aerial photogrammetry for archaeological studies. *Journal of Archaeological Science: Reports*, 14:758–773.

Pinaya, W. H. L., Vieira, S., Garcia-Dias, R., and Mechelli, A. (2020). Autoencoders. In *Machine learning*, pages 193–208. Elsevier.

Pix4D (2019). Machine learning meets photogrammetry. https://www.pix4d.com/.

Pollefeys, M., Van Gool, L., Vergauwen, M., Verbiest, F., Cornelis, K., Tops, J., and Koch, R. (2004). Visual modeling with a hand-held camera. *International Journal of Computer Vision*, 59(3):207–232.

Radenović, F., Tolias, G., and Chum, O. (2016). Cnn image retrieval learns from bow: Unsupervised fine-tuning with hard examples. In *European conference on computer vision*, pages 3–20. Springer.

Ren, R., Fu, H., and Wu, M. (2019). Large-scale outdoor slam based on 2d lidar. *Electronics*, 8(6):613.

Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28.

Rodríguez Santiago, A. L. (2013). Implementación de un sistema de visión mono-cámara basado en el registro de imágenes 2d. Master's thesis, Universidad Tecnológica De La Mixteca, Huajuapan de León, Oaxaca, México.

Rodríguez-Santiago, A. L., Arias-Aguilar, J. A., Takemura, H., and Petrilli-Barceló, A. E. (2021a). A deep learning architecture for 3d mapping urban landscapes. *Applied Sciences*, 11(23):11551.

Rodríguez-Santiago, A. L., Arias-Aguilar, J. A., Takemura, H., and Petrilli-Barcelo, A. E. (2021b). High-resolution reconstructions of aerial images based on deep learning. *Computación y Sistemas*, 25(4):739–749.

Rothermel, M., Wenzel, K., Fritsch, D., and Haala, N. (2012). Sure: Photogrammetric surface reconstruction from imagery. In *Proceedings LC3D Workshop, Berlin*, volume 8.

Rublee, E., Rabaud, V., Konolige, K., and Bradski, G. (2011). Orb: An efficient alternative to sift or surf. In *2011 International conference on computer vision*, pages 2564–2571. Ieee.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., and Bernstein, M. (2015). Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252.

Saripalli, S., Montgomery, J. F., and Sukhatme, G. S. (2002). Vision-based autonomous landing of an unmanned aerial vehicle. In *Proceedings 2002 IEEE international conference on robotics and automation (Cat. No. 02CH37292)*, volume 3, pages 2799–2804. IEEE.

Sazonov, I., Wang, D., Hassan, O., Morgan, K., and Weatherill, N. (2006). A stitching method for the generation of unstructured meshes for use with co-volume solution techniques. *Computer Methods in Applied Mechanics and Engineering*, 195(13-16):1826–1845.

Schlosser, A. D., Szabó, G., Bertalan, L., Varga, Z., Enyedi, P., and Szabó, S. (2020). Building extraction using orthophotos and dense point cloud derived from visual band aerial imagery based on machine learning and segmentation. *Remote Sensing*, 12(15):2397.

Schonberger, J. L. and Frahm, J.-M. (2016). Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113.

Shi, W., Caballero, J., Huszár, F., Totz, J., Aitken, A. P., Bishop, R., Rueckert, D., and Wang, Z. (2016). Real-time single image and video super-resolution using an efficient sub-pixel

convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1874–1883.

Solai, P. (2018). Convolutions and Backpropagations. https://medium.com/@pavisj/convolutions-and-backpropagations-46026a8f5d2c.

Srivastava, R. K., Greff, K., and Schmidhuber, J. (2015). Highway networks. *arXiv preprint arXiv:1505.00387*.

Stachniss, C., Hahnel, D., and Burgard, W. (2004). Exploration with active loop-closing for fastslam. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566)*, volume 2, pages 1505–1510. IEEE.

Sun, Y., Lv, Y., Song, B., Guo, Y., and Zhou, L. (2021). Image stitching method of aerial image based on feature matching and iterative optimization. In *2021 40th Chinese Control Conference (CCC)*, pages 3024–3029. IEEE.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9.

Szeliski, R. et al. (2007). Image alignment and stitching: A tutorial. *Foundations and Trends® in Computer Graphics and Vision*, 2(1):1–104.

Tang, J., Folkesson, J., and Jensfelt, P. (2018). Geometric correspondence network for camera motion estimation. *IEEE Robotics and Automation Letters*, 3(2):1010–1017.

Wang, H., Jiang, Z., Yi, L., Mo, K., Su, H., and Guibas, L. J. (2020). Rethinking Sampling in 3D Point Cloud Generative Adversarial Networks. *arXiv e-prints*, page arXiv:2006.07029.

Wang, L., Zhang, Y., and Feng, J. (2005). On the euclidean distance of images. *IEEE transactions on pattern analysis and machine intelligence*, 27(8):1334–1339.

Wang, N., Zhang, Y., Li, Z., Fu, Y., Liu, W., and Jiang, Y.-G. (2018). Pixel2mesh: Generating 3d mesh models from single rgb images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 52–67.

Wang, T.-H., Hu, H.-N., Lin, C. H., Tsai, Y.-H., Chiu, W.-C., and Sun, M. (2019a). 3d lidar and stereo fusion using stereo matching network with conditional cost volume normalization. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5895–5902. IEEE.

Wang, X., Yu, K., Wu, S., Gu, J., Liu, Y., Dong, C., Qiao, Y., and Loy, C. C. (2019b). Esrgan: Enhanced super-resolution generative adversarial networks. In *Computer Vision – ECCV 2018 Workshops*, pages 63–79. Springer International Publishing.

Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P. (2004). Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612.

Wang, Z. and Yang, Z. (2020). Review on image-stitching techniques. *Multimedia Systems*, 26(4):413–430.

Weerasekera, C. S., Latif, Y., Garg, R., and Reid, I. (2017). Dense monocular reconstruction using surface normals. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2524–2531. IEEE.

Wu, B., Zhou, X., Zhao, S., Yue, X., and Keutzer, K. (2019). Squeezesegv2: Improved model structure and unsupervised domain adaptation for road-object segmentation from a lidar

point cloud. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 4376–4382.

Wu, S., Zhong, S., and Liu, Y. (2018). Deep residual learning for image steganalysis. *Multimedia tools and applications*, 77(9):10437–10453.

Yang, C.-Y., Ma, C., and Yang, M.-H. (2014). Single-image super-resolution: A benchmark. In *European Conference on Computer Vision*, pages 372–386. Springer.

Yeu, C.-W., Lim, M.-H., Huang, G.-B., Agarwal, A., and Ong, Y.-S. (2006). A new machine learning paradigm for terrain reconstruction. *IEEE Geoscience and Remote Sensing Letters*, 3(3):382–386.

Zamorski, M., Zięba, M., Klukowski, P., Nowak, R., Kurach, K., Stokowiec, W., and Trzciński, T. (2020). Adversarial autoencoders for compact representations of 3d point clouds. *Computer Vision and Image Understanding*, 193:102921.

Zhang, K. and Li, X. (2014). A graph-based optimization algorithm for fragmented image reassembly. *Graphical Models*, 76(5):484–495.

Zhang, Y., Liu, Z., Liu, T., Peng, B., and Li, X. (2019a). Realpoint3d: An efficient generation network for 3d object reconstruction from a single image. *IEEE Access*, 7:57539–57549.

Zhang, Y., Wu, H., and Yang, W. (2019b). Forests growth monitoring based on tree canopy 3d reconstruction using uav aerial photogrammetry. *Forests*, 10(12):1052.

Zhang, Z. (2016). Derivation of backpropagation in convolutional neural network (cnn). *University of Tennessee, Knoxville, TN*.

Zhang, Z., Zhang, Y., Zhao, X., and Gao, Y. (2018). Emd metric learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Zheng, J., Zhu, Y., Wang, K., Zou, Q., and Zhou, Z. (2022). Deep learning assisted optimization for 3d reconstruction from single 2d line drawings. *arXiv preprint arXiv:2209.02692*.

Zhou, Y.-T. and Chellappa, R. (1988). Computation of optical flow using a neural network. In *ICNN*, pages 71–78.