

UNIVERSIDAD AMAZÓNICA DE PANDO
ÁREA DE CIENCIAS Y TECNOLOGÍA
CARRERA INGENIERÍA DE SISTEMAS



TESIS DE GRADO

**MODELO FOTOGRAMÉTRICO DEEP LEARNING PARA LA DETECCIÓN DE
VIVIENDAS EN LOS PREDIOS DE LA ZONA CENTRAL DEL MUNICIPIO DE
COBIJA**

**TESIS DE GRADO PRESENTADO PARA OBTENER EL TÍTULO ACADÉMICO DE
LICENCIADO EN INGENIERÍA DE SISTEMAS**

Postulante: David Mallea Quispe

Tutor: MSc. Lic. Humberto Fernández Calle

Asesor: MSc. Ing. Alex Yanahuaya Arce

Cobija - Pando – Bolivia

2020

AGRADECIMIENTOS

Agradezco a Dios quien es el guía de mi camino y está en cada paso que doy, por su infinita bondad y haberme dado salud para lograr mis objetivos.

Por supuesto a mi familia, a mis padres y hermanos, son ustedes un Modelo de constancia y de lucha, gracias familia por estar siempre apoyándome, quiero hacerles saber que son y siempre serán mi principal motivación.

A mi tutor metodológico, M.Sc. Lic. Humberto Fernández Calle por su colaboración, en el proceso de desarrollo de la tesis.

A mi asesor, M.Sc. Ing. Alex Yanahuaya Arce de manera especial por haberme brindado la oportunidad y confianza en el desarrollo de la tesis. Fue gracias a usted Ingeniero que pude realizar el objetivo académico más importante de mi vida hasta ahora, por eso y muchas cosas más le estaré eternamente agradecido.

A mis amigos de la Carrera de Ingeniería de Sistemas, con quienes estuvimos juntos por este camino de aprendizaje.

A la Universidad Amazónica de Pando, por acogerme y brindarme muchas oportunidades.

A cada uno de los docentes que intervinieron en mi formación como profesional a lo largo de la carrera.

DEDICATORIA

A mi padre, Juan Luis Mallea Uriarte, por ser un ejemplo de vida y superación, por haber sacrificado gran parte de su vida para formarme y educarme, por todo el cariño y amor que me brinda.

A mi madre Elena Quispe Salazar, tus Impresionantes esfuerzos son invaluable. Junto a mi padre me has educado, brindándome cuidados y atención, eres la madre más hermosa del universo, donde me encontraría de no ser por tu ayuda, tu compañía y tu amor, eres el pilar fundamental en mi familia.

A mi hija Maylin Mallea Villca eres el tesoro más grande que Dios me dio, la razón de mi existencia y mi motivación para surgir más.

A mis hermanos Luis Esteban, Marcelo, Luis y Nely, quienes, a lo largo de mi vida, he aprendido mucho de ustedes, hermanos son los mejores amigos de mi vida.

RESUMEN

La fotogrametría de drones es una tendencia actual y creciente que acompaña a las nuevas tecnologías en el campo de la ingeniería, así mismo el Deep Learning es una tecnología que se asocia a la tecnología de sistemas de información geográfica. El problema principal de la investigación consiste en: ¿Cómo detectar las viviendas de los predios de la zona central del Municipio de Cobija a través de Tecnología de Redes Neuronales? El objetivo de la tesis de grado es Diseñar el Modelo Fotogramétrico Deep Learning para la detección de vivienda en los predios de la Zona Central del Municipio de Cobija. Para responder a la pregunta de investigación, se diseña una metodología de levantamiento fotogramétrico con drones, desde el análisis y estudio de la región a aplicar la investigación, luego se captura y procesa las imágenes posteriormente se diseña la inclusión de Deep Learning para establecer un modelo de detección de viviendas en el centro del Municipio de Cobija. Los principales resultados de la investigación en relación al modelo fotogramétrico Deep Learning son: la altura ideal a base de la resolución espacial (GSD), para cada muestra obtenida, en la muestra 1 a 77m. y GSD 0.018m/px., muestra 2 a 76m GSD 0.017m/px y muestra 3 a 70m GSD 0.0164 m/px. que permite obtener un ortomosaico de calidad donde se visualizan las viviendas del Centro del Municipio de Cobija. En cuanto al procesamiento de imágenes, se obtuvo 235 imágenes procesadas en el software PIX4D. Mediante el modelo diseñado se obtuvo resultados aceptables tales como en la muestra 1 a un 87.9%, muestra 2 a un 74.1% y muestra 3 a un 82.9%, y en conjunto el modelo aplicado a todas las muestras tuvo un resultado de 84.3%, logrando detectar un total de 122 viviendas. En el modelo de detección de vivienda surge una dificultad por la densidad vegetal que cubre partes de las viviendas, haciéndolo no detectable o detectar una falsa vivienda, otra dificultad está en la diversidad de materiales de construcción de los techos, automóviles o sombrillas de las motocicletas los mismos que pueden ser detectados por el modelo como viviendas. Por lo tanto; Se recomienda estandarizar los techados o terrazas mediante el ente regulador de viviendas (Catastro) del Municipio de Cobija, para poder establecer un patrón de buena calidad y generar un data set más preciso que ayude a detectar las viviendas.

Palabras Clave: Fotogramétrico, Drones, Deep Learning

ABSTRACT

Drone photogrammetry is a current and growing trend that accompanies new technologies in the field of engineering. Likewise, Deep Learning is a technology that is being established and associated in the technology of geographic information systems, therefore, the main problem of the research consists of How to detect the houses of the properties of the central area of (GSD)the Municipality of Cobija through Neural Network Technology? The objective is to Design the Deep Learning Photogrammetric Model for the detection of housing in the properties of the Central Zone of the Municipality of Cobija. A methodology for photogrammetric survey with drones is designed, from the analysis and study of the region to apply the research, Image capture and image processing, therefore a methodology is also designed for the inclusion of Deep Learning considering from the acquisition of data sets , extraction of patterns, detection and evaluation to establish a housing detection model in the center of the Municipality of Cobija. By applying the aforementioned methodologies, the results were obtained, the ideal height was found based on spatial resolution (GSD), for each sample obtained, in sample 1 at 77m. and GSD 0.018m / px., shows 2 at 76m GSD 0.017m / px and shows 3 at 70m GSD 0.0164 m / px. to achieve a quality orthomosaic where the houses of the Municipality of Cobija are displayed. Regarding the image processing, 235 images processed in the PIX4D software were obtained. Therefore, the model designed for the detection of dwellings, obtained acceptable results, having from each sample different results measured by the confusion matrix and applying the Accuracy formula (Accuracy), sample 1 at 87.9%, sample 2 at 74.1% and sample 3 at 82.9%, and as a whole the model applied to all samples had a result of 84.3%, managing to detect a total of 122 homes. In the housing detection model, a difficulty arises due to the plant density that covers parts of the houses and deteriorating an established pattern of the construction material and confusing that pattern to the model making it not detect or detect a false home, another difficulty consists of diversity of construction materials or deteriorated confusing the model by detecting cars or motorcycle umbrellas. Thus; It is recommended to standardize the roofs or terraces through the housing regulator (Cadastre) of the Municipality of Cobija, in order to establish a good quality pattern and generate a more precise data set that helps to detect the dwellings.

Keywords: Photogrammetric, Drones, Deep Learning

ÍNDICE

	Pág.
CAPITULO I	1
1.MARCO INTRODUCTORIO	1
1.1 INTRODUCCIÓN	2
1.2 PLANTEAMIENTO DEL PROBLEMA	2
1.2.1 Problema Principal	3
1.3 HIPÓTESIS	3
1.3.1 Variables	3
1.3.2 Operacionalización de Variables	3
1.4 OBJETIVOS	4
1.4.1 Objetivo General	4
1.4.2 Objetivos Específicos	4
1.5 JUSTIFICACION	5
1.5.1 Económica	5
1.5.2 Social	5
1.5.3 Científica	5
1.6 ALCANCES Y LIMITES	5
1.6.1 Alcances	5
1.6.2 Limites	6
CAPITULO II	7
2. MARCO TEÓRICO	7
2.1 DEEP LEARNING	8
2.1.1 Aplicaciones prácticas de Deep Learning	8
2.1.2 Red Neuronal Convolutacional	9
2.1.3 Arquitectura Red Neuronal Convolutacional	10
2.1.4 Aprendizaje Supervisado	12
2.1.5 TensorFlow	12
2.1.6 Python	13
2.1.7 Clasificación de Pixeles en Sistemas de información Geográfica	13
2.1.8 Detectar Objetos en Sistemas de información Geográfica	13
2.1.9 Rasters	13

2.2 SISTEMAS DE INFORMACIÓN GEOGRÁFICA (SIG)	15
2.2.1 Componentes de los SIG.....	15
2.2.2 Fotogrametría.....	16
2.2.3 Proceso Fotogramétrico.....	17
2.2.4 Adquisición de Imágenes.....	17
2.2.5 Modelo Fotogramétrico Deep Learning.....	19
2.3 DRONE	19
2.3.1 Usos de los drones	19
2.3.2 Geomagnetismo	20
2.4 ESTADO DEL ARTE	20
2.4.1 Investigaciones Internacionales	20
2.4.2 Investigación Nacional.....	22
CAPITULO III	23
3. MARCO METODOLÓGICO	23
3.1 ENFOQUE DE LA INVESTIGACIÓN	24
3.1.2 Diseño de la Metodología.....	24
3.2 DESGLOSE DE DISEÑO DE LA METODOLOGÍAS	25
3.2.1 METODOLOGÍA DE LEVANTAMIENTO FOTOGRAMÉTRICO CON DRONES	25
3.2.1.1 Análisis y Estudio de la región a aplicar la investigación.....	27
3.3.1.2 Captura de Imágenes	29
3.3.1.3 Procesamiento de Imágenes	35
3.2.2 METODOLOGÍA DE INCLUSIÓN DE DEEP LEARNING	38
3.2.2.1 Adquisición de Data Sets:.....	38
3.2.2.2 Extracción de Patrones.....	39
3.2.2.3 Detección	39
3.2.2.4 Evaluación.....	40
3.3 POBLACIÓN Y MUESTRA	41
3.4 TÉCNICAS DE PROCESAMIENTO Y ANÁLISIS DE DATOS	41
3.6.1 Matriz de Confusión	42
3.6.2 TensorBoard	43
CAPITULO IV	44
4. DESARROLLO DE LA INVESTIGACIÓN	44

4.1 INTRODUCCIÓN	45
4.2 LEVANTAMIENTO FOTOGRAMÉTRICO CON DRONES	45
4.2.1 Análisis y Estudio de la región a aplicar la investigación	45
4.2.2. Captura de Imágenes	47
4.2.3. Procesamiento de Imágenes	54
4.3 INCLUSIÓN DE DEEP LEARNING	56
4.3.1 Adquisición de Data Sets:	57
4.3.2 Extracción de Patrones	58
4.3.3 Detección	60
4.3.4 Evaluación	65
4.4 RESULTADOS DE LA INVESTIGACIÓN	69
CAPITULO V	71
5. CONCLUSIONES Y RECOMENDACIONES	71
5.1 CONCLUSIONES	72
5.2 RECOMENDACIONES	73
6 GLOSARIO	74
7 REFERENCIA BIBLIOGRAFÍA	75
8 ANEXOS	78
8.1 A- ARBOL DE PROBLEMAS	78
8.2 B- MATRIZ DE CONSISTENCIA	78
8.3 C- CODIGO-JUPITER-NOTEBBOK CONVERTIDOR DE PASCAL	79
8.4 D- CODIGO DEL MODELO FOTOGRAMETRICO DEEP LEARNING DE DETECCIÓN DE VIVIENDAS	81

ÍNDICE DE TABLA

Tabla 1: Operacionalización de Variables Independientes	4
Tabla 2: Operacionalización de Variables Dependientes	4
Tabla 3: Dimensiones de la región de interés	29
Tabla 4: Rango de Vuelo Permitido con el dron e	29
Tabla 5: Parámetros climáticos y geomagnetismo.....	31
Tabla 6: Formula de Altura.....	31
Tabla 7: Muestreo de Altura	32
Tabla 8: Puntos de Referencia	32
Tabla 9: Cuadro de datos al Iniciar el Vuelo Planificado	33

Tabla 10: Resistencia al Viento	33
Tabla 11: Lista de App para Vuelo Planificado Elaboración Propia	34
Tabla 12: Formula de Obturación de Cámara.....	34
Tabla 13: Obturación de disparo.....	35
Tabla 14: Referencia de Misión de Vuelo	35
Tabla 15: Software de Procesamiento de Imágenes (ortomosaico).....	36
Tabla 16: Equipo Computacional Requerido.....	36
Tabla 17: Herramientas para establecer un marco de trabajo	36
Tabla 18: Proceso de Imágenes.....	37
Tabla 19 Ubicación y versión de instalación de dependencias de Deep Learning para documentar.....	39
Tabla 20 Salida Inclusión de Deep Learning	40
Tabla 21 Evaluación de Muestra DEM-Ortomosaico	41
Tabla 22: Matriz de Confusión	42
Tabla 23: TP, FP, FN y TN (Matriz de confusión).....	42
Tabla 24: Informe de Condiciones físicas del Territorio.....	45
Tabla 25: Dimensiones de la región de interés	47
Tabla 26: Parámetros climáticos y geomagnetismo del día.....	48
Tabla 27: Resultado de Muestreo N° 1	49
Tabla 28: Resultado de Muestreo de Altura N°2.....	50
Tabla 29: Resultado de Muestreo N°3	51
Tabla 30: Puntos de referencia levantados.....	51
Tabla 31: Verificación de Vuelo N°1	51
Tabla 32: Verificación de Vuelo N°2	52
Tabla 33: Verificación de Vuelo N°3	52
Tabla 34: Muestra N°1 Obturación de disparo	52
Tabla 35: Muestra N°2 Obturación de disparo	52
Tabla 36: Muestra N°3 Obturación de disparo	53
Tabla 37: Referencia de Vuelo Muestra N°1	53
Tabla 38: Referencia de Vuelo Muestra N°2.....	53
Tabla 39: Referencia de Vuelo Muestra N°3.....	54
Tabla 40: Proceso de imágenes de muestras 1,2 y 3.....	56
Tabla 41 Ubicación y versión de instalación de dependencias de Deep Learning	59
Tabla 42 Muestra 1 con la inclusión de Deep Learning.....	62
Tabla 43 Muestra 2 con la Inclusión de Deep Learning	63
Tabla 44 Muestra 3 con la inclusión de Deep Learning.....	64
Tabla 45 Evaluación de la Muestra 1	65
Tabla 46 Evaluación de la Muestra 2	66
Tabla 47 Evaluación de la Muestra 3	67
Tabla 48 Evaluación de Matriz de confusión Muestra 1.....	67
Tabla 49 Evaluación de Matriz de confusión Muestra 2.....	68
Tabla 50 Evaluación de Matriz de Confusión Muestra 3.....	68
Tabla 51 Evaluación de Matriz de confusión en conjunto de las muestras.....	69

Tabla 52 Comparación de Viviendas Detectadas con Viviendas del Ortomosaico	70
---	----

ÍNDICE DE FIGURA

Figura 1: Red Neuronal Convolutacional	9
Figura 2: Arquitectura de una red Neuronal Convolutacional.....	10
Figura 3: Capa de Partida Convolución.....	11
Figura 4: Capa de reducción	11
Figura 5: Ráster en una Imagen	14
Figura 6: Factores de vuelo.....	18
Figura 7: Aumento de Grado de Resolución	20
Figura 8: Diseño de la Metodología	24
Figura 9: Desglose de la Metodología de Fotogrametría con Drones	26
Figura 10: Restricción Zonas Geo de DJ.....	27
Figura 11: Significado de las Zonas GEO por Colores	28
Figura 12: Conexión de Satélites con el Drone.....	30
Figura 13: Drone con Tecnología RTK	30
Figura 14: Metodología Inclusión de Deep Learning Red Convolutacional.....	38
Figura 15: Interfaz de TensorBoard.....	43
Figura 16 Ortomosaicos y su formato.....	57
Figura 17 Diversidad del Ortomosaico	57
Figura 18 Etiquetado para el Aprendizaje Supervisado.....	58
Figura19 Entrada Inicio Data Sets en la Plataforma ArcGis Pro.....	59
Figura 20 Proceso de Entrenamiento	60
Figura 21 Entrada y Proceso de Inicio de Detección.....	61
Figura 22 Comparativa de Exactitud (Accuracy) de las Muestras 1,2 y 3.....	68
Figura 23: Árbol de Problemas.....	78
Figura 24 Convertidor de Pascal Casilla 1	79
Figura 25 Convertidor de Pascal Casilla 2.....	79
Figura 26 Convertidor de Pascal Casilla 3.....	80
Figura 27 Convertidor de Pascal Casilla 4.....	80
Figura 28: Código del Modelo Fotogramétrico Deep Learning para la detección de viviendas	81

CAPITULO I

1.MARCO INTRODUCTORIO

1.1 INTRODUCCIÓN

Continuamente, nos vemos ante avances tecnológicos que afecta a todos los aspectos de la vida, es decir, tanto en la cotidianidad, la ciencia, la educación, etc., esta situación nos lleva a investigar acerca los beneficios que representa en el ámbito profesional de la Ingeniería de Sistemas; siendo más específicos, hablamos de la denominación con Deep Learning para la de geográficos con la ayuda de Drones.

La fotogrametría es una colección de técnicas que permiten hacer reconstrucciones de terrenos y medidas a partir de imágenes aéreas. La fotogrametría con drones es una tendencia actual y creciente que va de la mano de las nuevas tecnologías en el campo de la ingeniería.

Deep Learning es una tecnología que se está estableciendo y contextualizando dentro de la tecnología de sistemas de información geográfica dando un enfoque nuevo, de ver de diferente perspectiva teniendo en cuenta su evolución y establecer funcionalidades específicas que ayuden a un entorno de desarrollo

La problemática abordada tiene que ver con Deep Learning como componente de aplicación a los Sistemas de Información Geográfica, siguiendo el objetivo principal de conocer a cabalidad las características y funcionamiento de esta tecnología de reciente incursión en el mundo; de igual forma, la finalidad de la presente investigación es proponer el uso de la Deep Learning como una herramienta que permita mejorar el flujo y forma de detección de vivienda con el uso de drones.

1.2 PLANTEAMIENTO DEL PROBLEMA

Es esencial conocer adecuadamente el funcionamiento de Deep Learning (Aprendizaje Profundo) a base de Redes Convolucionales dedicado a imágenes fotogramétricas mediante el uso de drones, para la detección de vivienda de los predios del Municipio de Cobija para la obtención de información.

Debido a la falta de conocimiento en el área tecnológica de Inteligencia Artificial (Deep Learning) y drones, se denota un atraso en el procesamiento de datos por tanto se

presenta demoras en procesar datos para pasar a información real por la mala manipulación de software capaz de predecir y determinar resultados fiables.

En Municipio de Cobija se establece realizar un mapeo estandarizado con un dron adecuado dedicado a la región y procesar las imágenes (ortomosaicos) en un software específico (ArcGis) que soporte Deep Learning y las herramientas del lenguaje Python, donde se diseñara el modelo con Red Neuronales Convolucionales para la detección de vivienda de las imágenes fotogramétricas. Buscando soluciones adecuadas, se puede pronosticar un mejor flujo de trabajo, que permita mejorar los tiempos y esfuerzos al momento de desempeñar la labor de detección de viviendas.

1.2.1 Problema Principal

¿Cómo detectar las viviendas de los predios de la zona central del Municipio de Cobija a través de Tecnología de Redes Neuronales?

1.3 HIPÓTESIS

El Modelo Fotogramétrico Deep Learning permite la detección de viviendas en los predios de la Zona Central del Municipio de Cobija.

1.3.1 Variables

Variable independiente

Fotogramétrico, Deep Learning

Variable dependiente

Detección de viviendas

1.3.2 Operacionalización de Variables

	Variables	Conceptualización	Dimensión	Indicadores	Técnicas	Instrumentos
V · I ·	Deep Learning	Modelo imita la conectividad del cerebro humano, clasificando conjuntos de datos y encontrando correlaciones entre ellos.	Red neuronal Convolutiva	-Desempeño del Modelo	-Arquitectura de Deep Learning	-TensorFlow -ArcGisPro -Pix4D
			Aprendizaje supervisado			
			Modelo de reconocimiento de imágenes		-Modelos de reconocimiento de imágenes	

Variables		Conceptualización	Dimensión	Indicadores	Técnicas	Instrumentos
V · I ·	Fotogrametría	Es el arte, ciencia y tecnología de obtener información fidedigna de los objetos físicos mediante proceso de registro medición e interpretación	Imágenes ortomosaicos	-Procesamiento de Imágenes	-Metodología de Levantamiento fotogramétrico	-Muestreo de imágenes (Rstudio)

Tabla 1: Operacionalización de Variables Independientes

Fuente: Elaboración Propia

Variables		Conceptualización	Dimensión	Indicadores	Técnicas	Instrumentos
V.D	Detección de viviendas	Superioridad de poder separar aspectos catastrales como ser predios	Eficiencia de abstracción	-Tiempo de ejecución -Proceso de abstracción	-Métricas de Eficiencia de Aprendizaje	-Matriz de confusión

Tabla 2: Operacionalización de Variables Dependientes

Fuente: Elaboración Propia

1.4 OBJETIVOS

1.4.1 Objetivo General

Diseñar el Modelo Fotogramétrico Deep Learning para la detección de vivienda en los predios de la Zona Central del Municipio de Cobija.

1.4.2 Objetivos Específicos

- Desarrollar el levantamiento fotogramétrico mediante el uso del drone adecuado, para luego procesar las imágenes (ortomosaico) de los predios seleccionados.
- Establecer el conjunto de patrones de las imágenes fotogramétricas procesadas para el entrenamiento del modelo.
- Definir una arquitectura Deep Learning para la detección de vivienda de las imágenes fotogramétricas de los predios de Cobija.
- Evaluar la capacidad del modelo en la detección de vivienda de los predios seleccionados del Municipio de Cobija.

1.5 JUSTIFICACION

1.5.1 Económica

El Modelo Fotogramétrico Deep Learning desarrollado en el software ArcGis Pro, y la precisión de drones en el levantamiento de imágenes fotogramétricas, tendrá una eficacia en la presentación de metadatos y dando resultados inmediatos reduciendo tiempo y costos.

1.5.2 Social

Con el modelo Fotogramétrico Deep Learning y la aplicabilidad de drones en el levantamiento de imágenes fotogramétricas de precisión se beneficiarán las entidades públicas que van enfocadas al estudio de territorio y levantamiento de tierras urbanas y rurales como Catastro y la I.G.M. (Instituto Geográfico Militar).

1.5.3 Científica

El empleo de inteligencia artificial en modelos fotogramétricos con drones crea nuevos conocimientos científicos en la resolución de problemas tecnológicos y también sirve de base para futuras investigaciones.

1.6 ALCANCES Y LIMITES

A continuación, se presentan los alcances y los límites de la presente investigación.

1.6.1 Alcances

- Se estableció a seguir la metodología de Fotogrametría planteada con Drones para obtener resultados favorables
- Se georreferenció las imágenes (ortomosaico) en el software Pix4D y ArcGis PRO.
- Se implementó las funcionalidades de Deep Learning (aprendizaje supervisado) enfocados a sistemas de información geográfica en el software ArcGIS PRO.
- Se comprobó los resultados al concluir el proyecto de investigación para corroborar la Hipótesis.

1.6.2 Limites

La presente investigación solo pretende enfocarse en las nuevas tecnologías dentro del contexto de Sistemas de Información Geográfica, drones y su inclusión de Deep Learning.

- No se contempla en áreas dispersas rurales, donde abarca densidad de vegetación.
- No se contempla en otros departamentos dentro de Bolivia, considerando los factores de varianza de altura en el proceso de fotogrametría con drones en determinados lugares.
- El modelo solo se empleará en la Zona Central del Municipio de Cobija (distrito uno y dos).

CAPITULO II

2. MARCO TEÓRICO

2.1 DEEP LEARNING

Deep Learning (Aprendizaje Profundo), se trata de técnicas creadas ya de los años 80, cuando varias universidades y laboratorios de investigación se propusieron aplicar modelos estadísticos para encontrar patrones dentro de objetos donde reina una cierta aleatoriedad, y ser capaces de diferenciarlos automáticamente (IBERDROLA, 2017).

Este modelo imita la conectividad del cerebro humano, clasificando conjuntos de datos y encontrando correlaciones entre ellos. Basándose en las redes neuronales, el Deep Learning tiene la capacidad de jerarquizar el aprendizaje, procesando la información por niveles. Así, su aprendizaje sucede por etapas. (Grapsas, 2019).

Deep Learning es el estudio que permite a la máquina aprender por sí sola de los errores y de la información que recibe. Su algoritmo es capaz de identificar patrones y anomalías para crear un modelo ideal y prevenir futuros errores. De esta manera, se ajustan los parámetros y la máquina es capaz de obtener directamente un buen resultado. (Alcalá, 2019)

Deep Learning busca poder formalizar ideas, y crear un marco sobre el cual poder crear estructuras y modelos que permitan automatizar dichas tareas que realiza nuestro cerebro.

2.1.1 Aplicaciones prácticas de Deep Learning

Varios sitios y blogs de tecnología muestran un creciente interés por la tecnología Deep Learning y su avance en el entorno tecnológico (ilifebelt, 2018), la aplicación de dicha tecnología que llevan a cabo distintas aplicaciones, estos son:

- **Colorización automática de imágenes en blanco y negro:** es una tarea tradicionalmente hecha con esfuerzo humano, porque se precisa un criterio visual muy alto, sin embargo, Deep learning contiene la capacidad de identificar objetos y su contexto dentro de una fotografía en blanco y negro, para colorearla.
- **Un sistema que agrega audio a películas sin sonido:** se trata de que un modelo de Deep learning asocia los cuadros de video con una base de datos pregrabada en el proceso para seleccionar un sonido que coincida con lo que detecta visualmente.

- **Traducción automática:** esta es una tarea donde algunas palabras dadas, frases u oraciones de un idioma, se traducen automáticamente a otro, Deep learning está alcanzando resultados óptimos en traducción automática de texto y traducción automática de imágenes.
- **Clasificación de objetos y detección en fotografías:** se han alcanzado impresionantes resultados usando extensas redes neuronales, se contiene una gran variación de detección de objetos implica la detección de varios objetos en la misma escena, este sistema puede a su vez detectar un objeto y proporcionar información extra de sus características.

2.1.2 Red Neuronal Convolutiva

Las redes neuronales convolucionales son similares a las redes neuronales multicanal, su principal ventaja es que cada parte de la red se le entrena para realizar una tarea, esto reduce significativamente el número de capas ocultas, por lo que el entrenamiento es más rápido. Además, presenta invarianza a la traslación de los patrones a identificar. (Calvo, 2017)

Las redes neuronales convolucionales son muy potentes para todo lo que tiene que ver con el análisis de imágenes, debido a que son capaces de detectar características simples como por ejemplo detención de bordes, líneas, etc. y componer en características más complejas hasta detectar lo que se busca. (Calvo, 2017)



Figura 1: Red Neuronal Convolutiva
Fuente: (Cabello, 2018)

Las Redes Neuronales Convolucionales reducen drásticamente la cantidad de parámetros que deben ajustarse, las CNN manejan de manera eficiente la alta dimensionalidad de las imágenes en bruto.

2.1.3 Arquitectura Red Neuronal Convocucional

Una red neuronal convolucional es una red multicapa que consta de capas convolucionales y de reducción alternadas, y al finalmente tiene capas de conexión total como una red perceptrón multicapa. (Clavijo, 2017)

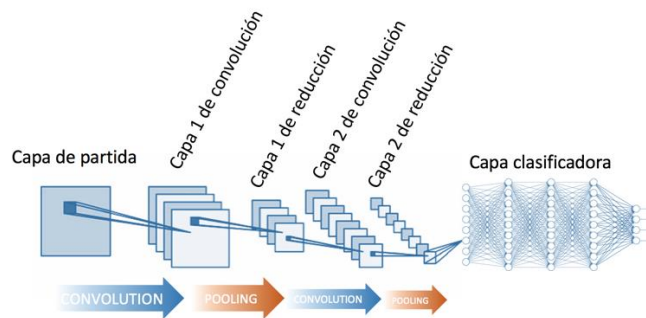


Figura 2: Arquitectura de una red Neuronal Convocucional
Fuente: (Calvo, 2017)

- **Convolución**

En la convolución se realizan operaciones de productos y sumas entre la capa de partida y los n filtros (o kernel) que genera un mapa de características. Las características extraídas corresponden a cada posible ubicación del filtro en la imagen original.

La ventaja es que el mismo filtro (igual neurona) sirve para extraer la misma característica en cualquier parte de la entrada, con esto que consigue reducir el número de conexiones y el número de parámetros a entrenar en comparación con una red multicapa de conexión total.

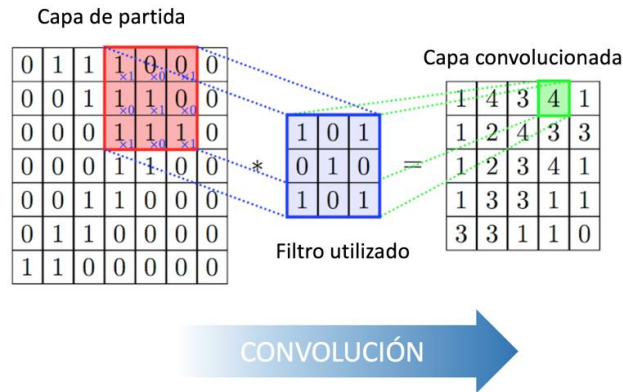


Figura 3: Capa de Partida Convolución
Fuente: (Calvo, 2017)

- **Reducción**

En la reducción se disminuye la cantidad de parámetros al quedarse con las características más comunes.

La forma de reducir parámetros se realiza mediante la extracción de estadísticas como el promedio o el máximo de una región fija del mapa de características, al reducir características el método pierde precisión, aunque mejora su compatibilidad. (Clavijo, 2017)

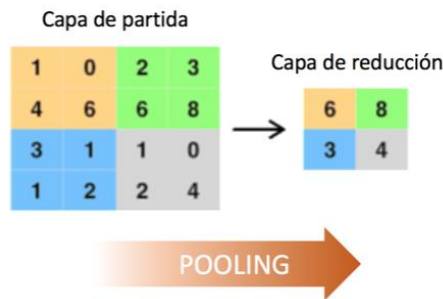


Figura 4: Capa de reducción
Fuente: (Calvo, 2017)

- **Clasificador red perceptrón multicapa**

El final de las capas convolucional y de reducción, se suele utilizar capas completamente conectadas en la que cada pixel se considera como una neurona separada al igual que en un perceptrón multicapa.

La última capa de esta red es una capa clasificadora que tendrá tantas neuronas como el número de clases a predecir.

2.1.4 Aprendizaje Supervisado

El aprendizaje supervisado proporciona una ruta directa para convertir datos en información real y procesable. Al utilizar los datos como un recurso, les permite a las organizaciones comprender y prevenir los resultados no deseados o impulsar los resultados deseados para lo que sea que estén tratando de predecir. Ha requerido gran cantidad de tiempo y la experiencia técnica de equipos especializados con el fin de construir, escalar y desplegar modelos predictivos precisos. Además, dado que los modelos de aprendizaje supervisado hacen predicciones del mundo real basado en los datos del pasado, los modelos deben ser reconstruidos periódicamente con el fin de mantener sus predicciones sin que se conviertan en obsoletas ya que en ocasiones el comportamiento de los datos puede cambiar. (Comment, 2018)

2.1.5 TensorFlow

TensorFlow es una biblioteca de software de código abierto basada en Python desarrollada por Google. Tiene una gran popularidad en el aprendizaje automático y el área de aprendizaje profundo debido a su simplicidad, flexibilidad y compatibilidad. TensorFlow está direccionado a las redes neuronales recurrentes y redes neuronales convolucionales. (James, 2019)

TensorFlow es una biblioteca de software de código abierto para el cálculo numérico de alto rendimiento. Su arquitectura flexible permite una fácil implementación de computación en una variedad de plataformas (CPU, GPU) y desde escritorios hasta clústeres de servidores y dispositivos móviles y periféricos. Desarrollado originalmente por investigadores e ingenieros del equipo Google Brain dentro de la organización AI de Google, cuenta con un sólido respaldo para el aprendizaje automático y el aprendizaje en profundidad, y el núcleo de computación numérica flexible se utiliza en muchos otros dominios científicos. El sistema es flexible y puede utilizarse para expresar una amplia variedad de algoritmos, incluidos algoritmos de entrenamiento e inferencia para modelos de redes neuronales profundas, se utilizó para realizar investigaciones y para implementar sistemas de aprendizaje automático en el área de la ciencia de la computación y otros campos, incluidos el reconocimiento de voz, visión artificial robótica, recuperación de información,

procesamiento del lenguaje natural, extracción de información geográfica y descubrimiento de fármacos computacionales (TensorFlow, 2019)

2.1.6 Python

Python es un lenguaje de programación poderoso y fácil de aprender. Cuenta con estructuras de datos eficientes y de alto nivel y un enfoque simple pero efectivo a la programación orientada a objetos. La elegante sintaxis de Python y su tipado dinámico, junto con su naturaleza interpretada, hacen de éste un lenguaje ideal para scripting y desarrollo rápido de aplicaciones en diversas áreas y sobre la mayoría de las plataformas. El intérprete de Python y la extensa biblioteca estándar están a libre disposición en forma binaria y de código fuente para las principales plataformas, y puede distribuirse libremente. En el sitio de python contiene también distribuciones y enlaces de muchos módulos libres de Python de terceros, programas y herramientas, y documentación adicional. El intérprete de Python puede extenderse fácilmente con nuevas funcionalidades y tipos de datos implementados en C o C++ (u otros lenguajes accesibles desde C). Python también puede usarse como un lenguaje de extensiones para aplicaciones personalizables (Rossum, 2017)

2.1.7 Clasificación de Píxeles en Sistemas de información Geográfica

En una implementación de análisis de ráster, esta herramienta ejecuta un modelo de aprendizaje profundo entrenado en una imagen de entrada para generar un ráster clasificado publicado como capa de imágenes alojadas en el portal.

2.1.8 Detectar Objetos en Sistemas de información Geográfica

En una implementación de análisis de ráster, esta herramienta ejecuta un modelo de aprendizaje profundo entrenado en un ráster de entrada para generar una clase de entidad que contiene los objetos que identifica. La clase de entidad se puede compartir como capa de entidades alojada en el portal. Las entidades pueden ser cuadros o polígonos de delimitación alrededor de los objetos encontrados o puntos en el centro de los objetos.

2.1.9 Rasters

En su forma más simple, un ráster consta de una matriz de celdas (o píxeles) organizadas en filas y columnas (o una cuadrícula) en la que cada celda contiene un valor

que representa información, como la temperatura. Los rásteres son fotografías aéreas digitales, imágenes de satélite, imágenes digitales o incluso mapas escaneados. (ArcGis, 2019)

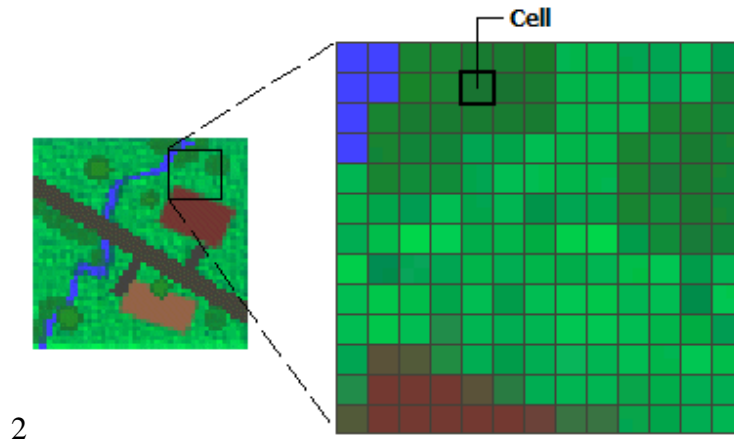


Figura 5: Ráster en una Imagen
Fuente: (ArcGis, 2019)

Dataset Rasters

Un dataset ráster define cómo se almacenan los píxeles, por ejemplo, el número de filas y columnas, la cantidad de bandas, los valores de píxeles reales y otros parámetros específicos de los formatos de ráster.

Metadatos

Un metadato constituye la información estructurada y organizada de un conjunto de datos que permite consultar, evaluar, comparar, acceder, y/o utilizar la información, describiendo su autor, semántica, calidad, modo de identificación, restricciones de uso, mantenimiento, distribución, sistema de referencia, contenido entre otros. La generación de metadatos geográficos describe la información geográfica de tal forma que se pueda obtener un inventario tanto de los datos presentes en un producto geográfico, como de los productos geográficos existentes.

ISO 19115-2

Una importante fuente de información utilizada en los Sistemas de Información Geográfica son las imágenes de satélite, orto-fotografías, modelos digitales, etc. Estos datos

de tipo ráster y malla poseen características intrínsecas particulares, que deben quedar reflejadas mediante los metadatos. (Geoportal, 2018)

A pesar de que la norma ISO 19115 contiene algunos elementos destinados a documentar este tipo de información, éstos resultan escasos para documentar de forma correcta y completa la información ráster. En consecuencia y para complementar esta norma, se está elaborando: La norma ISO 19115- 2 “Información geográfica metadatos para imágenes y datos de cuadrícula”, que se trata una extensión de la norma ISO 19115 para datos ráster e imágenes. (Geoportal, 2018)

2.2 SISTEMAS DE INFORMACIÓN GEOGRÁFICA (SIG)

Es una herramienta integradora que pretende abarcar todas las funcionalidades para trabajar con variables y elementos localizados. Gracias a la tecnología, la información recopilada puede ser aprovechada de mejor manera, para posteriormente someterla a un análisis minucioso; básicamente, un SIG permite realizar distintas operaciones, como ser la lectura, edición, almacenamiento y gestión de datos espaciales; análisis de dichos datos para incluir consultas sencillas y elaboración de modelos complejos; finalmente, generar resultados expresados en mapas, informes, gráficos, etc., siendo esta una de las operaciones más importantes y fundamentales (Olaya, 2014).

Este sistema de información diseñado para trabajar con datos referenciados mediante coordenadas espaciales o geográficas obtiene información georreferenciada mediante un conjunto de operaciones, este proceso es considerado un elemento integrador que engloba a un conjunto de componentes interrelacionados.

2.2.1 Componentes de los SIG

Se estructura con 5 elementos principales: hardware, software, datos, cartografía, metodología y personas (usuarios y operadores) (Gutierrez Palacios, 2016).

Hardware o componente físico, compuesto por una computadora y una serie de periféricos de entrada y salida (mesas digitalizadoras, escáneres, teclado, etc.). También se considera las unidades de almacenamiento.

Software, con el que se opera y manipula los datos. Se caracteriza por la facilidad de acceso, capacidad de almacenamiento, procesamiento, análisis y evaluación del SIG.

Datos, se constituyen en la base de cualquier sistema, sin ellos, el hardware y el software no tienen razón de ser. Por la dificultad de recolección de alguna información específica, se considera como el elemento más costoso del proyecto porque requiere actualización constante.

Cartografía, muestra el espacio geográfico donde se muestran, de forma georreferenciada todos los datos obtenidos.

Metodología y personas, la primera está sujeta a un plan de trabajo diseñado en función de los objetivos planteados al iniciar un proyecto. Por su parte, las personas son quienes echan a andar el SIG, siendo profesionales entendidos en la materia; a estos se denominan especializados, puesto que tienen el conocimiento óptimo para llevar adelante un proyecto completo.

2.2.2 Fotogrametría

La fotogrametría es la técnica (o colección de técnicas) que permiten hacer reconstrucciones de terrenos y medidas a partir de imágenes aéreas. Este es un proceso muy complejo en el que el objetivo principal es convertir datos bidimensionales (imágenes planas) en información cartográfica tridimensional; los avances tecnológicos han permitido desarrollar de manera óptima la recolección de información con aviones, cámaras digitales e incluso satélites artificiales.

Formalmente, la Sociedad Americana de Fotogrametría y Sensores Remotos (ASPRS), define la fotogrametría como el arte, ciencia y tecnología para obtener información confiable sobre objetos físicos y el ambiente, a través de procesos de grabación, medición e interpretación de imágenes y patrones de energía electromagnética radiada u otras emisiones. (ASPRS, s.f.)

En la actualidad, el uso de drones ha potenciado el uso de la fotogrametría aérea en muchos campos a diferencia de los que anteriormente estaba disponible. Las técnicas actuales se basan en las tradicionales, sin embargo, requieren un equipamiento específico, en

contrapartida, la complejidad de las operaciones se ha reducido sustancialmente (Aerials Insights, 2018).

2.2.3 Proceso Fotogramétrico

La fotogrametría abarca métodos de medición e interpretación de imágenes a fin de obtener la forma y ubicación de un objeto a partir de una o más fotografías de ese objeto. En principio, los métodos fotogramétricos se pueden aplicar en cualquier situación en la que el objeto a medir puede ser fotográficamente grabado. El propósito principal de una medición fotogramétrica es la reconstrucción tridimensional de un objeto en forma digital (coordenadas y elementos geométricos derivados) o forma gráfica (imágenes, dibujos, mapas). La fotografía o imagen representa un almacén de información que se puede volver a acceder en cualquier momento. (T Luhmann, 2016)

2.2.4 Adquisición de Imágenes

Existen varios parámetros a definir antes de realizar un levantamiento fotogramétrico, como la resolución espacial, la altura y velocidad de vuelo del sistema UAS, la frecuencia de captura, entre otros. Entender la relación entre estos conceptos permite obtener una base de imágenes sólida para las siguientes etapas de medición y reconstrucción. (Quiros, Procesamiento de imágenes y fotogrametría con VANTs, 2019).

Resolución Espacial

La resolución espacial es la distancia real representada entre dos centros de píxel, su unidad es m/pix (metro por píxel) y suponiendo que el píxel es cuadrado, sería equivalente al ancho o largo de cada uno, también es conocida como Distancia de Muestra de Tierra por sus siglas en inglés GSD (Ground Sample Distance). La resolución espacial no es constante en toda la imagen debido a irregularidades de pendiente en el terreno u objeto de estudio por lo que se recomienda usar una GSD promedio (Arriola, 2018).

Altura de vuelo

La altura de vuelo depende directamente de la longitud focal, las dimensiones del sensor y la resolución espacial.

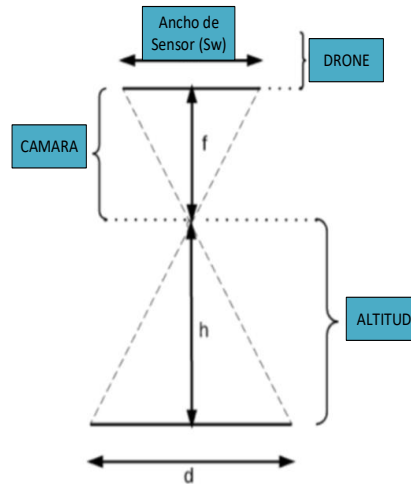


Figura 6: Factores de vuelo
Fuente: (Arriola, 2018)

Su descripción matemática se representa en la siguiente ecuación

La altura de vuelo h se define como

$$h = f \cdot d / Sw$$

$$d = GSD \cdot Pw$$

$$h = f \cdot GSD \cdot Pw / Sw$$

h = altura de vuelo [m]

f = distancia focal [mm]

d = ancho de área real cubierta [m]

Sw = ancho de sensor [mm]

GSD = resolución espacial [m/pixel]

Pw = ancho de cada imagen [pixel]

Obturación de Cámara

También llamada velocidad de disparo, y es el tiempo que el obturador deja pasar la luz al sensor. La correcta regulación de este parámetro permite evitar imágenes sobre expuestas, su estimación depende directamente de la resolución espacial y de la velocidad del sistema del Dron (Arriola, 2018)

$$OC = GSD / 2S$$

OC: obturación de cámara [s]

GSD: resolución espacial [m/pixel]

S: velocidad de vuelo [m/s]

2.2.5 Modelo Fotogramétrico Deep Learning.

El modelo fotogramétrico Deep Learning es el conjunto de imágenes aéreas procesadas en diferentes dimensiones 2D o 3D estableciendo la identificación de características y el reconocimiento de patrones descritos en el modelo.

2.3 DRONE

Este término se ha utilizado para referirse a estos vehículos no tripulados, en resumen, “hay otras denominaciones para definir un dron, que pueden ayudarnos a identificar rápidamente en qué consisten. Algunos reciben el nombre de cuadricópteros, sexacópteros u octacópteros, dependiendo de si tiene 4, 6 u 8 hélices. Por tanto, podemos definir un dron como un vehículo que se controla por control remoto, que para elevarse utiliza la fuerza de giro de sus motores unidos a las hélices” (Equipo de Expertos Universidad Internacional de Valencia, 2018).

Su funcionamiento es prácticamente el mismo que el de un avión, con la diferencia de que se controla remotamente, y obviamente, requiere de la habilidad de un piloto; sin embargo, también existen modelos guiados por GPS, estos pueden ser programados para realizar recorridos largos y tienen la misión principal de realizar vuelos de observación.

2.3.1 Usos de los drones

Su uso, en un principio, fue netamente militar, esto para lograr ventaja en el campo de batalla, con el paso del tiempo y la constante mejora de la tecnología, estos aparatos han encontrado un uso civil, aplicado a las áreas del entretenimiento, transporte, agricultura, topografía, inspección, etc.

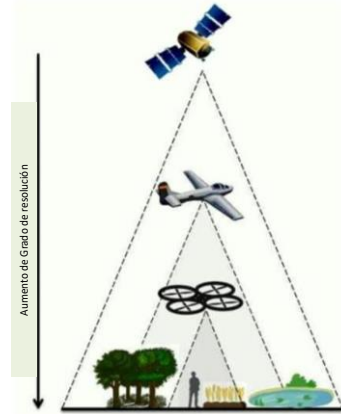


Figura 7: Aumento de Grado de Resolución
Fuente: (Fallas, 2018)

Sin lugar a duda una de las ramas de la ciencia que más se ha beneficiado de los drones es la cartografía, ya que con las cámaras precisas y diferentes tipos de sensores son capaces de hacer mapeos muy detallados en poco tiempo y a bajo costo, pudiendo incluso generar modelos en 3D. Para este cometido, los drones usan los principios de la fotografía aérea, según el tamaño del área a cubrir y el detalle con que se necesite representarla. Después se calcula el número de fotos para crear un mosaico y la forma que debe tener el vuelo para cubrir esa área (Velázquez Olivera, 2017).

2.3.2 Geomagnetismo

El geomagnetismo es la ciencia que se encarga de estudiar el origen, las propiedades y las variaciones de este campo magnético terrestre. Los polos magnéticos son los que se encuentran en puntos en los que el eje del dipolo está en intersección con la superficie terrestre (meteorologíaenred)

El geomagnetismo se mide en kilopondio (kp), el cual tiende a afectar a dispositivos que tengan sincronización con satélites.

2.4 ESTADO DEL ARTE

2.4.1 Investigaciones Internacionales

La Investigación realizada el año 2017 por R. Vizcaya Cárdenas¹, J. M. Flores Albino, V. M. Landassuri Moreno y S. Lazcano Salas denominada “*Desempeño de una Red Neuronal Convolutiva para Clasificación de Señales de Tránsito Vehicular*”, la presente

investigación tiene como objetivo ver el desempeño de la Red Neuronal Convolutiva haciendo varios experimentos, que consisten en entrenar una red durante diferentes cantidades de épocas, con el propósito de medir el tiempo de entrenamiento y exactitud. El error en la clasificación más bajo que se obtuvo fue de 8.87%, cuando el tamaño de lote fue de 32 y el experimento de 1400 épocas, lo que significa que de cada 100 imágenes de tránsito vehicular que se le presenten a la CNN, esperamos que aproximadamente 91 de ellas las clasifique correctamente. (R. Vizcaya Cárdenas¹, 2017)

La Investigación realizada el año 2018 por Miguel Ángel Chiccon Apaza denominada “*Fusión de Datos para Segmentación Semántica en Aplicaciones Urbanas de Teledetección Área usando Algoritmos de Aprendizaje Profundo*” esta investigación sobre diferentes enfoques de fusión de datos utilizando aprendizaje profundo aplicando en el área de la teledetección urbana. Dos enfoques han sido investigados: apilamiento de canales de entrada y fusión tardía con corrección residual. Ambos enfoques se implementaron tomando como base la arquitectura U-net. Se logró el entrenamiento de extremo a extremo y basados en transferencia de aprendizaje. Se obtuvieron resultados aceptables. Estos resultados reflejan la necesidad de entrenar con un tamaño de batch mayor para poder mejorar los resultados lo cual involucra la utilización de hardware de alta prestaciones. Otra opción de mejora es la utilización de pesos de redes pre-entrenadas de mayor complejidad como Resnet-50 o Inception. (Apaza, 2018)

La Investigación realizada el año 2018 por Edison Fabian Enríquez Jaramillo denominada “*Deep Learning para la Detección de Peatones y Vehículos sobre FPGA*” en esta investigación fue entrenar y probar una red neuronal convolutiva, con la intención de valorar los tiempos de entrenamiento y el error en la clasificación de imágenes de señales de tránsito vehicular empleadas en México y otros países. La CNN que se implementó es pequeña comparada con las reportadas en el estado del arte en este tipo de aplicaciones, esto fue así para hacer el entrenamiento en un CPU, y analizar su implementación en un FPGA. De los resultados obtenidos se observa que para el experimento en el que se cuenta con una mayor cantidad de información (banco de 12,000 imágenes), el entrenamiento converge en menor cantidad de épocas que en el caso que se tiene menor cantidad de información (8,000 imágenes), esto para todos los casos de experimentación. La exactitud en la clasificación es

mejor en el caso que se tiene mayor cantidad de información, 96.75% contra 91.13 %. En los dos casos, el desempeño mejora con forme se reduce el tamaño del lote. Como es de esperarse, el tiempo de entrenamiento al incrementar la cantidad de información.

2.4.2 Investigación Nacional

A nivel Nacional, Bolivia no existe alguna investigación en el campo de Deep Learning en imágenes fotogramétricas para la detección de vivienda de los predios.

CAPITULO III

3. MARCO METODOLÓGICO

3.1 ENFOQUE DE LA INVESTIGACIÓN

La investigación científica se concibe como un conjunto de procesos sistemáticos y empíricos que se aplican al estudio de un fenómeno; es dinámica, cambiante y evolutiva. Se puede manifestar de tres formas: cuantitativa, cualitativa y mixta (Hernández Sampieri et al, 2010).

La investigación aplicada busca la generación de conocimiento con aplicación directa a los problemas de la sociedad o el sector productivo, se basa en el avance teórico y hallazgos tecnológicos de la investigación básica para generar conocimiento que pueda ser utilizado directamente o a mediano plazo en el mejoramiento (Roberto Sampieri Lucio, 2014).

El diseño experimental se caracteriza por manipular una o varias variables independientes, recolectando datos y observar el comportamiento de la variable dependiente y determina conclusiones. (Tomala, 2015)

En la presente investigación se empleó el enfoque cuantitativo y tipo de investigación Aplicativa y diseño Experimental, porque se determinó las bases técnicas y características del tema de investigación.

3.1.2 Diseño de la Metodología

En la figura 8 se detalla el proceso que se utilizó para realizar la investigación con sus diferentes metodologías iniciando el levantamiento fotogramétrico, inclusión Deep Learning y finalmente los resultados de la evaluación.

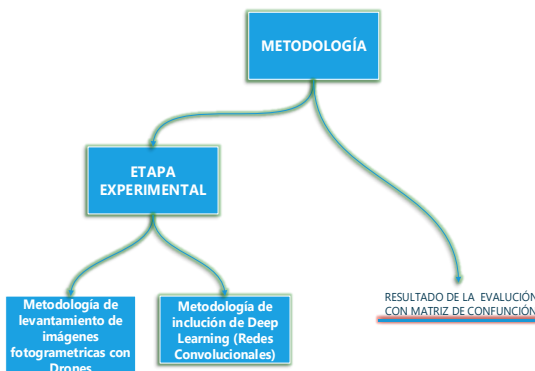


Figura 8: Diseño de la Metodología

Fuente: Elaboración Propia

3.2 DESGLOSE DE DISEÑO DE LA METODOLOGÍAS

Esta Metodología se desglosa en diferentes técnicas y métodos para realizar su cometido de desarrollar el Levantamiento Fotogramétrico con Drones y tener su visión de la inclusión de Deep Learning.

3.2.1 METODOLOGÍA DE LEVANTAMIENTO FOTOGAMÉTRICO CON DRONES

La metodología a seguir se basa en la tesis: “Diseño de un sistema de cuantificación autónoma de biomasa basado en procesamiento de imágenes y fotogrametría con vehículos aéreos no tripulado” (Cordero-Quirós, 2018) , esta metodología está enfocado al área urbana Zona Central del Municipio de Cobija. Considerando los factores que presenta hacer el levantamiento fotogramétrico se añade más etapas.

La Metodología está estructurada en tres etapas las cuales conllevan también actividades que se deben desarrollar para cumplir cada etapa y obtener imágenes fotogramétricas precisas.

- Análisis y Estudio de la región a aplicar la investigación.
- Captura de imágenes.
- Procesamiento de Imágenes.

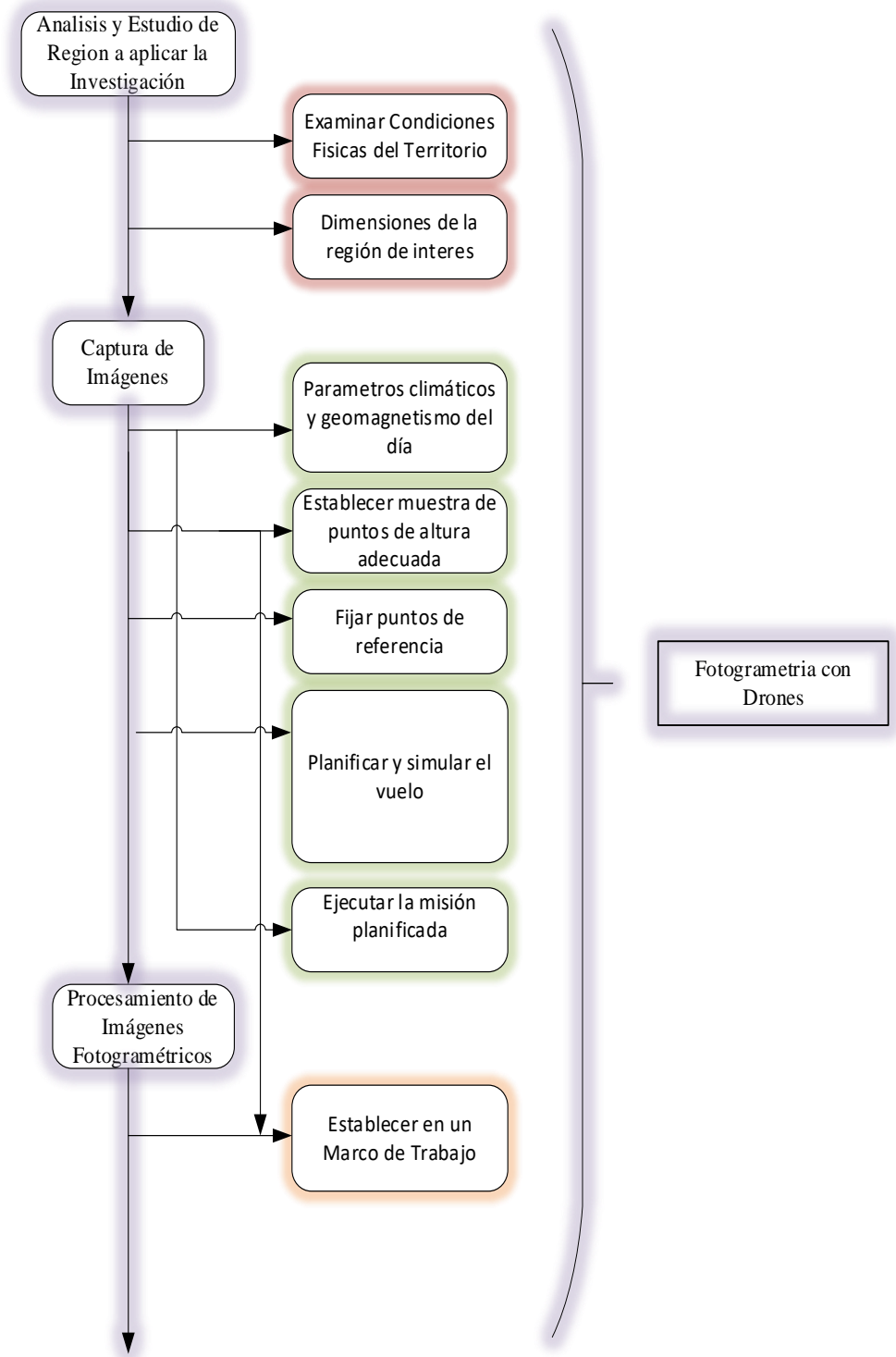


Figura 9: Desglose de la Metodología de Fotogrametría con Drones
Fuente: Elaboración Propia

3.2.1.1 Análisis y Estudio de la región a aplicar la investigación

Esta etapa inicial está conformada por dos actividades las cuales son esenciales para iniciar el proceso fotogramétrico con drones. Mediante experiencia de vuelo con drones se presentan las dos actividades principales de esta etapa las cuales son:

- **Examinar condiciones físicas del territorio:**

Se debe inspeccionar el territorio donde se va desarrollar el levantamiento fotogramétrico considerando la densidad de vegetación y construcciones que se contempla dentro del área de estudio, lo cual se considera área urbana, también nos permitirá determinar zonas de riesgo para el drone en caso de emergencias.

- **Dimensiones de la región de interés:**

Mediante herramienta propia del drone (Flysafe geo-map DJI) o visualizar mediante herramienta Google Maps o Google Earth, delimitamos una aproximación superficial de nuestro área o territorio a abarcar el marco de trabajo donde se desarrollará el vuelo planificado.

Teniendo en cuenta las nuevas normas de vuelo establecidas por cada país se estableció las zonas geo las cuales debemos tener en cuenta al abarcar en el área de interés si nos afecta, se debe hacer las respectivas solicitudes acorde a cada empresa de la marca de los drones, exige en la línea de la marca DJI las Zonas GEO en: aeropuertos, edificios del gobierno, recintos Militares, Institutos de Investigación y recintos penitenciarios.

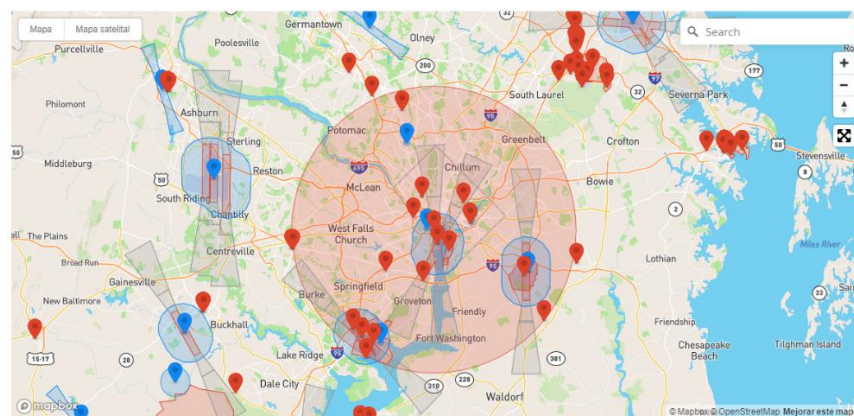
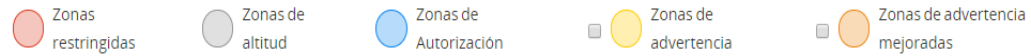
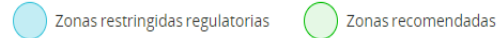


Figura 10: Restricción Zonas Geo de DJI
Fuente: DJI Fly

Zonas DJI GEO



Otros











-  Zonas restringidas En estas zonas, que aparecen en rojo en la aplicación DJI GO, los usuarios recibirán una advertencia y se evitará el vuelo. Si cree que tiene la autorización para operar en una Zona restringida, comuníquese con flysafe@dji.com o Desbloqueo en línea.
-  Zonas de altitud: las zonas de altitud aparecerán en gris en el mapa. Los usuarios reciben advertencias en DJI GO o DJI GO 4 y la altitud de vuelo es limitada.
-  Zonas de Autorización. En estas zonas, que aparecen en azul en el mapa DJI GO, los usuarios recibirán una advertencia y el vuelo está limitado de forma predeterminada. Los usuarios autorizados pueden desbloquear las zonas de autorización con una cuenta verificada por DJI.
-  Zonas de advertencia. En estas zonas, que pueden no aparecer necesariamente en el mapa DJI GO, los usuarios recibirán un mensaje de advertencia.
Ejemplo de zona de advertencia: espacio aéreo de clase E
-  Zonas de advertencia mejoradas. En estas zonas, GEO le pedirá al momento del vuelo que desbloquee la zona siguiendo los mismos pasos que en una zona de autorización, pero no necesita una cuenta verificada o una conexión a Internet en el momento de su vuelo.
-  Área densamente poblada: esta área se muestra en rojo en el mapa. En circunstancias normales, la población de esta área está más concentrada, así que no vuele sobre esta área.
(Ejemplo: bloque comercial)
-  Zonas restringidas regulatorias: debido a las regulaciones y políticas locales, los vuelos están prohibidos dentro del alcance de algunas áreas especiales.
(Ejemplo: prisión)
-  Vuelo recomendado: esta área se muestra en verde en el mapa. Se recomienda que elija estas áreas para los arreglos de vuelo.

Figura 11: Significado de las Zonas GEO por Colores

Fuente: DJI Fly

En otras líneas de drones viene incluido la base de datos predeterminado de las Zonas Geo en este caso DJI los clasifica por colores, visualizar la siguiente Figura 11.

En la tabla 3 se llenará acorde su respectiva asignación los datos recolectados mediante la herramienta de zonas geo de DJI 2.0, y denotar zonas de interés restringidos o libres con el dron e a trabajar.






Dimensiones de la región de interés				
Herramienta		Drone (RPA)		
Enlace Web de la Herramienta				
País		Departamento		Ciudad
Imagen de región de interés				
Nivel de Restricciones				
 Zonas restringidas				
 Zonas de Autorización				
 Zonas de altitud				
 Zonas de advertencia				
 Zonas de advertencia mejoradas				
Observaciones				

Tabla 3: Dimensiones de la región de interés
Fuente: Elaboración Propia

3.3.1.2 Captura de Imágenes

En esta segunda etapa está conformada por cinco actividades las cuales nos permite priorizar un vuelo más seguro y eficaz acorde de las exigencias que se tenga que cubrir el drone sin percances.

- **Parámetros climáticos y geomagnetismo del día**

Para todo vuelo mucho depende de los parámetros climáticos porque cada drone en especial tiene su límite de resistencia a ráfagas de viento.

El geomagnetismo es el campo magnético terrestre cuya variación espacial y temporal afectan al GPS del drone, los rangos permitidos se muestran en la tabla 4

Rango de Vuelo Permitido	0-4	kp(kilopondio)
Rango de Vuelo No Permitido	5-9	kp(kilopondio)

Tabla 4. Rango de Vuelo Permitido con el drone
Fuente: Elaboración Propia

La aplicación desarrollada tanto en dispositivos Android e iOS ayudan a determinar el día de vuelo permitido y tener una mejor planificación de levantamiento fotogramétrico, estas aplicaciones que nos ayudan con esta actividad son:

- App Uavforecast para Android e IOS
- App Magnetología para Android e IOS

Otro factor esencial a tener en cuenta es la cantidad de satélites que debe captar el drone lo recomendable es como mínimo 10 satélites si está por debajo de lo mencionado mejor no hacer volar, el drone no tendrá la precisión que requerimos y nos dará meta datos erróneos. En la Figura 12 se muestra como los satélites conectan con el drone y considerando que tengamos la tecnología de drone con RTK¹ se conecta a través de una estación terrestre como se muestra en la figura 13.

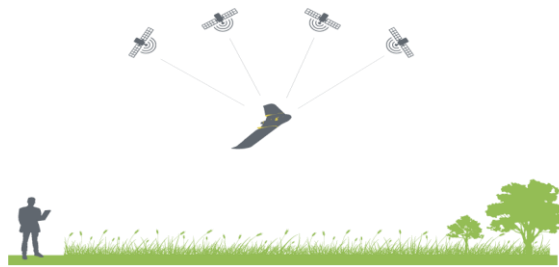


Figura 12: Conexión de Satélites con el Dron

Fuente: (geodrone, 2019)

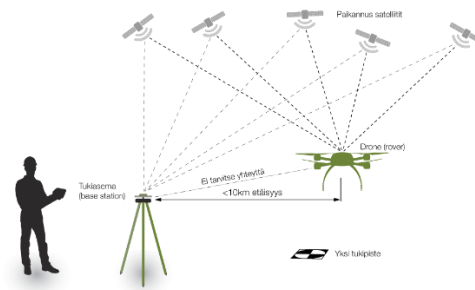


Figura 13: Dron con Tecnología RTK

Fuente: (geodrone, 2019)

¹ RTK: **RTK** (del inglés *Real Time Kinematic*) o navegación cinética satelital en tiempo real, es una técnica usada para la topografía

La tabla 5 se llenará acorde su respectiva asignación los datos recolectados mediante software que se da a mencionar para el geomagnetismo, ráfagas de viento y satélites disponible, acorde al drone que se empleará.

Parámetros climáticos y geomagnetismo del día			
Modelo de Drone			
Software			
Fecha	Ráfagas de viento Km/h	Geomagnetismo Kp(Kilopondio)	Satélites disponibles
Observaciones			
	Vuelos Permitidos		
	Vuelos No Permitidos		

Tabla 5: Parámetros climáticos y geomagnetismo
Fuente: Elaboración Propia

• **Establecer Muestra de Puntos de Altura Adecuada**

Al determinar los puntos de altura adecuada está establecido por el Proceso Fotogramétrico que están bajo fórmulas matemáticas donde determinaremos la altura de vuelo acorde funciones de GSD (Resolución Espacial), a continuación, presentamos las fórmulas las cuales debemos tomar en cuenta en el muestreo de puntos para hallar la altura adecuada visualizar la siguiente tabla 6.

Nombre de la Formula	Formula	Donde
Altura	$h = f \cdot d / Sw$ $d = GSD \cdot Pw$ $h = f \cdot GSD \cdot Pw / Sw$	h = altura de vuelo [m] f = distancia focal [mm] d = ancho de área real cubierta [m] Sw = ancho de sensor [mm] GSD = resolución espacial [m/pixel] Pw = ancho de cada imagen [pixel]

Tabla 6: Formula de Altura
Fuente: Elaboración Propia

En la siguiente tabla 7 se deberá llenar los muestreos juntamente con el dron para tener metadatos exactos, obtener la altura ideal con respecto a la resolución espacial y describir si hay observaciones de cada muestra.

Muestreo de Altura							
Drone:		Altura Ideal		m	GSD Ideal		m/px
Muestra (h)						N°	
Coordenadas		Imagen de Referencia				Altura	
$h = \frac{f * d}{S_w} =$	Altura de Vuelo			h		m	
	Distancia Focal (Acorde a formato)			f		mm	
	Ancho de Área real Cubierta			d		m	
	Ancho de sensor			S_w		mm	
	Resolución Espacial			GSD		m/px	
	Ancho de cada Imagen (Acorde a Formato)			P_w		px	
$d = GSD * P_w =$	Resolución Espacial					m/px	
	Ancho de cada Imagen (Acorde a Formato)					px	
Observaciones							

Tabla 7: Muestreo de Altura
Fuente: Elaboración Propia

• **Fijar Puntos de referencia**

Establecer los puntos de referencia nos ayudan a determinar una georreferencia más precisa para ello se puede considerar dos opciones, la primera opción es tomar muestras de tres puntos o más dentro del área donde se va realizar el vuelo planificado con un dispositivo externo GPS de buena precisión, la segunda opción es solicitar al ente mayor el Instituto Geográfico Militar (I.G.M.) puntos para la georreferenciación precisa y confiable.

En la tabla 8 se debe llenar con las coordenadas e imágenes de referencia proporcionadas.

Coordenadas	Imagen de Referencia

Tabla 8: Puntos de Referencia
Fuente: Elaboración Propia

- **Planificar y simular el vuelo**

Una vez que el drone este en lugar de despegue desde el software de planificación de vuelo se debe considerar los aspectos de revisión del hardware como: posicionamiento de hélices, calibración de la brújula, calibración del IMU y calibración de la cámara (sensor RGB); tomando en cuenta la tabla 9 donde se toma datos al iniciar el vuelo.







Verificación de vuelo					
Drone		Fecha		N° de Vuelo	
Estado del GPS	Estado del IMU	Satélites disponibles	Nivel de Geomagnetismo Kp	Viento Km/h	Ráfagas de VientoKm/h
 Excelente	 Excelente				
 Buena	 Buena				
 Mala	 Mala				

Tabla 9: Cuadro de datos al Iniciar el Vuelo Planificado

Fuente: Elaboración Propia

Otro factor es que a cada drone tiene una resistencia al viento en específico considerar la tabla 10.

Tipo de Drone	Marca	Modelo	Resistencia mínimo al viento	Resistencia máxima al viento
Multirotor	DJI	Phantom 4 Pro	29 Km/h	38Km/h
Multirotor	DJI	Mavic 2Pro	29 Km/h	38Km/h
Multirotor	DJI	Phantom 4 ProV2.0	29 Km/h	38Km/h
Multirotor	AutelRobotics	AutelEvo	20 Km/h	30 Km/h
Multirotor	Parrot	Anafi	30 Km/h	50 Km/h
Multirroto	Parrot	Bluegrass	30 Km/h	50 Km/h
Multirroto	Yuneec	H520	26 Km/h	40 Km/h

Tabla 10: Resistencia al Viento

Fuente: Elaboración Propia

Una vez establecido la planificación de vuelo el drone determinara la Cantidad de Baterías que va abarcar al realizar el levantamiento fotogramétrico para ello hay software que ayudan a realizar los vuelos planificados.

En la tabla 11 se mencionan aplicaciones que verifican la parte de hardware del dron al realizar el vuelo fotogramétrico.

Herramienta	Plataforma Android	Plataforma IOS	Plataforma Windows	Opera	Licencia de Pago	Licencia Libre
Pix4D	Si	Si	Si	DJI y Parrot	SI	No
DroneDeploy	Si	Si	Si	DJI y Parrot	Si	No
Map Pilot	No	Si	No	DJI	Si	No
Drone Harmony	Si	No	No	DJI	Si	No
Precision Fligth	Si	Si	No	DJI	No	Si

Tabla 11: Lista de App para Vuelo Planificado Elaboración Propia
Fuente: Elaboración Propia

Al establecer con que software se elige al realizar el vuelo planificado se debe considerar si todas las funciones son compatibles con el dron para no tener percances en pleno vuelo o ejecución del vuelo. Al realizar la simulación de vuelo planificado nos determina el tipo de barrido que va realizar y en qué puntos se va detener si se va requerir más batería.

La obturación de disparo también llamado velocidad de disparo nos ayuda a determinar el tiempo en que el obturador de la cámara deja pasar luz al sensor la cual este sujeto a la resolución espacial y la velocidad al cual este planificado el vuelo, en la siguiente tabla 12 se denota la fórmula que utilizaremos para determinar la obturación.

Nombre de la Fórmula	Fórmula	Donde
Obturación de la Cámara	$OC = GSD / 2S$	OC: obturación de cámara [s] GSD: resolución espacial [m/px] S: velocidad de vuelo [m/s]

Tabla 12: Formula de Obturación de Cámara
Fuente: Elaboración Propia

En la siguiente tabla 13 se debe llenar juntamente con cada muestreo una vez obtenido el GSD y conocer de cada vuelo planificado su vuelo.

Numero de Muestra				
$OC = GSD / 2S$	OC	obturación de cámara		[s]
	GSD	resolución espacial		[m/px]
	S	velocidad de vuelo		[m/s]

Tabla 13: *Obturación de disparo*

Fuente: *Elaboración Propia*

• Ejecutar la Misión Planificada

Al tener todos los parámetros ya establecidos y considerando los datos que debemos demarcar en la tabla 3 y 4, podemos ejecutar la misión planificada.

Al finalizar la misión planificada se recomienda tomar control del vuelo en caso que el dron no esté demarcando el punto inicial de partida, como medida de seguridad y no tener accidentes, ya que al trabajar en zona urbana se debe tener en cuenta con edificios, tendido de red eléctrica o personas que están transitando.

Al concluir la misión planificada se deberá tener en cuenta los resultados sincronizados como referencia de la misión de vuelo, en la siguiente tabla 14 se deberá llenar con los datos que proporciona el software de misiones planificadas.

Muestra N°	
Software	
Cantidad de Imágenes	
Tiempo de Vuelo	
Tipo de Vuelo	
Angulo de la Cámara	

Tabla 14: *Referencia de Misión de Vuelo*

Fuente: *Elaboración Propia*

3.3.1.3 Procesamiento de Imágenes

Ya una vez obtenido las imágenes ortomosaicos de la Misión Planificada se procesa en un software que se realice el proceso del modelado, en la siguiente tabla 15 se mencionan algún software enfocado al modelo fotogramétrico con drones.

Herramienta	Windows	Mac	Linux	En la Nube	Licencia de Pago	Licencia Libre
Pix4D	Si	Si	No	Si	Si	no
Drone2Map	Si	Si	No		Si	No
DroneDeploy	Si	Si	No	Si	Si	No
Maps Made Easy	No	Si	No	Si	Si	No
Hexagon Geospatial	No	No	No	Si	Si	No
Agisoft Photoscan	Si	Si	No	No	Si	No
OpenDroneMap	Si	No	Si	No	No	Si

Tabla 15: Software de Procesamiento de Imágenes (ortomosaico)

Fuente: Elaboración Propia

Al procesar las imágenes (ortomosaico) requiere de un buen equipo para que renderice las imágenes las cuales se recomienda trabajar para mayor eficiencia de un equipo computacional el cual se detalla en la siguiente tabla 16

Procesador	i7 o i9	Ryzen 5 o 7
RAM	32 GB	
Tarjeta de Video	GTX 1080	Radeon R9 Fury

Tabla 16: Equipo Computacional Requerido

Fuente: Elaboración Propia

• Establecer en un Marco de Trabajo

Para establecer el marco de trabajo se requiere de herramientas de análisis se les llama Sistemas de Información Geográfica (SIG) y son capaces de gestionar y analizar información geográfica y con referencia espacial. En la tabla 17 se detallan las herramientas más sobresalientes.

Software	Modalidad		Costo		Soporte Inteligencia artificial	
	Online	Offline	OpenSouce	Paga	Machine Learning	Deep Learning
ArcGisPro	Si	Si		Si	Si	Si
QGIS	No	Si	Si		Si	No
gvSIG	No	Si	Si		Si	

Tabla 17: Herramientas para establecer un marco de trabajo

Fuente: Elaboración Propia

En la siguiente tabla 18 se debe detallar la información básica de entrada de imágenes hasta su salida final.

Proceso de Imágenes					
Tipo de Drone		Modelo de Drone			
Nombre del Proyecto					
Fecha de Procesamiento					
Modelo de la Cámara		Resolución			RGB
Total cantidad de imágenes a Procesar		Numero de imágenes localizadas			
Proporciones de imágenes iniciales					
(imagen conjunta de vuelos)			(Vuelo individual)		
			(Vuelo individual)		
			(Vuelo individual)		
Imágenes individuales de cada muestra					
Muestra N°	1	GSD		Cantidad de Imagens	
(imagen ortomosaico)			(imagen rgb)		
Modelo fotogrametrico global					
(imagen ortomosaico) global			(imagen rgb global)		
Detalle de Procesamiento					
Hardware	(datos físicos del equipo computacional)				
Sistema operativo	(versión del sistema operativo)				
Sistema de Coordinados					
Sistema de coordenadas de imagen	(sistema de coordenadas de entrada)				
Sistema de coordenadas de salida	(sistema de coordenadas de salida)				
Modelo Fotogramétrico	(formato de modelo fotogramétrico 2D y 3D)				

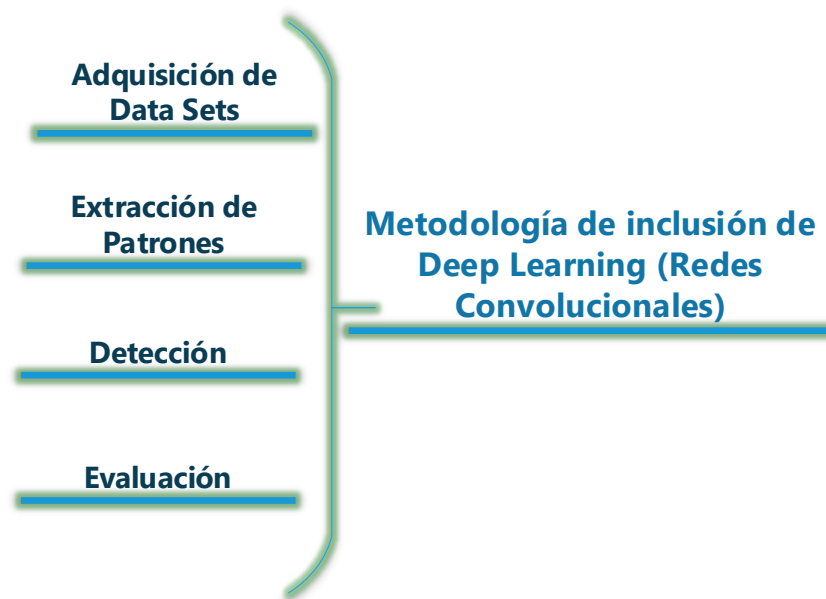
Tabla 18: Proceso de Imágenes

Fuente: Elaboración Propia

3.2.2 METODOLOGÍA DE INCLUSIÓN DE DEEP LEARNING

La Metodología está estructurada en cinco etapas las cuales conllevan también actividades que se deben desarrollar para cumplir cada etapa y obtener en las imágenes fotogramétricas la Inclusión de Deep Learning.

En la Figura 14 se muestra las cinco etapas que conforma la metodología para poder desarrollar con la plataforma ArcGis Pro.



*Figura 14: Metodología Inclusión de Deep Learning Red Convolutional
Fuente: Elaboración Propia*

3.2.2.1 Adquisición de Data Sets:

Se seleccionan conjuntos de ejemplos etiquetados de lo que se desea detectar que servirán como entrenamiento para el aprendizaje de modelos. Las siguientes consideraciones son tomadas en cuenta para elegir los data sets.

- **Tipo de imagen:** las imágenes deben contener el objeto que se desea detectar, considerando los diferentes enfoques que se pueden connotar con los sensores que captan la imagen.

- **Diversidad:** es recomendable adquirir mayor pluralidad de imágenes.
- **Etiquetado:** debido a que el tipo de aprendizaje tiene un enfoque supervisado, es necesario que las imágenes estén etiquetadas.

3.2.2.2 Extracción de Patrones

Extracción de características: se obtiene una selección de atributos que sirven como representación de las imágenes de los ejemplos obtenidos de las bases de datos. Se empleará una red neuronal convolucional (CNN siglas de Convolutional Neural Network) para llevar a cabo dicha extracción de características.

- **Entrada:** imagen que solo contiene el objeto a analizar con un tamaño cuadrado de $N \times N$.
- **Entrenamiento:** Lo primordial para entrenar un modelo se procede a instalar las dependencias de las herramientas de Deep Learning en Sistemas de Información Geográfica, para contener un desglose de las versiones se debe llenar la siguiente tabla 19.

Ubicación	Python	TensorFlow	CUDA	cuDNN
Escritorio				
ArcGis-Pro				

Tabla 19: Ubicación y versión de instalación de dependencias de Deep Learning para documentar

Fuente: Elaboración Propia

Una vez ya identificado todas las instalaciones de Deep Learning se procede a convertir el data sets mediante la herramienta que se ejecuta en la plataforma Jupyter-Notebook (Convert to Pascal) y luego entrenar el modelo

3.2.2.3 Detección

Los data sets de características de cada imagen obtenidos en el paso anterior, sirven como datos de entrenamiento para los algoritmos de aprendizaje, teniendo como objetivo el de aprender modelos capaces de detectar.

- **Entrada:** ortomosaico de características de cada imagen del conjunto de ejemplos de entrenamiento.
- **Procesamiento:** ortomosaico de características pasa por un algoritmo de aprendizaje que sirve como entrenamiento para conocer y aprender.
- **Salida:** crea un modelo capaz de detectar el objeto requerido. y los datos de salida se debe llenar en la siguiente tabla considerando los resultados de detección mediante el diagrama de dispersión.

Entrada Muestra Ortomosaico				
Muestra #				
(Ortomosaico)				
Hora de Ejecución(hh:mm)		Padding		Batch Size
Grafica de Diagrama de Dispersión				
(Grafica de Dispersión)				
Observaciones				

*Tabla 20: Salida Inclusión de Deep Learning
Fuente: Elaboración Propia*

3.2.2.4 Evaluación

Por último, se evalúa el modelo aprendido usando conjuntos de imágenes de pruebas, además de aplicar medidas de evaluación.

Entrada: conjunto de imágenes de prueba etiquetadas.

Proceso: las imágenes pasan por el modelo aprendido y son detectados el objeto.

Evaluación: Por mediante del Modelo Digital de Elevación (DEM), se puede verificar el nivel de detección en caso de un reconocimiento de construcción y considerando las variables de la matriz de confusión, que se puede llenar en la siguiente tabla 21.

Evaluación Muestra 1		
Modelo Digital del Modelo		Ortomosaico con Detección
Resultados de Detección de Viviendas		
Verdadero Positivo	TP (True Positive):	
Falso Negativo	FN (False Negative):	
Falso Positivo	FP (False Positive):	
Verdadero Negativo	TN (True Negative):	
Observaciones		

*Tabla 21: Evaluación de Muestra DEM-Ortomosaico
Fuente: Elaboración Propia*

3.3 POBLACIÓN Y MUESTRA

De acuerdo a la revisión bibliográfica respecto a Deep Learning (Redes Convolucionales) y empleando la metodología de levantamiento fotogramétrico se establece de la siguiente manera.

- **Población:** Se establece puntos estratégicos de la Zona Central del Municipio de Cobija.
- **Muestra:** Se establece la cantidad de imágenes(ortomosaico) acorde a planificación de Vuelo con el dron Mavic 2 Pro, que son 80 imágenes por Hectárea.

3.4 TÉCNICAS DE PROCESAMIENTO Y ANÁLISIS DE DATOS

Las métricas en el aprendizaje profundo son técnicas que se utilizan para realizar una evaluación correcta de los datos.

Previamente a conocer las distintas métricas que se utilizan, debemos definir los siguientes elementos que se utilizan para la evaluación de estas. A la hora de realizar un algoritmo con la intención de predecir los datos mediante un aprendizaje supervisado, es decir, sabiendo los resultados esperados, se pueden dar cuatro casos. (Sitiobigdata, 2019)

- TP (True Positive): cuando en el conjunto de datos el valor era uno, es decir, verdadero, y en la predicción también es uno.
- FN (False Negative): cuando en el conjunto de datos el valor era uno, pero en la predicción el resultado es cero.

- FP (False Positive): cuando en el conjunto de datos el valor era cero, es decir, falso, y en la predicción el resultado es uno
- TN (True Negative): cuando en el conjunto de datos el valor era cero y en la predicción también es cero.

3.6.1 Matriz de Confusión

Una vez definidos FP, FN, TP y TN, podemos proceder con el análisis de la matriz de confusión se muestra en la tabla 22

	Positivo (1)	Negativo (0)
Positivo (1)	TP	FP
Negativo (0)	FN	TN

Tabla 22: Matriz de Confusión

Fuente: (Sitiobigdata, 2019)

Como hemos visto, la matriz de confusión es una técnica muy útil que se puede utilizar para identificar rápidamente verdaderos y falsos positivos y falsos negativos visualmente. De esta manera, podemos determinar fácilmente qué salió mal en el modelo de aprendizaje supervisado.

Exactitud (Accuracy): La exactitud se define como el porcentaje total de elementos clasificados correctamente, es decir, el número de predicciones que se han realizado correctamente dividido por el número total de predicciones.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Aplicando la fórmula de Exactitud (Accuracy), se debe llenar la siguiente tabla 23 una vez obtenido las muestras.

TRUE POSITIVE	FALSE POSITIVE
Numero de TP	Numero de FP
FALSE NEGATIVE	TRUE NEGATIVE
Numero de FN	Numero de TN

Tabla 23: TP, FP, FN y TN (Matriz de confusión).

Fuente: Elaboración Propia

3.6.2 TensorBoard

En esta tesis se utiliza TensorFlow, la cual contiene una herramienta denominada. TensorBoard que es una interfaz para mostrar gráficos y otras herramientas para comprender, depurar y optimizar modelos. La figura 15 proviene lo que genera TensorBoard es cual se muestra su panel principal. La cual contiene diferentes módulos destinadas al nivel de información y ejecución del modelo, los cuales son:

- Escalares: Muestra diferentes informaciones útiles durante el entrenamiento del modelo
- Gráficos: Mostrar el modelo
- Histograma: muestra los pesos con un histograma
- Distribución: Muestra la distribución del peso
- Proyector: Mostrar análisis de componentes principales y algoritmo T-SNE. La técnica utiliza para la reducción de dimensionalidad

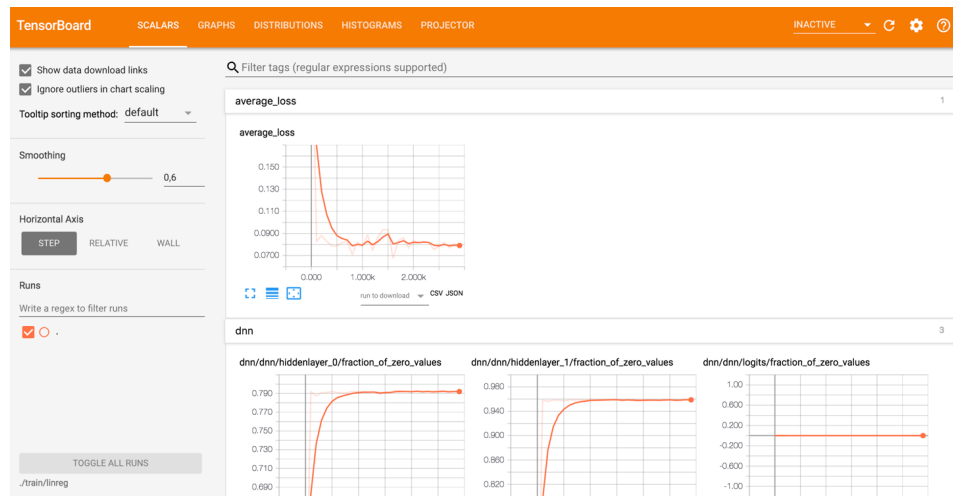


Figura 15: Interfaz de TensorBoard
Fuente: (TensorBoard, 2019)

CAPITULO IV

4. DESARROLLO DE LA INVESTIGACIÓN

4.1 INTRODUCCIÓN

Para el desarrollo de la investigación se aplica la metodología herramientas y técnicas descrita en el capítulo anterior, en este sentido se ha desarrollado el levantamiento fotogramétrico mediante el uso del dron, para luego procesar las imágenes (ortomosaico) de los predios seleccionados, luego se establece el conjunto de patrones de las imágenes fotogramétricas procesadas para el entrenamiento del modelo, después se ha definido una arquitectura Deep Learning para la detección de vivienda de las imágenes fotogramétricas de los predios de Municipio de Cobija, posteriormente pasa a evaluar la capacidad del modelo en la detección de vivienda de los predios seleccionados del Municipio de Cobija. y finalmente se llega a representar los resultados obtenidos.

4.2 LEVANTAMIENTO FOTOGRAMÉTRICO CON DRONES

La metodología de levantamiento fotogramétrico con drones, consiste desde el análisis y estudio de la región a aplicar la investigación, Captura de imágenes y procesamiento de imágenes

4.2.1 Análisis y Estudio de la región a aplicar la investigación

- **Examinar condiciones físicas del territorio**

En la tabla 24 se presenta el informe de las condiciones físicas del territorio dentro del área de interés.

Informe de examinar las condiciones físicas del territorio					
País	Bolivia	Departamento	Pando	Ciudad	Cobija
Fecha de Examinación	11 de octubre del 2019				
Observaciones					
En el Municipio de Cobija se presenta áreas desniveladas en sus diferentes distritos, con una alta densidad de áreas verdes, como objeto de estudio dentro de la detección de viviendas se considera todos estos factores los cuales implican dificultad de visibilidad, desnivel de altura de vuelo en diferentes distritos del Municipio de Cobija.					

Tabla 24: Informe de Condiciones físicas del Territorio
Fuente: Elaboración Propia

• **Dimensiones de la región de interés**

En la siguiente tabla 25 se describe toda información requerida para contemplar las dimensiones de la región de interés con sus respectivas restricciones y acciones a seguir si se presentan varias restricciones en la figura 12 se denotan las descripciones de colores que establece las normas Zonas Geo 2.0 de la línea DJI.

Dimensiones de la región de interés					
Herramienta	Flysafe geo-map DJI	Drone (RPA)	Mavic 2 Pro		
Enlace Web de la Herramienta		https://www.dji.com/flysafe/geo-map			
País	Bolivia	Departamento	Pando	Ciudad	Cobija
Imagen de región de interés					






Nivel de Restricciones	
 Zonas restringidas	Se presenta esta restricción por la zona aeroportuaria y se desbloquea bajo solicitud gubernamental, en un horario de vuelo establecido
 Zonas de Autorización	Presenta autorización al iniciar vuelo como un back log
 Zonas de altitud	No aparece en la zona de interés
 Zonas de advertencia	No aparece en la zona de interés
 Zonas de advertencia mejoradas	No aparece en la zona de interés
Observaciones	Ninguna respecto a la zona de interés

Tabla 25: Dimensiones de la región de interés

Fuente: Elaboración Propia

Por lo tanto, la tabla 25 se presenta la imagen de interés la cual solo presenta una zona de autorización la cual se desbloquea mediante un vuelo con back log lo cual es libre sin ingresar a zona roja que es área de aeropuerto.

4.2.2. Captura de Imágenes

• Parámetros climáticos y geomagnetismo del día

En el Municipio de Cobija se contempla un tipo de clima tropical húmedo (Oca, 2015), esto involucra que el factor de ráfagas de viento tiene que ser menor a lo que soporta cada drone para desarrollar el modelo fotogramétrico.

En la tabla 26, se describe los parámetros climáticos y geomagnetismo de los días que se desarrolló el muestro con el drone, teniendo como resultado de ocho días bajos en ráfagas de viento y un nivel alto de geomagnetismo de 5Kp.

Parámetros climáticos y geomagnetismo del día			
Modelo de Drone		Mavic 2 Pro	
Software		Uavforecast	
Fecha	Ráfagas de viento Km/h	Geomagnetismo Kp(Kilopondio)	Satélites disponibles
11-10-2019	27	2	12
12-10-2019	21	2	12
13-10-2019	34	2	16

14-10-2019	12	5	7
15-10-2019	22	1	19
16-10-2019	11	3	11
17-10-2019	18	4	9
18-10-2019	22	4	8
Observaciones			
El muestreo se realizó un 11 de octubre del 2019, hasta el 18 de octubre el cual se presentó un 14 de octubre un geomagnetismo de 5 Kp			
	Vuelos Permitidos		
	Vuelos No Permitidos		

Tabla 26: *Parámetros climáticos y geomagnetismo del día*
Fuente: *Elaboración Propia*

• **Establecer Muestra de Puntos de Altura Adecuada**

Al establecer la muestra de puntos de altura en el Municipio de Cobija se denota aspectos de desnivelación de suelos y la altura máxima de inmuebles permitidos por la unidad de catastro, para tener una altura ideal se aplica la fórmula de altura con función a la resolución espacial (GSD.)

En la figura 18, el diagrama de dispersión conectado detalla las muestras que se tomaron en el Municipio de Cobija, para luego obtener la altura ideal.

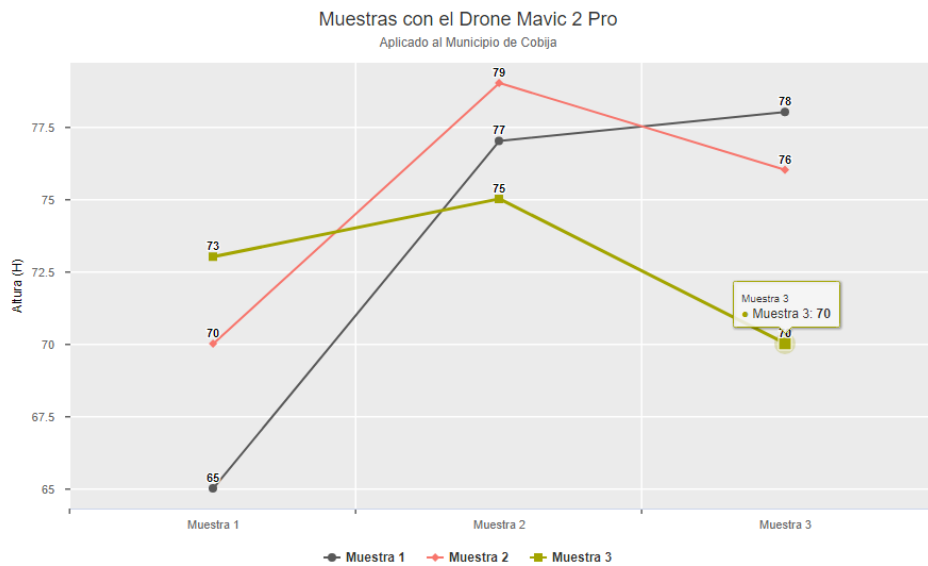


Figura 18: *Muestras con el drone (RStudio)*
Fuente: *Elaboración Propia*

La primera muestra está ubicada en el distrito uno del Municipio de Cobija, en la calle Tarija y 9 de febrero área baja, la cual se realizó muestras con el drone Mavic 2 Pro a tiempo real con el GPS del drone. En la siguiente tabla 27 se detalla todo lo necesario para poder proceder a un vuelo planificado considerando altura a función GSD.


Muestreo de Altura (h)							
Drone	Mavic2Pro	Altura Ideal	77	m	GSD Ideal	0,018	m/px
Muestra						N°	1
Coordenadas		Imagen de Referencia				Altura(m)	
-11.016136	-68.75506					77	
$h = \frac{f * d}{S_w} =$		Altura de Vuelo			h	77	m
		Distancia Focal (Acorde a formato)			f	0,10	mm
		Ancho de Área real Cubierta			d	98,5	m
		Ancho de sensor			S_w	0,13	mm
		Resolución Espacial			GSD	0,018	m/px
		Ancho de cada Imagen (Acorde a Formato)			P_w	5472	px
$d = GSD * P_w =$		Resolución Espacial			0,018	m/px	
		Ancho de cada Imagen (Acorde a Formato)			5472	px	
Observaciones		Sin ninguna observación alineal					

Tabla 27: Resultado de Muestreo N° 1

Fuente: Elaboración Propia

La segunda muestra está ubicada interseccionada entre el distrito 1 y 2 del Municipio de Cobija en calle sin nombre y Av. 9 de febrero, se estableció la altura en esta zona de área media con urbanización continua, las pruebas de altura se realizaron con el drone Phantom 4 Pro Avanced y Mavic 2 Pro, denotando como prioridad la precisión en GSD y GPS, el Mavic 2 Pro es más preciso. En la siguiente tabla 28 se detalla todo lo necesario para proceder a un vuelo planificado.

Muestreo de Altura							
Drone	Mavic2Pro	Altura Ideal	76	m	GSD Ideal	0,0178	m/px
Muestra (h)						N°	2



Coordenadas		Imagen de Referencia	Altura		
-11.022989	-68.766006		76		
$h = \frac{f * d}{S_w} =$		Altura de Vuelo	<i>h</i>	76	m
		Distancia Focal (Acorde a formato)	<i>f</i>	0,103	mm
		Ancho de Área real Cubierta	<i>d</i>	97,4	m
		Ancho de sensor	<i>S_w</i>	0,13	mm
		Resolución Espacial	<i>GSD</i> <i>D</i>	0,0178	m/px
		Ancho de cada Imagen (Acorde a Formato)	<i>P_w</i>	5472	px
$d = GSD * P_w =$		Resolución Espacial	0,0178	m/px	
		Ancho de cada Imagen (Acorde a Formato)	5472	px	
Observaciones		Sin ninguna observación			

Tabla 28: Resultado de Muestreo de Altura N°2

Fuente: Elaboración Propia

En la tercera muestra está ubicada en el distrito 4 del Municipio de Cobija en calle sin nombre y Av. Acre, al establecer la altura de vuelo ideal se presenta una declinación de suelo para ello se toma desde la zona más alta y plana para compensar el GSD, se realizó el vuelo de muestra con el drone Mavic 2 Pro, observando que el área de interés tiene una urbanización dispersa, en la siguiente tabla 29 se detalla todo lo necesario para proceder a un vuelo planificado.

Muestreo de Altura							
Drone	Mavic2Pro	Altura Ideal	70	m	GSD Ideal	0,0164	m/px
Muestra (h)						N°	3
Coordenadas		Imagen de Referencia			Altura		
-11.034538	-68.75564				70		
$h = \frac{f * d}{S_w} =$		Altura de Vuelo	<i>h</i>	70	m		
		Distancia Focal (Acorde a formato)	<i>f</i>	0,103	mm		
		Ancho de Área real Cubierta	<i>d</i>	89,7	m		
		Ancho de sensor	<i>S_w</i>	0,132	mm		
		Resolución Espacial	<i>GSD</i> <i>D</i>	0,0164	m/px		
		Ancho de cada Imagen (Acorde a Formato)	<i>P_w</i>	5472	px		

$d = GSD * P_w =$	Resolución Espacial	0,0164	m/px
	Ancho de cada Imagen (Acorde a Formato)	5472	px
Observaciones	Sin ninguna observación		

Tabla 29: Resultado de Muestreo N°3

Fuente: Elaboración Propia

• Fijar Puntos de referencia

Para perfilar una mejor precisión se fija puntos de coordenadas para obtener una georreferenciación, en la siguiente tabla 30 se llena las coordenadas cercanas a los puntos de muestras.




Coordenadas		Imagen de Referencia	
-11.01632	-11.01632 -		
-11.022511	-68.766844		
- 11.034555	- 68.755634		

Tabla 30: Puntos de referencia levantados

Fuente: Elaboración Propia

• Planificar y simular el vuelo

Para planificación de vuelo se debe considerar aspectos técnicos los cuales se obtiene los metadatos del drone correlacionados a las muestras en diferentes lugares del Municipio de Cobija. En las siguientes tablas 31, 32 y 33. Se levanta los metadatos del drone.







Verificación de vuelo					
Drone	Mavic2Pro	Fecha		N° de Vuelo	1
Estado del GPS	Estado del IMU	Satélites disponibles	Nivel de Geomagnetismo Kp	Viento Km/h	Ráfagas de Viento Km/h
 Excelente	 Excelente	21	1	2	3
 Buena	 Buena				
 Mala	 Mala				

Tabla 31: Verificación de Vuelo N°1

Fuente: Elaboración Propia

Verificación de vuelo					
Drone	Mavic2Pro	Fecha		N° de Vuelo	2
Estado del GPS	Estado del IMU	Satélites disponibles	Nivel de Geomagnetismo Kp	Viento Km/h	Ráfagas de Viento Km/h
Excelente	Excelente	19	2	3	5
Buena	Buena				
Mala	Mala				

Tabla 32: Verificación de Vuelo N°2
Fuente: Elaboración Propia

Verificación de vuelo					
Drone	Mavic2Pro	Fecha		N° de Vuelo	3
Estado del GPS	Estado del IMU	Satélites disponibles	Nivel de Geomagnetismo Kp	Viento Km/h	Ráfagas de Viento Km/h
Excelente	Excelente	20	1	2	3
Buena	Buena				
Mala	Mala				

Tabla 33: Verificación de Vuelo N°3
Fuente: Elaboración Propia

La obturación de disparo también llamado velocidad de disparo nos ayuda a determinar el tiempo en que el obturador de la cámara deja pasar luz al sensor, en las diferentes muestras que realizamos variante al GSD y a la velocidad que se dirige el dron, en las siguientes tablas 34, 35 y 36 se detalla la obturación de disparo de cada muestra.

Numero de Muestra	1			
$OC = GSD / 2S$	OC	obturación de cámara	0,0013	[s]
	GSD	resolución espacial	0,018	[m/px]
	S	velocidad de vuelo	2*6,74	[m/s]

Tabla 34: Muestra N°1 Obturación de disparo
Fuente: Elaboración Propia

Numero de Muestra	2			
$OC = GSD / 2S$	OC	obturación de cámara	0,0012	[s]
	GSD	resolución espacial	0,0178	[m/px]
	S	velocidad de vuelo	2*7,2	[m/s]

Tabla 35: Muestra N°2 Obturación de disparo
Fuente: Elaboración Propia

Numero de Muestra	3			
$OC = GSD / 2S$	OC	obturación de cámara	0,0009	[s]
	GSD	resolución espacial	0,0154	[m/px]
	S	velocidad de vuelo	2*8,21	[m/s]

Tabla 36: Muestra N°3 Obturación de disparo

Fuente: Elaboración Propia

• Ejecutar la Misión Planificada

Para ejecutar la Misión planificada en cada punto de muestra del Municipio de Cobija, se copia los datos al software de planificación de vuelo, al finalizar el proceso se detalla de cada muestra uno, dos y tres en las tablas 37, 38 y 39 los resultados más relevantes del vuelo planificado.

Muestra N°	1			
Software	Pix4D			
Cantidad de Imágenes	84			
Tiempo de Vuelo	5	min	23	s
Tipo de Vuelo	Cuadrícula Simple			
Angulo de la Cámara	90°			

Tabla 37: Referencia de Vuelo Muestra N°1

Fuente: Elaboración Propia

Muestra N°	2			
Software	Pix4D			
Cantidad de Imágenes	46			
Tiempo de Vuelo	3	min	23	s
Tipo de Vuelo	Cuadrícula Simple			
Angulo de la Cámara	90°			

Tabla 38: Referencia de Vuelo Muestra N°2

Fuente: Elaboración Propia

Muestra N°	3			
Software	Pix4D			
Cantidad de Imágenes	105			
Tiempo de Vuelo	5	min	38	s

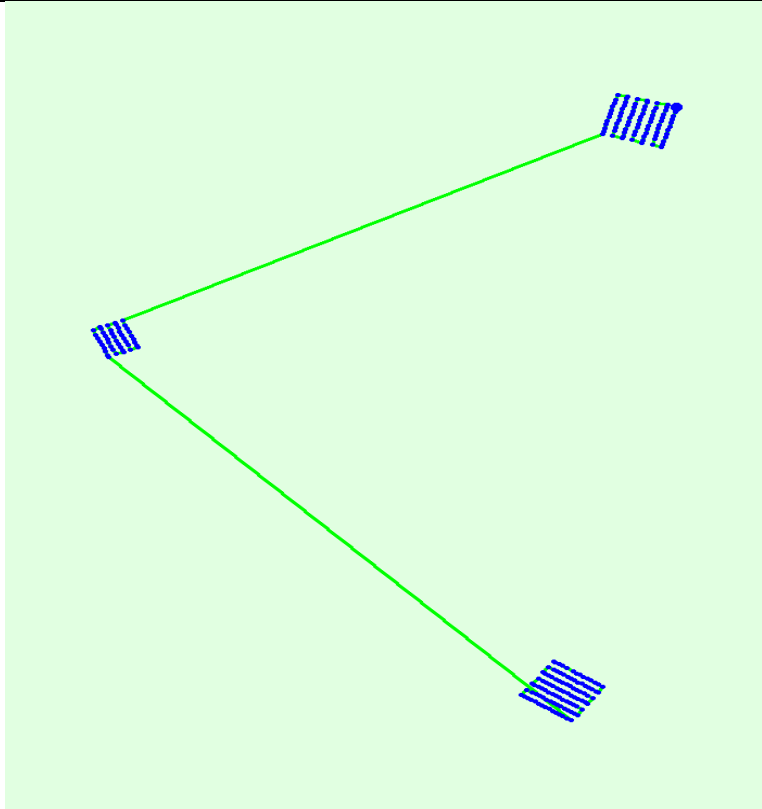
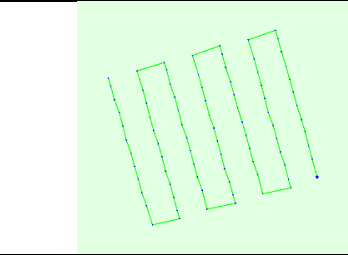
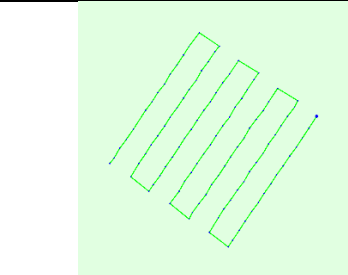
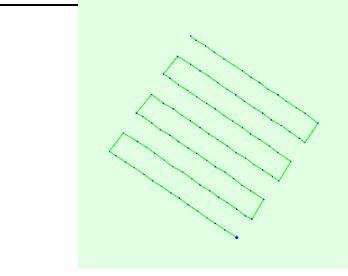
Tipo de Vuelo	Cuadrícula Simple
Angulo de la Cámara	90°


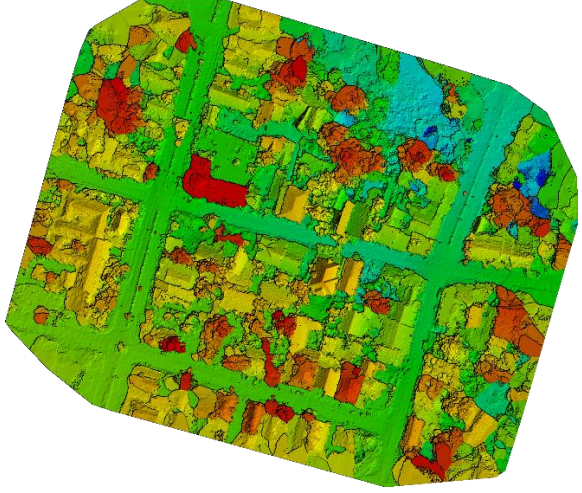

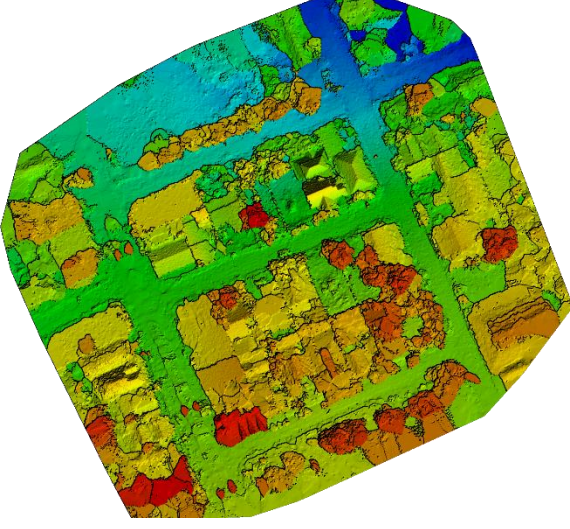

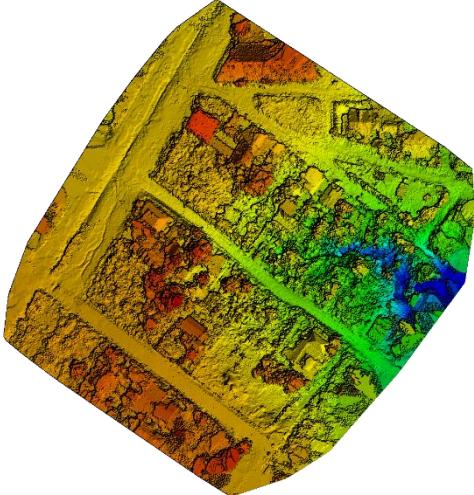
Tabla 39: Referencia de Vuelo Muestra N°3
Fuente: Elaboración Propia

4.2.3. Procesamiento de Imágenes

- **Establecer en un Marco de Trabajo**

Se establece como plataforma de desarrollo el Software ArcGIS Pro 2.5, en donde deben pasar las imágenes aéreas para producir ortomosaicos, modelos fotogramétricos de las diferentes muestras, en la tabla 40 se detalla todo el proceso que se realiza desde la entrada de las imágenes tomadas por el drone hasta el modelo fotogramétrico.

Proceso de Imágenes					
Tipo de Drone	Multirrotor	Modelo de Drone	Mavic 2Pro		
Nombre del Proyecto	Tesis				
Fecha de Procesamiento	23	Octubre	2019		
Modelo de la Cámara	L1D-20c10.3	Resolución	5472	3648	RGB
Total cantidad de imágenes a Procesar	235	Numero de imágenes localizadas	235		
Proporciones de imágenes iniciales					
					
					
					

Imágenes individuales de cada muestra					
Muestra N°	1	GSD	0,018	Cantidad de Imagens	84
					
Muestra N°	2	GSD	0,0178	Cantidad de Imagens	46
					
Muestra N°	3	GSD	0,0164	Cantidad de Imagens	105
					

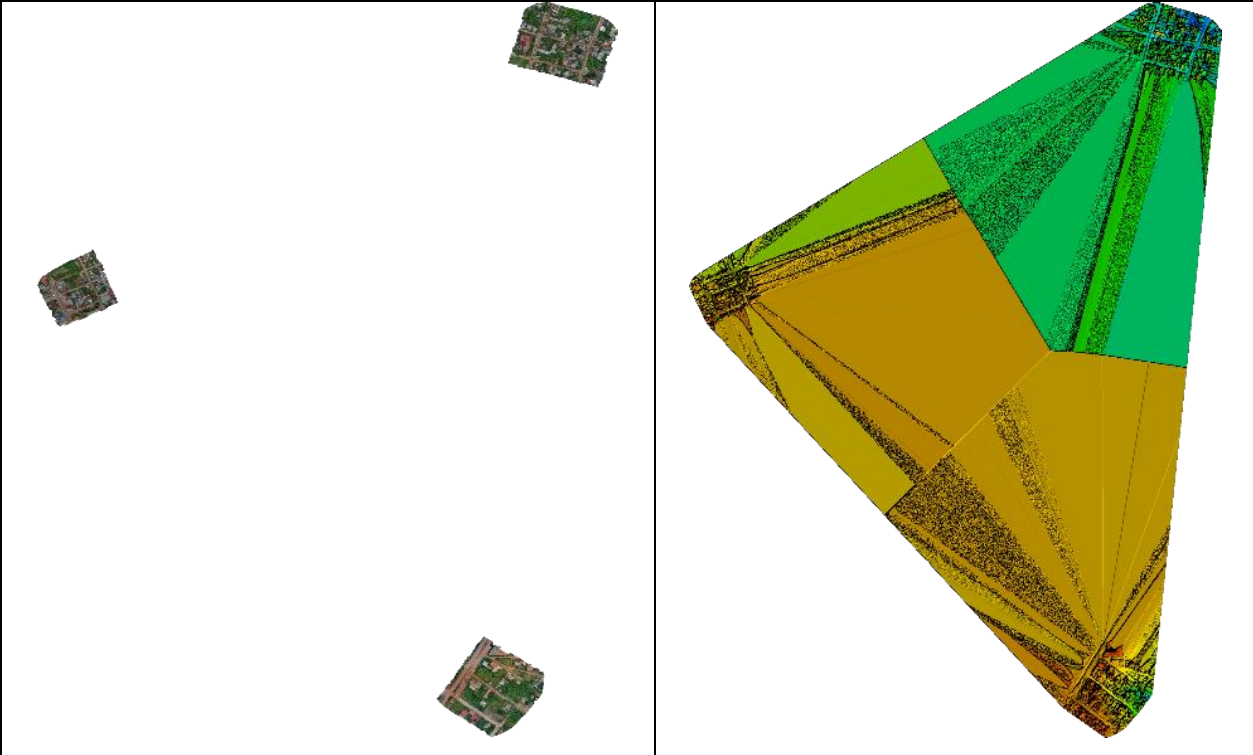
Modelo fotogrametrico global	
	
Detalle de Procesamiento	
Hardware	CPU: Intel (R) Core (TM) i7-9750H CPU @ 2.60GHz RAM: 32GB GPU: Intel (R) UHD Graphics 630 (Controlador: 27.20.100.7922), NVIDIA GeForce RTX 2070 con diseño Max-Q (Controlador: 26.21 .14.4259)
Sistema operativo	Windows 10 Home, 64 bits
Sistema de Coordinados	
Sistema de coordenadas de imagen	GCS_WGS_1984 (EGM 96 Geoide)
Sistema de coordenadas de salida	WGS_1984_UTM_Zone_19S (EGM 96 Geoid)
Modelo Fotogramétrico	2D

Tabla 40: Proceso de imágenes de muestras 1, 2 y 3

Fuente: Elaboración Propia

4.3 INCLUSIÓN DE DEEP LEARNING

La metodología inclusión de Deep Learning consiste desde la adquisición de data sets, extracción de patrones, detección y evaluación para establecer un modelo de detección de viviendas en el centro del Municipio de Cobija.

4.3.1 Adquisición de Data Sets:

- **Tipo de imagen:** El conjunto de imágenes, llega a formar un ortomosaicos en formato tiff, de los diferentes puntos de interés dentro del Municipio de Cobija, en la siguiente figura 16, se plasma el ortomosaicos y su formato.

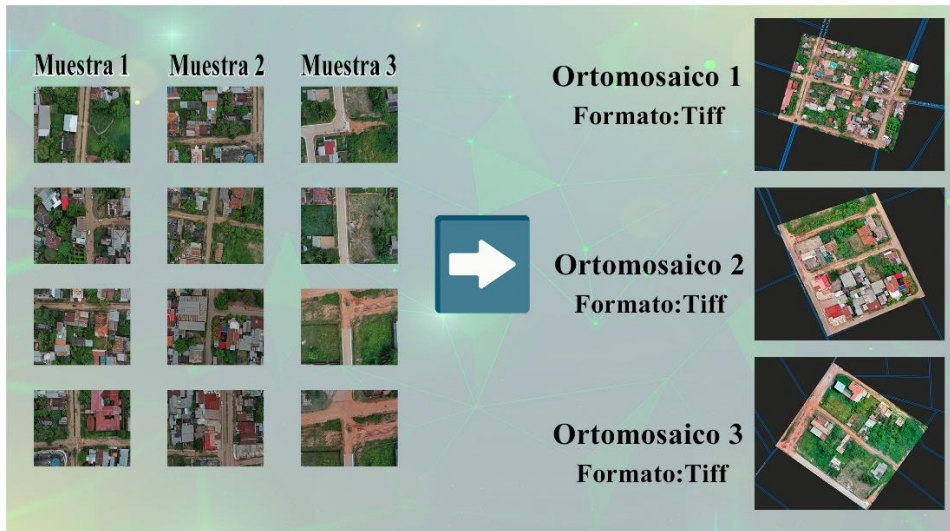


Figura 16: Ortomosaicos y su formato

Fuente: Elaboración Propia

- **Diversidad:** Dentro de los puntos de interés se presenta una diversidad de viviendas, avenidas, calles, red de energía eléctrica y vegetación, en la presente figura 17 se diversifica lo que hay en el ortomosaico

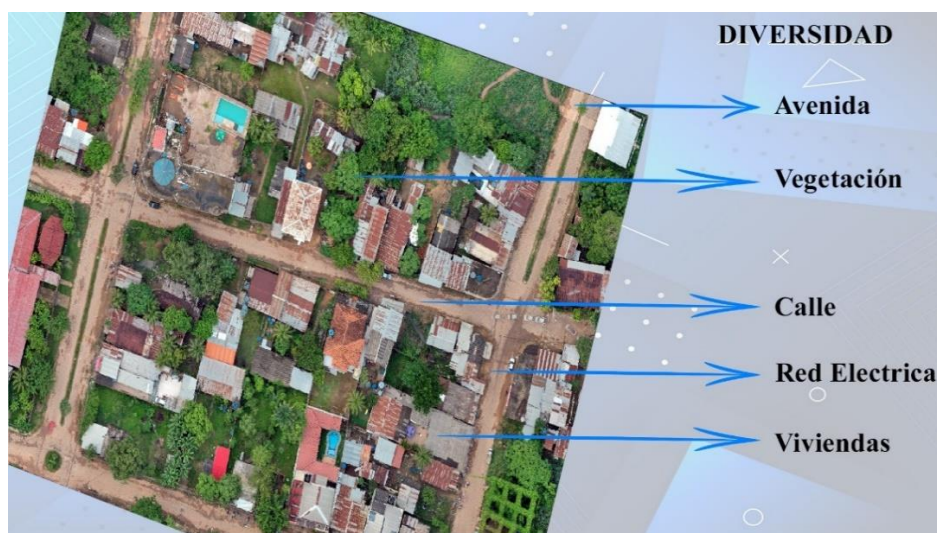


Figura 17: Diversidad del Ortomosaico

Fuente: Elaboración Propia

- **Etiquetado:** El tipo de aprendizaje es supervisado, así que se etiqueta nuestro objeto a detectar que son las viviendas de los puntos de interés del Municipio de Cobija, en la siguiente figura 18 de denota el etiquetado.



*Figura 18: Etiquetado para el Aprendizaje Supervisado
Fuente: Elaboración Propia*

4.3.2 Extracción de Patrones

- **Entrada:** Las viviendas son el objeto a detectar dentro del ortomosaico, se crea una clase vivienda la cual se tiene que etiquetar dentro de los patrones que caracteriza una vivienda para luego obtener un inicio del data sets y finalmente exportar el data sets en formato tiff en dimensión 1500 x 1500 pixel, todo este proceso se desarrolla en la plataforma de ArcGis Pro, en la siguiente figura 19 se detalla el proceso.



Figura19: Entrada Inicio Data Sets en la Plataforma ArcGis Pro
Fuente: Elaboración Propia

En la figura 19, se describe desde el proceso de crear una clase para identificar el objeto que es la vivienda con un valor = 1 denotando la clase de color rojo, luego pasa a etiquetar las viviendas de forma de polígono demarcando el contenido que presentan los diferentes patrones de cada muestra contando con 64 muestras luego se procede a exportar los data sets en formato tiff con una dimensión de cada imagen 1500 x 1500.

Entrenamiento: Se debe instalar las dependencias necesarias para poder trabajar con Deep Learning, en la siguiente tabla 41 se describe de las dependencias para la inclusión de Deep Learning con las versiones correspondientes para cada ubicación.

Ubicación	Python	TensorFlow	CUDA	cuDNN
Escritorio	3.7	2		
ArcGis-Pro	3.6	1.14-GPU	10	7.4

Tabla 41: Ubicación y versión de instalación de dependencias de Deep Learning
Fuente: Elaboración Propia

Al tener todas las dependencias para empezar el entrenamiento se procede a convertir a un formato de optimización en detección (Converter to Pascal) el cual se ejecuta en

Jupyter-Notebook, Una vez optimizado el entrenamiento se procede a formar la arquitectura del modelo basado en InceptionV2-2018

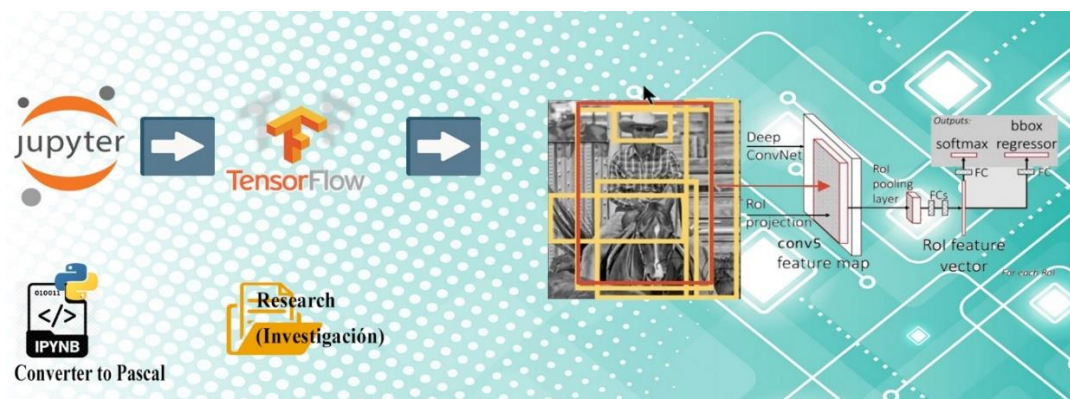


Figura 20: Proceso de Entrenamiento
Fuente: Elaboración Propia

En la siguiente figura 20 se describe la utilización del código de optimizador “Converter to Pascal” se divide en cuatro casillas las cuales están en anexos c, el cual la primera casilla se ejecuta para importar las herramientas de TensorFlow, la segunda casilla consta de análisis de los archivos xml que tiene los data sets para su uso en Deep Learning, la tercera casilla se aplica el análisis de los archivos map.txt los cuales contienen información de la imagen tiff como coordenadas de cada extremo mínimas y máximas, la cuarta casilla genera archivos (entrenamiento y prueba)de entrenamiento optimizado. Finalmente se procede a entrenar el modelo en TensorFlow con la arquitectura InceptionV2-2018 la cual tiene un ciclo de entrenamiento de 4381, y la función de activación se utiliza softmax. Y un tiempo estimado de dos horas, y finalmente nos genera un archivo con extensión dlpc.

4.3.3 Detección

- **Entrada:** Como entrada introducimos el ortomosaicos por muestra individuales uno dos y tres.
- **Procesamiento:** Se introduce el modelo entrenado en la plataforma de ArcGis para ejecutar, considerando las configuraciones predefinidas.

En la figura 21 se describe la entrada de cada muestreo individual como en conjunto, para probar el modelo, en el compilador de ArcGisPro en la herramienta de Using Deep Learning, donde aparecen los parámetros de estrada de muestra, salida de muestra,

ubicación del modelo generado con extensión dlpk y su código del modelo se describe en el anexo D y se tiene la opción de mejorar las variables de detección los cuales son: padding=256 y batch_size= 4, por último, queda ejecutar la detección.

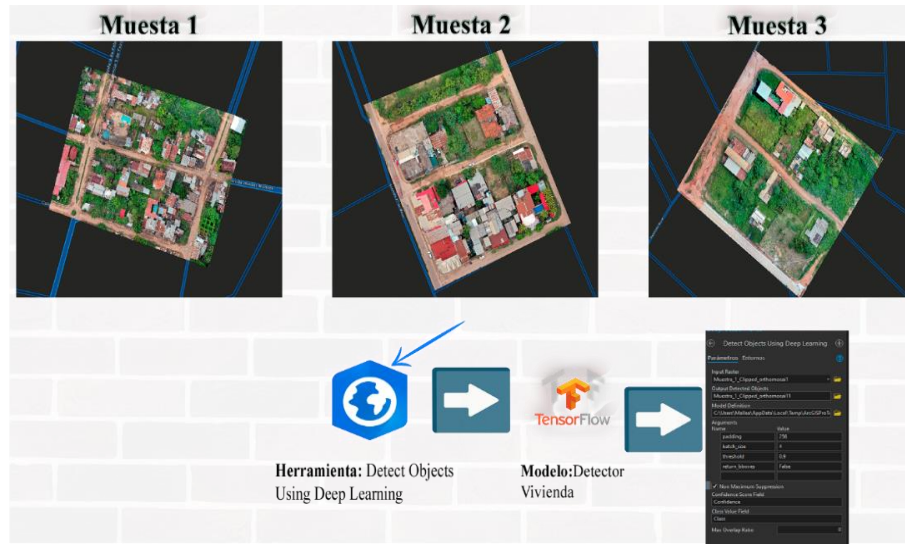


Figura 21: Entrada y Proceso de Inicio de Detección
Fuente: Elaboración Propia

- **Salida:** Visualización del comportamiento del modelo en la detección de viviendas del Municipio de Cobija.

En cada muestra de ortomosaico el modelo presenta un rango de confianza del 90 al 99 % de cada vivienda detectada, pero hay una varianza en el diagrama de dispersión obteniendo en la muestra 1 = 94.95381, muestra 2 = 93.94513 y muestra 3 = 95.04312. En las siguientes tablas 42, 43 y 44.

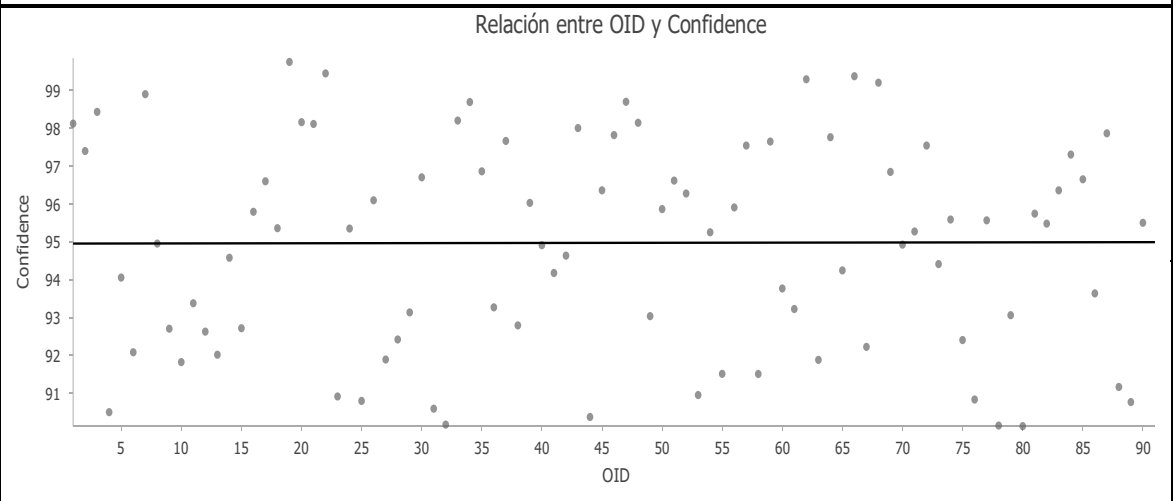
Entrada Muestra Ortomosaico

Muestra 1



Hora de Ejecución(hh:mm)	00:35	Padding	256	Batch Size	4
---------------------------------	-------	----------------	-----	-------------------	---

Grafica de Diagrama de Dispersión



Observaciones

Mediante el Diagrama de Dispersión se aprecia que mayoría de los puntos detectados están en un rango de entre 90 y 99 % de Confianza. Y obteniendo una tendencia lineal 94.95381

Tabla 42: Muestra 1 con la inclusión de Deep Learning
Fuente: Elaboración Propia

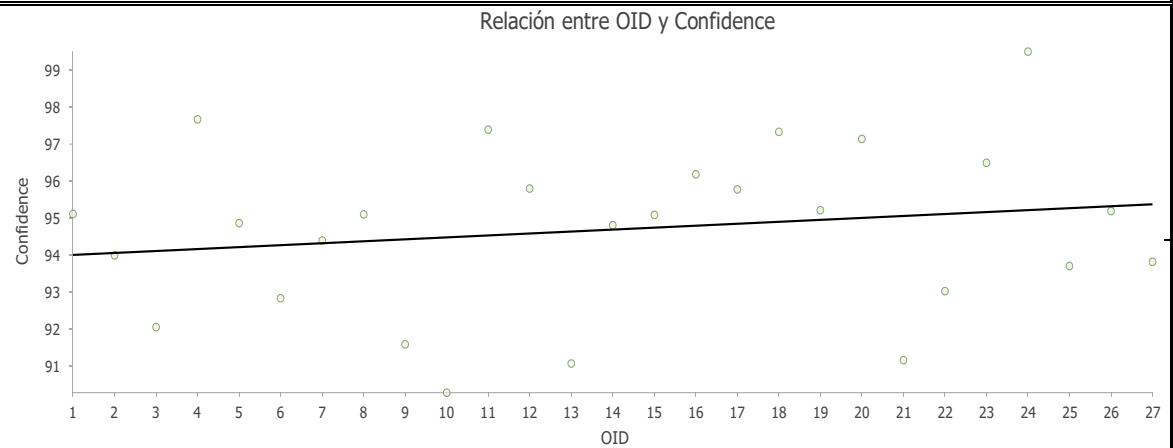
Entrada Muestra Ortomosaico

Muestra 2



Hora de Ejecución(hh:mm)	00:30	Padding	256	Batch Size	4
---------------------------------	-------	----------------	-----	-------------------	---

Grafica de Diagrama de Dispersión



Observaciones

Mediante el Diagrama de Dispersión se aprecia que mayoría de los puntos detectados están en un rango de entre 90 y 99 % de Confianza. Y obteniendo una tendencia lineal 93.94513.

Tabla 43: Muestra 2 con la Inclusión de Deep Learning
Fuente: Elaboración Propia

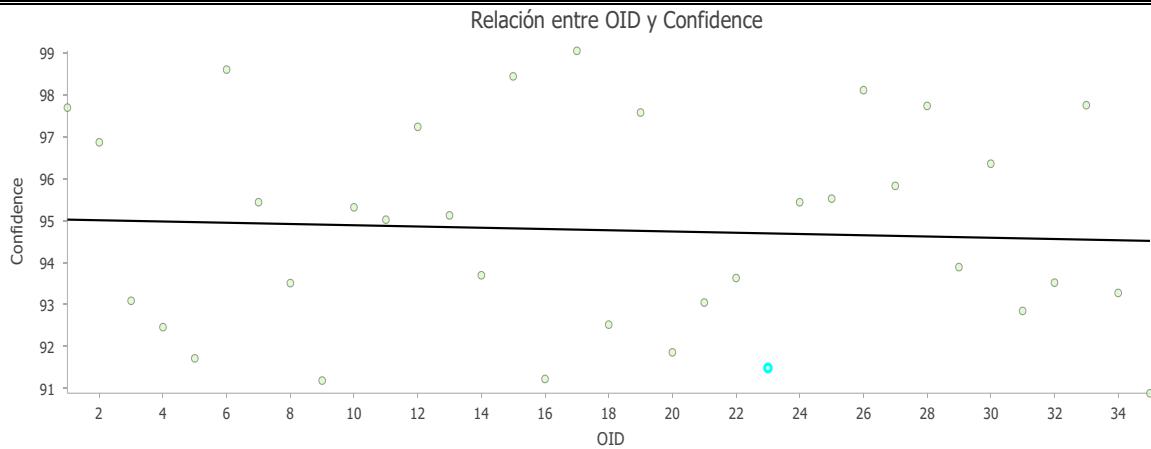
Entrada Muestra Ortomosaico

Muestra 3



Hora de Ejecución(hh:mm)	01:20	Padding	256	Batch Size	4
---------------------------------	-------	----------------	-----	-------------------	---

Grafica de Diagrama de Dispersión



Observaciones

Mediante el Diagrama de Dispersión se aprecia que mayoría de los puntos detectados están en un rango de entre 90 y 99 % de Confianza. Y obteniendo una tendencia lineal 95.04312.

Tabla 44: Muestra 3 con la inclusión de Deep Learning
Fuente: Elaboración Propia

4.3.4 Evaluación

Para evaluar la detección de viviendas podemos verificar y comprobando con el ortomosaico de modelo digital de elevación donde se denota las viviendas, por cada muestra el modelo detecto diferentes cantidades de viviendas,

Utilizando las variables definidas de la matriz de confusión tenemos verdadero positivo (TP), falso positivo (FN), falso positivo (FP) y verdadero negativo (TN), las cuales se van clasificando según lo que el modelo de detección va definiendo, en la muestra 1 llego a detectar 76 viviendas, en la muestra 2 detecto 18 viviendas y en la muestra 3 detecto 28 viviendas. En las siguientes tablas 45, 46 y 47 se visualiza la comparación entre el modelo de elevación digital y el ortomosaico con la detección de vivienda y llenando los valores a las variables definidas de la matriz de confusión y dando a conocer las observaciones.



Evaluación Muestra 1		
Modelo Digital del Modelo		Ortomosaico con Detección
		
Resultados de Detección de Viviendas		
Verdadero Positivo	TP (True Positive):	76
Falso Negativo	FN (False Negative):	10
Falso Positivo	FP (False Positive):	4
Verdadero Negativo	TN (True Negative):	1
Observaciones		
El modelo logro detectar la mayoría de las viviendas establecidas de la muestra 1 teniendo como resultado la detección de viviendas y asu el modelo detecta objetos que no son viviendas pero tienen un patrón similar detectando autos, vegetación o aguas residuales, visual y computado como verdadero positivo (T.P.) un alcance de 76 viviendas		

Tabla 45: Evaluación de la Muestra 1

Fuente: Elaboración Propia





Evaluación Muestra 2		
Modelo Digital del Modelo		Ortomosaico con Detección
		
Resultados de Detección de Viviendas		
Verdadero Positivo	TP (True Positive):	18
Falso Negativo	FN (False Negative):	6
Falso Positivo	FP (False Positive):	2
Verdadero Negativo	TN (True Negative):	1
Observaciones		
<p>El modelo logro detectar la mayoría de las viviendas establecidas de la muestra 2, el modelo llega a autos o sombrillas de motocicleta, las cuales coinciden con un patron similar al de techo de una vivienda, como resultado visual y computado como verdadero positivo (T.P.) un alcance de 18 viviendas</p>		

Tabla 46: Evaluación de la Muestra 2
Fuente: Elaboración Propia

Evaluación Muestra 3		
Modelo Digital del Modelo		Ortomosaico con Detección
		
Resultados de Detección de Viviendas		
Verdadero Positivo	TP (True Positive):	28
Falso Negativo	FN (False Negative):	4
Falso Positivo	FP (False Positive):	1
Verdadero Negativo	TN (True Negative):	2
Observaciones		

El modelo logro detectar la mayoría de las viviendas teniendo como resultado visual y computado como verdadero positivo (T.P.) un alcance de 28 viviendas

Tabla 47: Evaluación de la Muestra 3

Fuente: Elaboración Propia

Mediante la métrica de matriz de confusión se evaluó la exactitud el modelo de detección de viviendas, en cada muestra se establece un diferente porcentaje de exactitud, en la muestra 1 = 87.9 %, muestra 2 = 74.1% y muestra 3= 82.9%. En las siguientes tablas 48, 49 y 50 se denota cada tabla de la matriz de confusión para establecer un resultado de exactitud.

EVALUACIÓN DEL MODELO MEDIANTE MATRIZ DE CONFUSIÓN			
MUESTRA UNO			
Matriz de Confusión		TRUE POSITIVE	FALSE POSITIVE
		76	4
		Numero de TP	Numero de FP
		FALSE NEGATIVE	TRUE NEGATIVE
		10	1
		Numero de FN	Numero de TN
Exactitud (Accuracy)	$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$		
	$Accuracy = \frac{76 + 4}{76 + 1 + 4 + 10} = 0.88$		
Porcentaje de Aprendizaje	87.9%		

Tabla 48: Evaluación de Matriz de confusión Muestra 1

Fuente: Elaboración Propia

EVALUACIÓN DEL MODELO MEDIANTE MATRIZ DE CONFUSIÓN			
MUESTRA DOS			
Matriz de Confusión		TRUE POSITIVE	FALSE POSITIVE
		18	2
		Numero de TP	Numero de FP
		FALSE NEGATIVE	TRUE NEGATIVE
		6	1
		Numero de FN	Numero de TN
Exactitud	$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$		

(Accuracy)	$Accuracy = \frac{18 + 1}{18 + 1 + 2 + 6} = 0.74$
Porcentaje de Aprendizaje	74.1%

Tabla 49: Evaluación de Matriz de confusión Muestra 2
Fuente: Elaboración Propia

EVALUACIÓN DEL MODELO MEDIANTE MATRIZ DE CONFUSIÓN			
MUESTRAS TRES			
Matriz de Confusión		TRUE POSITIVE	FALSE POSITIVE
		28	1
		Numero de TP	Numero de FP
		FALSE NEGATIVE	TRUE NEGATIVE
		4	2
		Numero de FN	Numero de TN
Exactitud (Accuracy)	$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$		
	$Accuracy = \frac{28 + 2}{28 + 2 + 1 + 4} = 0.82$		
Porcentaje de Aprendizaje	82.9%		

Tabla 50: Evaluación de Matriz de Confusión Muestra 3
Fuente: Elaboración Propia

En la figura 22 se detalla la comparación de niveles de exactitud de cada muestra teniendo diferentes porcentajes.

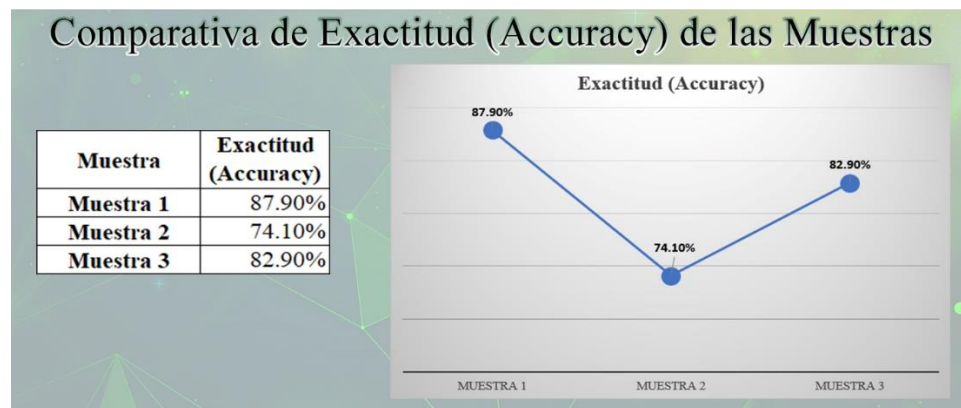


Figura 22: Comparativa de Exactitud (Accuracy) de las Muestras 1,2 y 3
Fuente: Elaboración Propia

Evaluado el modelo de detección de viviendas en conjunto de las muestras uno dos y tres se determina tener 84.3% de exactitud. En la siguiente tabla 51 se define el total

de las variables de la matriz de confusión de cada muestra y remplazando la fórmula de exactitud (Accuracy).

EVALUACIÓN DEL MODELO MEDIANTE MATRIZ DE CONFUSIÓN													
CONJUNTO DE LAS MUESTRAS UNO DOS Y TRES													
Matriz de Confusión	<table border="1"> <tr> <td>TRUE POSITIVE</td> <td>FALSE POSITIVE</td> </tr> <tr> <td>122</td> <td>7</td> </tr> <tr> <td>Numero de TP</td> <td>Numero de FP</td> </tr> <tr> <td>FALSE NEGATIVE</td> <td>TRUE NEGATIVE</td> </tr> <tr> <td>20</td> <td>4</td> </tr> <tr> <td>Numero de FN</td> <td>Numero de TN</td> </tr> </table>	TRUE POSITIVE	FALSE POSITIVE	122	7	Numero de TP	Numero de FP	FALSE NEGATIVE	TRUE NEGATIVE	20	4	Numero de FN	Numero de TN
	TRUE POSITIVE	FALSE POSITIVE											
	122	7											
	Numero de TP	Numero de FP											
	FALSE NEGATIVE	TRUE NEGATIVE											
20	4												
Numero de FN	Numero de TN												
Exactitud (Accuracy)	$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$												
	$Accuracy = \frac{122 + 4}{122 + 4 + 7 + 20} = 0.84$												
Porcentaje de Aprendizaje	84.3%												

Tabla 51: Evaluación de Matriz de confusión en conjunto de las muestras

Fuente: Elaboración Propia

4.4 RESULTADOS DE LA INVESTIGACIÓN

En la tabla 52 se plasma todos los resultados presentados de cada muestra haciendo mención, de las imágenes de entrada que conforma el ortomosaico, la cantidad de viviendas detectadas por el modelo de detección, las viviendas que están en cada muestra computadas, las viviendas no detectadas por el modelo y dando a conocer el nivel de exactitud del modelo de cada muestra y centrando todo en un nivel total.

Comparativa de Vivienda Detectadas contra Viviendas del Ortomosaico					
Numero de Muestras	Entrada de Imágenes	Viviendas del Ortomosaico	Viviendas detectadas (TP)	Viviendas No detectadas	Exactitud (Accuracy)%
Muestra 1	84	87	76	11	87.9%
Muestra 2	46	24	18	6	74.1%
Muestra 3	105	39	28	11	82.9%
Total	235	150	122	28	84.3%
Observaciones Por Muestra					

Muestra 1: En la muestra uno se presenta 87 viviendas computables, de las cuales aplicando el modelo de detección en plataforma ArcGisPro, se tiene como resultado TP viviendas detectadas 76 teniendo una diferencia de 11 viviendas no detectadas las cuales no se detectaron por motivo de patrones distintos y a su vez también detectando vehículos, sombrilla de motocicletas, vegetación y aguas residuales. Para más detalle visualizar la tabla 45

Muestra 2: En la muestra uno se presenta 24 viviendas computables, de las cuales aplicando el modelo de detección en plataforma ArcGisPro, se tiene como resultado TP viviendas detectadas 18 teniendo una diferencia de 6 viviendas no detectadas las cuales no se detectaron por motivo de patrones distintos y a su vez también detectando vehículos, sombrilla de motocicletas y vegetación. Para más detalle visualizar la tabla 46

Muestra 3: En la muestra uno se presenta 39 viviendas computables, de las cuales aplicando el modelo de detección en plataforma ArcGisPro, se tiene como resultado TP viviendas detectadas 28 teniendo una diferencia de 11 viviendas no detectadas las cuales no se detectaron por motivo de patrones distintos. Para más detalle visualizar la tabla 47

Tabla 52: *Comparación de Viviendas Detectadas con Viviendas del Ortomosaico*

Fuente: *Elaboración Propia*

CAPITULO V

5. CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES

La hipótesis de la presente tesis era El Modelo Fotogramétrico Deep Learning permite la detección de viviendas en los predios de la Zona Central del Municipio de Cobija. Que es evidente que el modelo fotogramétrico Deep Learning si detecta las viviendas en un porcentaje aceptable. Dentro del proceso del levantamiento fotogramétrico se concretó una altura ideal a función de la resolución espacial y se estableció un ortomosaico de buena calidad para proceder a la inclusión Deep Learning que se establece desde la recolección de los data sets, extracción de patrones y finalizando con la detección, llegando a la conclusión que el modelo si detecta las viviendas de cada predio, pero dentro de cada muestra hay deferentes patrones que cubre el techado de cada vivienda, mucha densidad de vegetación y la diversidad de materiales de construcción de los techados haciendo que el modelo detecte parte de la vivienda o en su defecto detectando automóviles, sombrillas de las motocicletas vegetación o aguas residuales.

Con el levantamiento fotogramétrico con drones, desde el análisis y estudio de la región a aplicar la investigación, se captura y procesa las imágenes, para hallar la altura ideal a base de la resolución espacial (GSD), para cada muestra obtenida, en la muestra 1 a 77m. y GSD 0.018m/px., muestra 2 a 76m GSD 0.017m/px y muestra 3 a 70m GSD 0.0164 m/px. para lograr obtener un ortomosaico de calidad donde se visualizan las viviendas del Municipio de Cobija.

En cuanto al procesamiento de imágenes, se obtuvo 235 imágenes procesadas en el software PIX4D, en un sistema de coordenadas WGS 1994 UTM Zona 19 como salida un ortomosaico 2D y su modelo digital de elevación (DEM).

Se Diseña una metodología para la inclusión de Deep Learning considerando desde la adquisición de data sets, extracción de patrones, detección y evaluación para establecer un modelo de detección de viviendas en el centro del Municipio de Cobija.

Con el modelo diseñado para la detección de viviendas, se obtuvo resultados aceptables teniendo de cada muestra diferentes resultados medidos mediante la matriz de confusión y aplicando la fórmula de Exactitud (Accuracy), la muestra 1 a un 87.9%, muestra

2 a un 74.1% y muestra 3 a un 82.9%, y en conjunto el modelo aplicado a todas las muestras tuvo un resultado de 84.3% logrando detectar un total de 122 viviendas.

Una de las dificultades del modelo de detección de viviendas, se establece en la densidad vegetal que cubre partes de las viviendas y deteriorando un patrón establecido del material de construcción y confundiendo ese patrón al modelo haciéndolo no detectar o detectar una falsa vivienda. La otra dificultad consiste en que las viviendas están construidas de diferentes materiales o deteriorado y se es difícil de concretar un patrón confundiendo la detección, incluso con automóviles o sombrillas de motocicletas.

5.2 RECOMENDACIONES

La presente investigación se ha llegado utilizar la aplicación de la fotogrametría con drones e inclusión Deep Learning, para la detección de viviendas en el centro del Municipio de Cobija, para futuras investigaciones e incursiones en estas tecnologías se recomienda diseñar una red neuronal convolucional para la detección de baches dentro del Municipio de Cobija para la prevención de accidentes viales.

Teniendo en cuenta el fallo de patrones por la densa vegetación sobre las viviendas se recomienda despejar los techos o terrazas de las viviendas para definir un patrón, se espera que disminuya el número de falsos positivos y falsos negativos de las diferentes muestras que abarca las viviendas en los predios de la zona central del Municipio de Cobija.

Se recomienda estandarizar los techados o terrazas mediante el ente regulador de viviendas (Catastro) del Municipio de Cobija, para poder establecer un patrón de buena calidad y generar un data set más preciso que ayude a detectar las viviendas.

De ser posibles, se recomienda mayor investigación a la detección de viviendas en el Municipio de Cobija para ir formando una Ciudad Smart y tener un control y supervisión de las modificaciones o construcción que se pretenda dentro de los predios. En caso de necesitar más resolución de las imágenes ortomosaico se recomienda utilizar un dron con mayor calidad en imagen.

6 GLOSARIO

A

Accuracy(Exactitud)

Fracción de predicciones que se realizaron correctamente en un modelo de clasificación. En la clasificación de clases múltiples.

B

Batch Size

Número de ejemplos que hay en un lote. Por ejemplo, el tamaño del lote de SGD es 1, mientras que el de un minilote suele ser entre 10 y 1,000. Por lo general, se fija el tamaño del lote durante el entrenamiento y la inferencia; sin embargo, TensorFlow permite tamaños de lotes dinámicos.

D

Data Sets

Colección de muestras.

M

Matriz de Confusión

Tabla de $N \times N$ que resume el nivel de éxito de las predicciones de un modelo de clasificación; es decir, la correlación entre la etiqueta y la clasificación del modelo. Un eje de una matriz de confusión es la etiqueta que el modelo predijo; el otro es la etiqueta real. N representa el número de clases.

S

Softmax

Función que proporciona probabilidades para cada clase posible en un modelo de clasificación de clases múltiples. Las probabilidades suman exactamente.

T

Training(Entrenamiento)

Proceso de determinar los parámetros ideales que conforman un modelo.

7 REFERENCIA BIBLIOGRAFÍA

- Aerials Insights. (10 de Enero de 2018). Aerials Insights. Recuperado el 20 de Febrero de 2019, de <http://www.aerial-insights.co/blog/fotogrametria-con-drones/>
- Alcalá, u. d. (2019). Máster en Deep Learning. Obtenido de Máster en Deep Learning: <https://master-deeplearning.com/objetivos-deep-learning-2019/>
- Apaza, M. A. (2018). FUSIÓN DE DATOS PARA SEGMENTACIÓN SEMÁNTICA EN APLICACIONES URBANAS DE . Lima: Universidad Catolica del Peru.
- ArcGis. (2 de 2 de 2019). ArcGis. Obtenido de <http://desktop.arcgis.com/es/arcmap/10.3/manage-data/raster-and-images/what-is-raster-data.htm>
- Arias, F. G. (2006). El proyecto de Investigación. Caracas: Episteme.
- Arriola, S. (2018). Methodology to Determine Dynamic Displaced Soil Volume Through UAS Photogrammetry. Costa Rica Institute of Tecnology, 78.
- ASPRS. (s.f.). www.asprs.org. Obtenido de <https://www.asprs.org/organization/what-is-asprs.html#0>
- Cabello, M. R. (28 de Febrero de 2018). www.slideshare.net. Obtenido de <https://www.slideshare.net/ManuelRodrigoCabello/deep-learning-python-c-y-azure>
- Calvo, D. (20 de Julio de 2017). Red Neuronal CNN. Obtenido de Red Neuronal CNN Clavijo: <http://www.diegocalvo.es/red-neuronal-convolucional/>
- Clavijo. (2017). Red Neuronal Convolucional. CNN-IBM, 28.
- Comment, L. A. (3 de Agosto de 2018). [ligdigonzalez](http://ligdigonzalez.com). Obtenido de [ligdigonzalez: http://ligdigonzalez.com/todo-sobre-aprendizaje-supervisado-en-machine-learning/](http://ligdigonzalez.com/todo-sobre-aprendizaje-supervisado-en-machine-learning/)
- Equipo de Expertos Universidad Internacional de Valencia. (21 de Marzo de 2018). VIU Universidad Internacional de Valencia. Recuperado el 14 de Febrero de 2019, de <https://www.universidadviu.com/que-es-un-dron-y-como-funciona/>
- Fallas, G. (2018). Desarrollo de un Sistema de software para estimar la biomasa de un bosque tropical utilizando imágenes aéreas y modelos 3D. Instituto Tecnologia de Costa Rica, 79.
- geodrone. (5 de Julio de 2019). videos drone. Obtenido de videos drone: <https://videodrone.fi/geodrone-x4l/>
- Geoportal. (2018). Metadatos de Información Geografica. Geoportal, 9.
- Grapsas, T. (26 de Febrero de 2019). rockcontent. Obtenido de rockcontent: <https://rockcontent.com/es/blog/deep-learning/>

- Gutierrez Palacios, J. (15 de Enero de 2016). CARTOMAP. Obtenido de cartomap.cl/utfsm/apuntes%20sig.pdf
- Hernández Sampieri, R., Fernández Collado, C., & Baptista Lucio, P. (2010). Metodología de la Investigación. Mexico D.F.: Mc Graw Hill.
- IBERDROLA. (7 de Agosto de 2017). IBERDROLA. Recuperado el 14 de Febrero de 2019, de ticeat: <https://www.ticeat.com/educacion/que-es-el-aprendizaje-profundo-o-deep-learning/>
- ilifebelt. (6 de 2 de 2018). deep learning. Obtenido de deep learning: <https://ilifebelt.com/aplicaciones-del-deep-learning/2018/02/>
- James, Y. T. (2019). Introducción a TensorFlow Deep Learning. Most.
- Jaramillo, E. F. (2019). Sistema inteligente para medición de areas a traves del procesamiento de imagenes obtenidas por uav's. Universidad del Azuay, 22.
- Linder, W. (2018). Digital photogrammetry a practical course. Berlin: Springer.
- Mining.com. (12 de Enero de 2017). How a sensor gimbal improves efficiency of an aerial mapping mission. Obtenido de <http://www.mining.com/web/how-a-sensor-gimbal-improves-efficiency-of-an-aerial-mapping-mission/>
- Mundo Contact. (13 de Septiembre de 2016). mundo contact. Recuperado el 14 de Febrero de 2019, de <https://mundocontact.com/8-aplicaciones-practicas-en-inteligencia-artificial/>
- Oca, I. M. (12 de Julio de 2015). Bolivia. Obtenido de Bolivia: <https://www.bolivia.com/geografiadebolivia/cap5.htm>
- Olaya, V. (16 de Octubre de 2014). Sistemas de Información Geográfica. Obtenido de https://www.icog.es/TyT/files/Libro_SIG.pdf
- Pérez, M. H. (19 de Julio de 2019). Retina. Obtenido de Retina: https://retina.elpais.com/retina/2019/07/16/tendencias/1563278002_625122.html
- Quiros, J. A. (2018). Diseño de un sistema de cuantificación automática de biomasa basado en procesamiento de imágenes y fotogrametría con vehículos aéreos no tripulados. Tec Tecnológico de Costa Rica, 91.
- Quiros, J. A. (2019). Procesamiento de imágenes y fotogrametría con VANTs. Tecnológico de Costa Rica, 78.
- R. Vizcaya Cárdenas, J. M. (2017). Desempeño de una Red Neuronal Convolutiva para Clasificación de señales de Tránsito Vehicular. Mexico: Sistemas Computacionales.
- Rosales, J. j. (2017). Extracción de conocimiento mediante la aplicación de inteligencia artificial a la información espacial. Mapping, 8.

- Rossum, G. v. (Octubre de 2017). python argentina. Obtenido de python argentina:
<http://www.python.org.ar/>
- Sitiobigdata. (19 de Enero de 2019). sitiobigdata. Obtenido de
<https://sitiobigdata.com/2019/01/19/machine-learning-metrica-clasificacion-parte-3/>
- T Luhmann, S. R. (2016). Fotogrametria de Distancia Corta. Escocia, Reino Unido: Whittles Publishing.
- Tamayo y Tamayo, M. (2008). El proceso de la Investigación Científica. Mexico D.F.: Limusa.
- TensorBoard. (21 de Octubre de 2019). TensorBoard. Obtenido de TensorBoard:
<https://guru99.es/tensorboard-tutorial/>
- TensorFlow. (9 de Septiembre de 2019). TensorFlow. Obtenido de TensorFlow:
<https://www.tensorflow.org/>
- Velázquez Olivera, A. (8 de Septiembre de 2017). El mundo de los drones. Cienciorama, 2-4.

8 ANEXOS

8.1 A- ARBOL DE PROBLEMAS

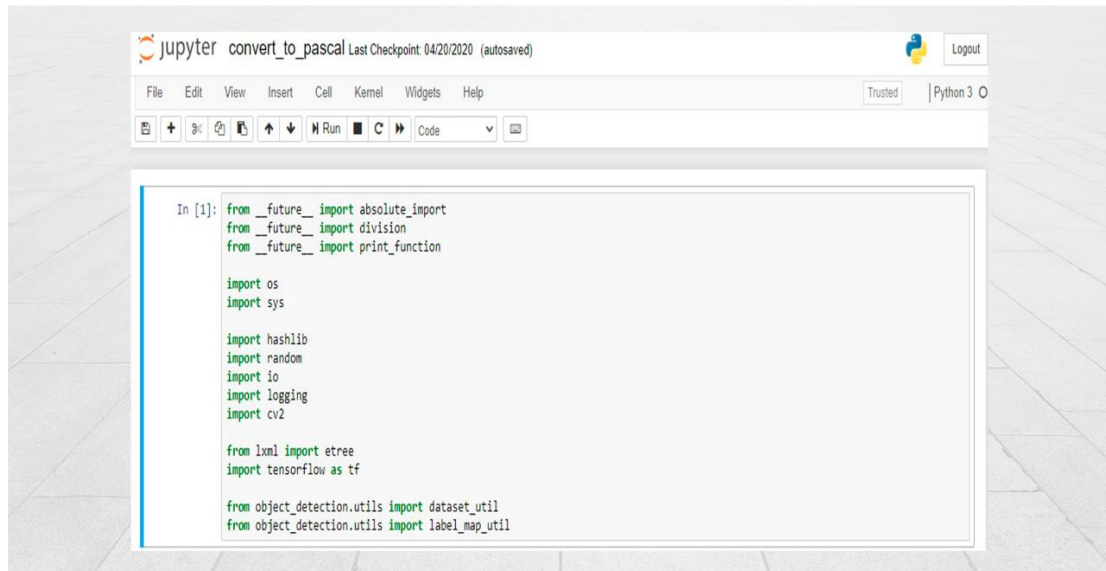


Figura 23: Árbol de Problemas
Fuente: Elaboración Propia

8.2 B- MATRIZ DE CONSISTENCIA

PROBLEMA	HIPOTESIS	OBJETIVO	VARIABLES	INDICADORES	METODOLOGÍA
¿Cómo detectar las viviendas de los predios de la zona central del Municipio de Cobija a través de Tecnología de Redes Neuronales?	El Modelo Fotogramétrico Deep Learning permite la detección de vivienda en los predios de la Zona Central del Municipio de Cobija.	Diseñar el Modelo Fotogramétrico Deep Learning para la detección de vivienda en los predios de la Zona Central del Municipio de Cobija	V.I	-Desempeño del Modelo	Tipo de Investigación La investigación es de tipo Aplicada
			-Deep Learning.		
			-Fotogrametría	-Procesamiento de Imágenes	Técnicas y procedimientos Se evaluará los resultados de salida que proporcione el software ArcGis mediante la matriz de confusión
			V.D.	Tiempo de ejecución	
			- detección de viviendas	Proceso de abstracción	

8.3 C- CODIGO-JUPITER-NOTEBBOK CONVERTIDOR DE PASCAL



```
In [1]: from __future__ import absolute_import
from __future__ import division
from __future__ import print_function

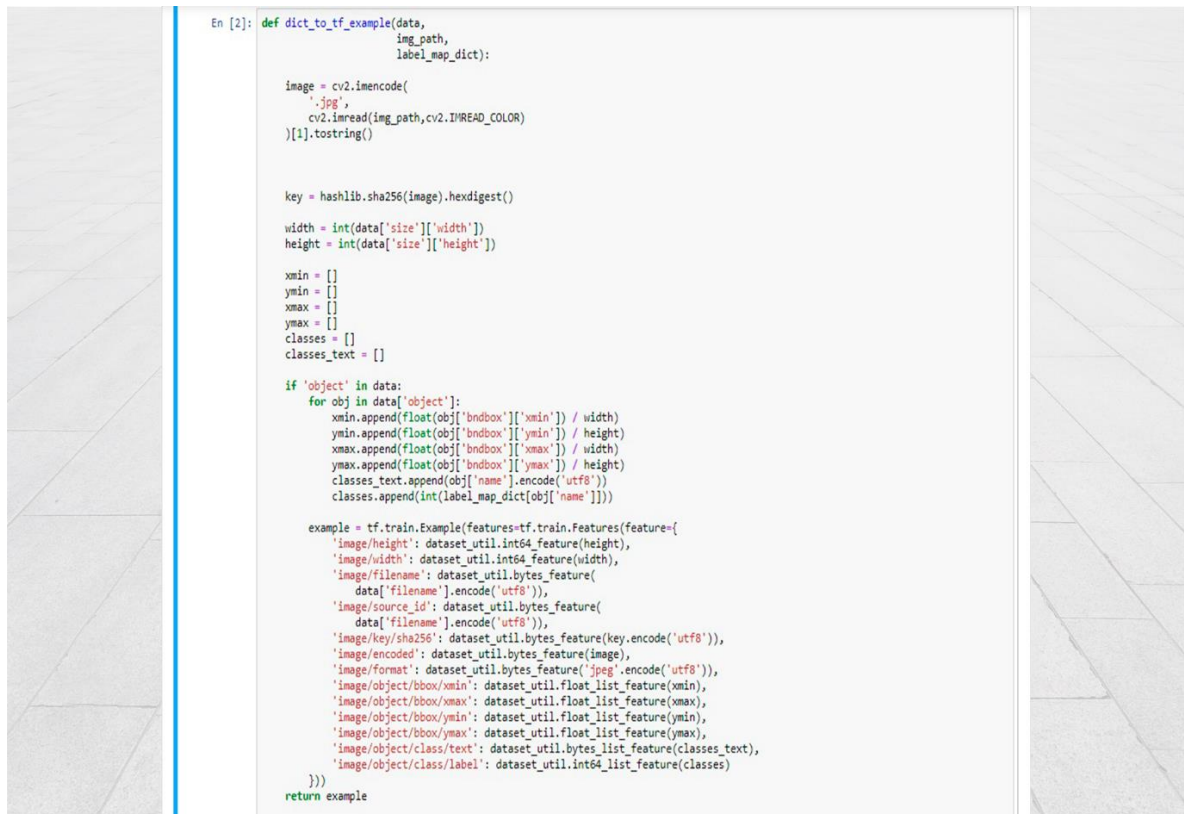
import os
import sys

import hashlib
import random
import io
import logging
import cv2

from lxml import etree
import tensorflow as tf

from object_detection.utils import dataset_util
from object_detection.utils import label_map_util
```

Figura 24: Convertidor de Pascal Casilla 1
Fuente: Elaboración Propia



```
En [2]: def dict_to_tf_example(data,
                               img_path,
                               label_map_dict):

    image = cv2.imencode(
        '.jpg',
        cv2.imread(img_path, cv2.IMREAD_COLOR)
    )[1].tostring()

    key = hashlib.sha256(image).hexdigest()

    width = int(data['size']['width'])
    height = int(data['size']['height'])

    xmin = []
    ymin = []
    xmax = []
    ymax = []
    classes = []
    classes_text = []

    if 'object' in data:
        for obj in data['object']:
            xmin.append(float(obj['bbox']['xmin']) / width)
            ymin.append(float(obj['bbox']['ymin']) / height)
            xmax.append(float(obj['bbox']['xmax']) / width)
            ymax.append(float(obj['bbox']['ymax']) / height)
            classes_text.append(obj['name'].encode('utf8'))
            classes.append(int(label_map_dict[obj['name']]))

    example = tf.train.Example(features=tf.train.Features(feature={
        'image/height': dataset_util.int64_feature(height),
        'image/width': dataset_util.int64_feature(width),
        'image/filename': dataset_util.bytes_feature(
            data['filename'].encode('utf8')),
        'image/source_id': dataset_util.bytes_feature(
            data['filename'].encode('utf8')),
        'image/key/sha256': dataset_util.bytes_feature(key.encode('utf8')),
        'image/encoded': dataset_util.bytes_feature(image),
        'image/format': dataset_util.bytes_feature('jpeg'.encode('utf8')),
        'image/object/bbox/xmin': dataset_util.float_list_feature(xmin),
        'image/object/bbox/xmax': dataset_util.float_list_feature(xmax),
        'image/object/bbox/ymin': dataset_util.float_list_feature(ymin),
        'image/object/bbox/ymax': dataset_util.float_list_feature(ymax),
        'image/object/class/text': dataset_util.bytes_list_feature(classes_text),
        'image/object/class/label': dataset_util.int64_list_feature(classes)
    }))

    return example
```

Figura 25: Convertidor de Pascal Casilla 2
Fuente: Elaboración Propia

```

In [3]: data_dir = 'C:/Users/Mallea/Desktop/Teonsor/models/research/object_detection/casaV1/Chips_casas'
mapfile = os.path.join(data_dir, 'map.txt')

# Read the mapfile:
with open(mapfile, 'r') as fp:

    mp = (
        st.strip().split(' ')
        for st in fp.readlines()
    )

    fixpath = lambda x: os.path.abspath(os.path.join(data_dir, x))
    mp = [
        (fixpath(x[0]), fixpath(x[1]))
        for x in mp
    ]

label_map_dict = label_map_util.get_label_map_dict('C:/Users/Mallea/Desktop/Teonsor/models/research/object_detection/casaV1/Chip

```

WARNING:tensorflow:From C:/Users/Mallea/Desktop/Teonsor/models/research/object_detection/utils/label_map_util.py:138: The name tf.gfile.GFile is deprecated. Please use tf.io.gfile.GFile instead.

Figura 26: Convertidor de Pascal Casilla 3
Fuente: Elaboración Propia

```

In [4]: train_writer = tf.python_io.TFRecordWriter('C:/Users/Mallea/Desktop/Teonsor/models/research/object_detection/casa_cobija/Imagenes
test_writer = tf.python_io.TFRecordWriter('C:/Users/Mallea/Desktop/Teonsor/models/research/object_detection/casa_cobija/Imagenes

records = []

for img_path, label_path in mp:
    with tf.gfile.GFile(label_path, 'r') as fid:
        xml_str = fid.read()
        xml = etree.fromstring(xml_str)
        data = dataset_util.recursive_parse_xml_to_dict(xml)['annotation']

        tf_example = dict_to_tf_example(data, img_path, label_map_dict)
        records.append(tf_example.SerializeToString())

percent_test = 0.05 # 5% test set
# Number of records in test set
N = int(round(len(records) * percent_test))

# Shuffle them
random.shuffle(records)

# Split
test, train = records[:N], records[N:]

for r in train:
    train_writer.write(r)
train_writer.close()
if os.path.exists('C:/Users/Mallea/Desktop/Teonsor/models/research/object_detection/casa_cobija/Imagenes_Chipset/train.record'):
    print ("Training set written")

for r in test:
    test_writer.write(r)
test_writer.close()
if os.path.exists('C:/Users/Mallea/Desktop/Teonsor/models/research/object_detection/casa_cobija/Imagenes_Chipset/test.record'):
    print ("Test set written")

```

C:/Users/Mallea/Desktop/Teonsor/models/research/object_detection/utils/dataset_util.py:79: FutureWarning: The behavior of this method will change in future versions. Use specific 'len(elem)' or 'elem is not None' test instead.
if not xml:

Training set written
Test set written

Figura 27: Convertidor de Pascal Casilla 4
Fuente: Elaboración Propia

8.4 D- CODIGO DEL MODELO FOTOGRAMETRICO DEEP LEARNING DE DETECCIÓN DE VIVIENDAS

```

model {
  faster_rcnn {
    num_classes: 1
    image_resizer {
      keep_aspect_ratio_resizer {
        min_dimension: 600
        max_dimension: 1024
      }
    }
    feature_extractor {
      type: "faster_rcnn_inception_v2"
      first_stage_features_stride: 16
    }
    first_stage_anchor_generator {
      grid_anchor_generator {
        height_stride: 16
        width_stride: 16
        scales: 0.25
        scales: 1.0
        scales: 2.0
        aspect_ratios: 0.5
        aspect_ratios: 1.0
        aspect_ratios: 2.0
      }
    }
    first_stage_box_predictor_conv_hyperparams {
      op: CONV
      regularizer {
        l2_regularizer {
          weight: 0.0
        }
      }
    }
    initializer {
      truncated_normal_initializer {
        stddev: 0.01
      }
    }
  }
  first_stage_nms_score_threshold: 0.0
  first_stage_nms_iou_threshold: 0.7
  first_stage_max_proposals: 300
  first_stage_localization_loss_weight: 2.0
  first_stage_objectness_loss_weight: 1.0
  initial_crop_size: 14
  maxpool_kernel_size: 2
  maxpool_stride: 2
  second_stage_box_predictor {
    mask_rcnn_box_predictor {
      fc_hyperparams {
        op: FC
        regularizer {
          l2_regularizer {
            weight: 0.0
          }
        }
      }
      initializer {
        variance_scaling_initializer {
          factor: 1.0
          uniform: true
          mode: FAN_AVG
        }
      }
      use_dropout: false
      dropout_keep_probability: 1.0
    }
  }
  second_stage_post_processing {
    batch_non_max_suppression {
      score_threshold: 0.0
      iou_threshold: 0.6
      max_detections_per_class: 100
      max_total_detections: 300
    }
    score_converter: SOFTMAX
  }
  second_stage_localization_loss_weight: 2.0
  second_stage_classification_loss_weight: 1.0
  train_config {
    batch_size: 4
    data_augmentation_options {
      random_horizontal_flip {
      }
    }
  }
}
optimizer {
  momentum_optimizer {
    learning_rate {
      manual_step_learning_rate {
        initial_learning_rate: 0.0002
        schedule {
          step: 1
          learning_rate: .0002
        }
        schedule {
          step: 900000
          learning_rate: .00002
        }
        schedule {
          step: 1200000
          learning_rate: .000002
        }
      }
      momentum_optimizer_value: 0.9
    }
    use_moving_average: false
  }
  gradient_clipping_by_norm: 10.0
  fine_tune_checkpoint: "C:/Users/Mallea/Desktop/Tensor/models/research/object_detection/casa_cobija/Imagenes_Chipset/faster_rcnn_inception_v2_coco_2018_01_28/model.ckpt"
  from_detection_checkpoint: true
  num_steps: 4381
}
train_input_reader {
  label_map_path: "C:/Users/Mallea/Desktop/Tensor/models/research/object_detection/casa_cobija/Imagenes_Chipset/buildin.pbtxt"
  tf_record_input_reader {
    input_path: "C:/Users/Mallea/Desktop/Tensor/models/research/object_detection/casa_cobija/Imagenes_Chipset/train.record"
  }
}
eval_config {
  num_examples: 67
  max_evals: 10
  use_moving_averages: false
}
eval_input_reader {
  label_map_path: "C:/Users/Mallea/Desktop/Tensor/models/research/object_detection/casa_cobija/Imagenes_Chipset/buildin.pbtxt"
  shuffle: false
  num_readers: 1
  tf_record_input_reader {
    input_path: "C:/Users/Mallea/Desktop/Tensor/models/research/object_detection/casa_cobija/Imagenes_Chipset/test.record"
  }
}

```

Figura 28: Código del Modelo Fotogramétrico Deep Learning para la detección de viviendas

Fuente: Elaboración Propia