Detección de Fenómenos Territoriales en Santiago de Cali a partir de imágenes VHR

Fernando Cardona Hansen

Nota de Aceptación

Certificamos que el presente Trabajo de Grado Satisface, en alcances y calidad, todos los requisitos que demanda un Trabajo de Grado de Maestría.

GERARDO MAURICIO SARRIA MONTEMIRANDA
Director

VALENTINA CORCHUELO GUZMÁN
Jurado

HERNÁN DARIO VARGAS CARDONA
Jurado

Aprobado en cumplimiento de los requisitos exigidos por la Pontificia Universidad Javeriana Cali, para optar el título de Magister en Ciencia de Datos.

HERNÁN CAMILO ROCHA NIÑO Ph. D.
Decano Facultad de Ingeniería y Ciencias

JUAN CARLOS MARTÍNEZ ARIAS
Director Posgrados de Ingeniería y Ciencias

Santiago de Cali, 01 de febrero de 2024

Acta de Correcciones al Documento de Trabajo de Grado

Santiago de Cali, 01 de febrero de 2024

Autor: Fernando Cardona Hansen

Título del Trabajo de Grado: "Detección de Fenómenos Territoriales en Santiago de Cali a partir de imágenes VHR"

Director: Gerardo Mauricio Sarria Montemiranda

Como indica el artículo 2.13 de las Directrices para Trabajo de Grado de Maestría, he verificado que el estudiante indicado arriba ha implementado todas las correcciones que los Jurados del Proyecto de Trabajo de Grado definieron que se efectuaran, como consta en el Acta de Evaluación correspondiente.

Firma del Director del Trabajo de Grado

Santiago de Cali, 09 de diciembre del 2023

Doctora
Gloría Inés Alvarez V.
Directora Maestría en Ciencia de Datos
Facultad de Ingeniería y Ciencias
Pontificia Universidad Javeriana de Cali

Asunto: Presentación para evaluación del proyecto aplicado

Cordial Saludo,

Con el fin de cumplir con los requisitos exigidos por la Universidad para optar por el título de Magíster en Ciencia de Datos, nos permitimos presentar a su consideración el proyecto denominado “Detección de fenómenos territoriales en Santiago de Cali usando imágenes VHR”, el cual fue realizado por el estudiante Fernando Cardona Hansen con código 8976071 perteneciente a la Maestría en Ciencia de Datos, bajo la dirección de Gerardo Mauricio Sarria Montemiranda.

El suscrito director del Proyecto Aplicado autoriza para que se proceda a hacer la evaluación de este proyecto, toda vez que ha revisado cuidadosamente el documento y avala que ya se encuentra listo para ser presentado y sustentado oficialmente.

Atentamente,

FERNANDO CARDONA HANSEN

C.C. 94.501.175 de Cali

GERARDO MAURICIO SARRIA MONTEMIRANDA

C.C. 94.495.699 de Cali

Documentación anexa:
Resumen del Proyecto Aplicado en formato digital (máximo 1 página).
Una copia digital (PDF) del documento del proyecto aplicado

ALCALDÍA DE
SANTIAGO DE CALI
DEPARTAMENTO ADMINISTRATIVO
DE TECNOLOGÍAS DE LA INFORMACIÓN
Y LAS COMUNICACIONES

Al contestar por favor cite estos datos:
Radicado No.: **20234134020000821**
Fecha: **2023-12-27**
TRD: **4134.020.13.1.953.000082**
Rad. Padre: **20234173010238653 2**

FERNANDO CARDONA HANSEN
Estudiante Maestría Ciencia De Datos
Universidad Javeriana Cali

Asunto: Respuesta a solicitud de acceso a imágenes satelitales de Santiago de Cali

Cordial saludo,

El Departamento Administrativo de Tecnologías de la Información y las Comunicaciones DATIC, autoriza a Fernando Cardona Hansen, identificado con número de cédula 94.501.175 de Cali, a hacer uso de imágenes aéreas y satelitales de Santiago de Cali en custodia de DATIC, específicamente el uso de la fuente de información denominada Ortofotomosaico Cali (2020), propiedad de la Alcaldía de Cali, a partir del 15 de julio de 2023 y hasta el 29 de diciembre de 2023, para ser tratadas con fines académicos en el desarrollo del proyecto de trabajo de grado: “Detección de fenómenos territoriales en Santiago de Cali a partir de imágenes VHR.”
Igualmente, autoriza hacer uso de la información contenida en las imágenes para elaboración de una base de datos etiquetadas de acuerdo con la presencia o ausencia del fenómeno de interés seleccionado.

Atentamente,

ROGER GONZÁLEZ PÉREZ
Subdirector
Subdirección de Tecnología Digital

Elaboró: Carlos Andres Torres Ricaurte – Contratista

FICHA RESUMEN
PROYECTO APLICADO – MAESTRÍA EN CIENCIA DE DATOS

TÍTULO: Detección de fenómenos territoriales en Santiago de Cali a partir de imágenes VHR

1. ÁREA DE TRABAJO: Detección de objetos en imágenes
2. TIPO DE PROYECTO (Aplicado, Innovación, Investigación): Aplicado
3. ESTUDIANTE(S): Fernando Cardona Hansen
4. CORREO ELECTRÓNICO: fernando.cardona.hansen@gmail.com
5. DIRECCIÓN Y TELÉFONO: Carrera 50 # 5-173. URESA 49-202 / 3183904372
6. DIRECTOR: Gerardo Mauricio Sarria Montemiranda
7. VINCULACIÓN DEL DIRECTOR: Profesor titular Facultad de Ingeniería y Ciencias Universidad Javeriana Cali
8. CORREO ELECTRÓNICO DEL DIRECTOR: gsarria@javerianacali.edu.co
9. CO-DIRECTOR (Si aplica): N/A
10. GRUPO O EMPRESA QUE LO AVALA: Departamento Administrativo de Tecnologías de la Información y Comunicaciones DATIC – Alcaldía de Santiago de Cali
11. OTROS GRUPOS O EMPRESAS: N/A
12. PALABRAS CLAVE (al menos 5): redes neuronales convolucionales, modelos YOLO de detección de objetos en imágenes, fenómenos territoriales, asentamiento informal.
13. FECHA DE INICIO: febrero 2023
14. FECHA DE FINALIZACIÓN: 07 de diciembre de 2023
15. RESUMEN: Las imágenes satelitales son una fuente de datos alternativa en proyectos de ciencia de datos adelantados dentro del sector público en Colombia. Los asentamientos informales son fenómenos propios del crecimiento urbano caracterizados por la concurrencia de condiciones físicas y sociales deficitarias como población en condiciones de vulnerabilidad y falta de acceso a infraestructura de servicios públicos básicos. El uso de imágenes satelitales para identificar y mapear eventos de interés territorial como los asentamientos informales, constituye una alternativa para la implementación de acciones gubernamentales oportunas que sustituyan las medidas reactivas. El proyecto “Detección de fenómenos territoriales en Santiago de Cali usando imágenes VHR” tiene el propósito de desarrollar un modelo de aprendizaje profundo para la detección de asentamientos informales en imágenes satelitales del perímetro urbano de Cali, que contribuya a identificación de este fenómeno por parte de la administración municipal.

DETECCIÓN DE FENÓMENOS TERRITORIALES EN SANTIAGO DE CALI A PARTIR DE IMÁGENES VHR

Fernando Cardona Hansen
Código 8976071

Proyecto Aplicado para optar al título de
Magister en Ciencia de Datos

Director
Gerardo Mauricio Sarria Montemiranda

FACULTAD DE INGENIERÍA Y CIENCIAS
MAESTRÍA EN CIENCIA DE DATOS
SANTIAGO DE CALI, ENERO DE 2024

TABLA DE CONTENIDO

LISTA DE FIGURAS

LISTA DE TABLAS

LISTA DE ANEXOS

INTRODUCCIÓN

La tarea de detección de objetos en imágenes y video en tiempo real está en el centro de las aplicaciones que dan capacidad de conducción autónoma total a automóviles y aviones, así como de las aplicaciones de videovigilancia actuales. La detección de objetos es, a la vez, una tarea de clasificación y de ubicación de instancias de objetos para el que se han desarrollado múltiples modelos de aprendizaje profundo entrenados en conjuntos de imágenes de gran escala.

El proyecto aplicado *Detección de fenómenos territoriales en Santiago de Cali a partir de imágenes VHR* se incribe en esa corriente, como ejercicio pionero en el desarrollo de un detector de asentamientos informales que permita a la Alcaldía de Santiago de Cali diseñar actuaciones oportunas para el ordenamiento y la gestión del territorio.

Para la detección de fenómenos territoriales, como para la detección de cualquier otro tipo objeto terrestre, se necesitan datos. También entender los algoritmos por medio de los cuales la redes neuronales convolucionales operan mecanismos para identificar con precisión instancias objetos en imágenes y video.

En este documento el lector encontrará las dos cosas: primero, a manera de cimiento conceptual, el acervo de conocimiento clave para entender por qué las redes neuronales permiten el aprendizaje profundo; segundo, cómo se elaboró una base de datos con instancias de objetos etiquetados, denominados fenómenos territoriales, que permitieron entrenar modelos de detección de objetos en imágenes denominados YOLO (*You Only Look Onc*e). Un elemento final del documento reflexiona sobre los resultados obtenidos por el detector, evaluando el logro de los objetivos propuestos y proponiendo algunas alternativas viables para mejorar los resultados.

El proyecto aplicado *Detección de fenómenos territoriales en Santiago de Cali a partir de imágenes VHR* fue motivado, acelerado y materializado gracias al Departamento Administrativo de Tecnologías de la Información y Comunicaciones DATIC de la Alcaldía de Cali. Desde DATIC se orquestaron los proyectos de inversión denominados: *Implementación de un modelo de Big Data* e *Implementación de un modelo de inteligencia artificial en la Alcaldía de Santiago de Cali*, que viabilizaron casos de uso de analítica avanzada con servicios de computación en la nube, realizados por la administración municipal entre 2021 y 2023.

El equipo Big Data e IA de la subdirección de Tecnología Digital de DATIC, instaló en la Alcaldía de Cali el abordaje de los problemas de explotación masiva de datos como proyectos de ciencia de datos y generó los marcos de trabajo institucionales para incorporar en el ADN de la entidad

procedimientos para la formulación, implementación y despliegue de casos de uso que requieren computación en la nube para aplicar técnicas de big data, machine learning e inteligencia artificial enfocadas a evidenciar el valor público generado por las actuaciones gubernamentales.

En ese contexto, desde DATIC se gestionaron y disponibilizaron las fuentes de imágenes VHR de Santiago de Cali utilizadas para elaborar la base de datos con imágenes etiquetadas de fenómenos territoriales y se habilitaron las instancias de almacenamiento y procesamiento necesarios para correr modelos que exigen alta carga computacional. El nivel directivo de DATIC y especialmente el equipo de trabajo Big Data e IA, dieron un estímulo invaluable para este proyecto.

1. DEFINICIÓN DEL PROBLEMA

1.1 Planteamiento del Problema

De acuerdo con el Banco Interamericano de Desarrollo - BID, los asentamientos informales son territorios que se desarrollan en condiciones físicas y sociales deficitarias donde se concentra población urbana en condiciones de vulnerabilidad y la falta de acceso a servicios básicos como agua potable, saneamiento, recolección de residuos y transporte [1].

Si bien se trata de un fenómeno complejo que puede evidenciarse en los centros urbanos de Latinoamérica y el Caribe, particularmente en Colombia los asentamientos informales son el resultado de oleadas de migración de población rural afectada por el conflicto armado, y de la demanda insatisfecha de vivienda urbana, suplida generalmente mediante autoconstrucción y de manera no controlada por las autoridades locales [2].

En los estudios de base para la formulación de Plan Nacional de Desarrollo 2018 - 2022, se proyecta para el 2050 que el 86% de la población colombiana vivirá en los centros urbanos del país, lo cual representa un desafío para que la oferta de vivienda y la provisión de servicios urbanos, no supere los límites y la capacidad político administrativa de los municipios. Particularmente, se considera que la expansión urbana no planificada en las ciudades colombianas ocasiona dinámicas como: suburbanización; presión sobre recursos naturales y suelos con vocación productiva, y localización de asentamientos humanos en áreas expuestas a riesgos ambientales [3].

En agosto de 2022, un hecho que evidenció la magnitud del fenómeno de asentamientos informales en la ciudad de Santiago de Cali, fue el desalojo masivo de aproximadamente 1500 personas que ocupaban 30 hectáreas de terreno localizado en el corregimiento de Navarro [4]. El impacto social y político de esta actuación administrativa, que fue noticia en medios de comunicación nacionales, mostró que institucionalmente no se cuenta con mecanismos para prevenir o anticipar la ocurrencia de este fenómeno asociado al crecimiento urbano informal, actuando de manera reactiva cuando los conflictos relacionados con la ocupación del suelo son incontrolables.

A través de la plataforma LandViewer, se observa en fotos del sensor Landsat, como en el periodo comprendido entre mayo de 2021 y septiembre de 2022, el polígono correspondiente a la ubicación de la Hacienda Aldovea, en el corregimiento de Navarro, es ocupado por estructuras irregulares que constituyen un asentamiento informal.

Figura 1. Polígono del predio Aldovea, corregimiento Navarro de Santiago de Cali, 2021. Foto Lansat, 2022.

Figura 2. Polígono del predio Aldovea, corregimiento Navarro de Santiago de Cali, 2022. Foto Lansat, 2022.

Este fenómeno desafía la mirada de las entidades del sector público que por su misionalidad están encargadas de planificar el territorio y las dinámicas de expansión urbana, de prevenir y mitigar situaciones de riesgo natural, de promover planes y programas de vivienda social, de proveer infraestructura de servicios públicos, de actualizar el catastro, de preservar el medio ecológico, de controlar la ocupación del espacio público y de atender a las poblaciones vulnerables.

¿Cómo puede la ciencia de datos aportar en la solución de este problema?

El Departamento Nacional de Planeación - DNP, en alianza con el BID, realizó un proyecto piloto para crear una herramienta de inteligencia artificial basada en el análisis de imágenes satelitales que mapear los asentamientos informales en ciudades Colombianas; utilizando imágenes satelitales de alta resolución (VHR) de la ciudad de Barranquilla y con base en áreas demarcadas por personal experto de la administración municipal, el equipo de científicos de datos entrenó algoritmo de segmentación semántica de imágenes, para reconocer de forma automática las características de sectores urbanos con asentamientos informales. Desarrollaron el algoritmo

denominado MAIIA (Mapeo de Asentamiento Informales con Inteligencia Artificial), para generar y actualizar mapas precisos de la ubicación y extensión de asentamientos informales en ciudades colombianas, mediante el análisis de imágenes satelitales.

La hipótesis general de un proyecto de ciencia de datos es que el desarrollo de una herramienta de aprendizaje profundo basada en redes neuronales para clasificar asentamientos informales en imágenes satelitales, como la realizada por el BID y el DNP, le ofrece a la Alcaldía de Santiago de Cali un recurso de información para la atención integral y oportuna a las demandas de ocupación del suelo para el crecimiento urbano. La hipótesis particular de este proyecto para optar por el título de maestría en ciencia de datos, es que el desarrollo de una herramienta de inteligencia artificial basada en redes neuronales convolucionales, es funcional y eficiente para detectar y mapear los asentamientos informales dentro del perímetro de la ciudad de Santiago de Cali.

1.2 Formulación del Problema

A partir de la hipótesis planteada, se formulan las siguientes preguntas problema: ¿Qué tipo de modelos de inteligencia artificial basados en redes neuronales artificiales se pueden implementar para detectar y mapear los asentamientos informales que existen dentro del perímetro urbano de Santiago de Cali? ¿Por qué los métodos de aprendizaje profundo basados en redes neuronales convolucionales son apropiados para resolver tareas de detección y clasificación de objetos en imágenes satelitales? ¿Cuáles son las características de las imágenes y los requerimientos técnicos necesarios para disponer de un base de datos de entrenamiento y prueba del modelo?

¿Cuáles son las estrategias y técnicas apropiadas que pueden aplicarse para la detección de asentamientos informales en Santiago de Cali? ¿Cómo medir el desempeño de la herramienta para la detección, según las principales métricas de evaluación de modelos de aprendizaje profundo?

1.3 Justificación

El aprovechamiento de datos en el sector público tiene el propósito de evidenciar el impacto de los programas y proyectos de inversión en la prestación eficiente de servicios públicos, el mejoramiento de la calidad de vida de los ciudadanos y la eficacia administrativa a través de la optimización de los recursos disponibles.

El uso de fuentes de datos no tradicionales como las imágenes aéreas y satelitales para el análisis de eventos de interés a través de modelos de aprendizaje automático, ha mostrado resultados positivos como los alcanzados por el Departamento Nacional de Estadística - DANE en el cálculo del Índice de Pobreza Multidimensional censal, aplicado en secciones rurales del territorio

nacional donde no se realizó en censo nacional de población y vivienda de 2018 por dificultades de acceso geográfico y de conflicto armado [5].

Desarrollar un modelo de aprendizaje automático que ayude a la administración municipal de Cali a identificar eventos de interés que afectan el ordenamiento territorial usando imágenes satelitales, resulta útil en múltiples aspectos: primero, como insumo de información para la atención oportuna del evento mediante el despliegue de programas de atención, mitigación o control; segundo, como experiencia que convoque la confluencia de más recursos técnicos y humanos formados en ciencia de datos para el desarrollo de soluciones analíticas basadas en inteligencia artificial. Tercero, porque ayuda a identificar espacialmente las dinámicas de crecimiento urbano informal que presionan los recursos limitados de suelo disponible que generan problemáticas ambientales, sociales y espaciales en el territorio.

La Alcaldía de Santiago de Cali propuso como meta en el Plan de Desarrollo 2020 – 2023, formular un modelo de Inteligencia Artificial para implementar análisis de tipo predictivo basado en técnicas de aprendizaje automático, que brinde apoyo a la toma de decisiones y permita evidenciar el valor público generado a partir de sus actuaciones y servicios al ciudadano.

En ese sentido, la apuesta del organismo encargado de gestionar la transformación digital y proveer la infraestructura de información y las comunicaciones en la Alcaldía de Cali, DATIC, fue diseñar un modelo de inteligencia artificial, con el objetivo de resolver problemas estratégicos del territorio al tiempo que gestionar de manera eficiente, productiva y transparente los datos disponibles. Para ello, dispuso de una arquitectura tecnológica de servicios en la nube para analítica de datos y conformó un equipo de apoyo para el desarrollo e implementación de proyectos de analítica avanzada. En el mes de noviembre de 2022, DATIC realizó un ejercicio de prototipado que convocó personal experto en análisis de información geográfica de la entidad, en temas de gestión del territorio, para el desarrollo de prototipos de inteligencia artificial basados en análisis de imágenes satelitales y en respuesta automatizada para atención al ciudadano.

A partir de la meta propuesta en Plan de Desarrollo Municipal, de las capacidades técnicas y tecnológicas existentes en DATIC y de la identificación de personas con conocimiento experto en secretarías y departamentos administrativos relacionados con el desarrollo territorial, se vislumbró la motivación y oportunidad la formulación de un proyecto de inteligencia artificial basado en el análisis de imágenes satelitales aplicado a la detección de asentamientos informales en la ciudad de Cali.

2. OBJETIVOS DEL PROYECTO

2.1 Objetivo General

Desarrollar una herramienta de aprendizaje automático para detectar y mapear asentamientos informales dentro del perímetro urbano de la ciudad de Santiago de Cali.

2.2 Objetivos Específicos

- Elaborar una base de datos con imágenes aéreas y satelitales, etiquetadas de acuerdo con la presencia o ausencia de asentamientos informales.

- Entrenar modelos basados en redes neuronales para la detección de asentamientos informales en la ciudad de Cali.

- Evaluar la capacidad de detección de la herramienta en un conjunto de prueba, usando las métricas: área bajo la curva (AUC), exhaustividad (recall), precisión y F1-score.

3. MARCO TEÓRICO Y ANTECEDENTES

3.1 Marco Teórico

La Ciencia de Datos es un campo multidisciplinario que usa un conjunto de herramientas para extraer conocimiento de los datos, dando soporte a la toma de decisiones complejas. Entre las principales herramientas de la ciencia de datos se encuentran el aprendizaje de máquina (machine learning) y el aprendizaje profundo (deep learning) [6]. El machine learning es la ciencia (y el arte) de programar computadoras para que puedan aprender de los datos. Los ejemplos que utiliza el sistema para aprender se denominan conjunto de entrenamiento, y cada ejemplo de entrenamiento se denomina muestra [7].

El aprendizaje de máquina estudia los algoritmos que pueden aprender a realizar tareas sin instrucciones específicas, basándose en patrones descubiertos en los datos, mientras el deep learning estudia un tipo específico de aprendizaje automático llamado redes neuronales profundas.

Las redes neuronales profundas han sido empleadas para resolver problemas de clasificación y de detección de objetos, porque permiten reconocer patrones complejos a partir de mapas de características en tipos de datos no estructurados y, con base en el aprendizaje, predecir la presencia de las clases de interés en nuevos datos.

3.1.1 Neurona artificial

Al igual que una neurona cerebral que consta de Dendritas, las cuales actúan como canales de entrada de las señales provenientes del exterior hacia el Soma o cuerpo celular, y el Axón, que actúa como canal de salida, una neurona artificial consta de tres elementos: 1) unos valores de entrada (x) o parámetros con la información inicial, cada uno de los cuales está determinado por un peso (w) que modula la importancia de la información recibida; 2) una función de activación f(x) que realiza la suma ponderada de los valores de entrada; y 3) un valor de respuesta (*y*).

Adicionalmente, entre los valores de entrada de una neurona artificial se incluye un valor de sesgo (b) o Bias, que permite disparar la función de activación a 0 o 1 para garantizar la convergencia a un mínimo adecuado.

Una neurona artificial se puede representar bajo la ecuación $Y = f(w1x1 + w2x2\ldots + b)$

En el cuerpo de la neurona artificial, la suma ponderada de todos los valores de entrada pasan por la función de activación para distorsionar el valor de salida, añadiendo valores no lineales a la

estructura de la red neuronal y así encadenar de forma efectiva la computación de varias neuronas. Algunas funciones de activación existentes se denominan: Escalonada, Sigmoidal, Tangente Hiperbólica (*TANH*) y Unidad Rectificada Lineal (*ReLU*).

La Función de activación Escalonada fija para un valor de entrada mayor al umbral, un valor de salida (output) igual a 1, y por el contrario, para un valor de entrada inferior al umbral, un valor de salida (output) igual a 0. Se representa con la siguiente ecuación: F(x) = {0 for X < 0, 1 for X > 0}.

La Función de activación Sigmoide representa probabilidades en el rango de 0 a 1. Se representa con la siguiente ecuación: F(x) = σ(x) = 1/(1+e **-x).

En la función de activación Tangente Hiperbólica, el rango varía entre (-1) y (1), mientras que en la función de activación Rectificada Lineal (ReLu), es igual a 0 para todos los valores de entrada menores que 0 y es igual a (x) para todos los valores de (x) mayores o iguales a 0.

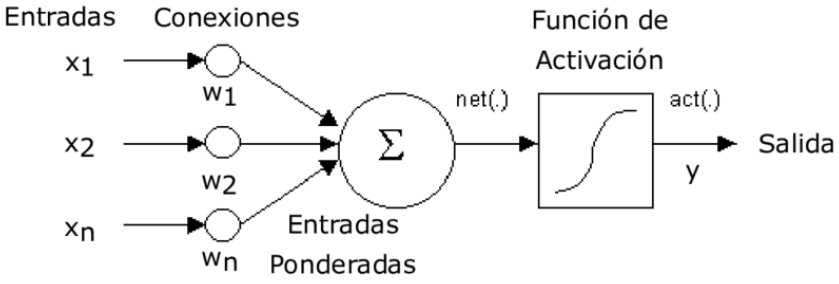


Figura 3. Estructura de una Neurona Artificial

3.1.2 Redes Neuronales Artificiales

Una red neuronal artificial puede entenderse como una función que cuenta con parámetros de entrada, con un cuerpo de la función donde se define la operación y con una salida que regresa el resultado [8]. Los elementos de una red neuronal son capas, en este caso los parámetros de entrada son neuronas y se denominan capa de entrada, cada una de las cuales se conecta con una neurona en la siguiente capa, denominada capa oculta que constituye el cuerpo de la función, y a su vez se conectan con una capa de salida, o retorno de la función, en la cual la neurona de salida entrega un número decimal entre 0 y 1.

Una red neuronal artificial puede aprender conocimiento jerarquizado, lo que significa que entre más capas se añaden a la red, más complejo es el conocimiento obtenido. La profundidad en la cantidad de capas es lo que da nombre al aprendizaje profundo. Una red neuronal se entrena con ejemplos y su resultado dependerá entonces de la calidad y de la cantidad de ejemplos con lo que se entrene.

En el aprendizaje profundo el módulo principal es la capa, dado que dentro de la misma capa se realizan las mismas operaciones con todas las neuronas: se aplican la misma fórmula de retropropagación y se aplica la misma función de activación.

El primer algoritmo clave dentro de la arquitectura de una red neuronal para realizar el aprendizaje automático se denomina retropropagación (*Backpropagation*) . Este algoritmo de aprendizaje supervisado permite que la información del error se propague hacia atrás en la red neuronal con el objetivo de minimizar la función de coste, ajustando los parámetros de pesos (*w*) y sesgos (*bias*) de la red a través de otro algoritmo denominado *Descenso del Gradiente* [9], que se explicará más adelante.

La función de coste es una medida del desempeño de un modelo de machine learning que cuantifica el error entre las predicciones hechas y los datos reales del modelo. El valor obtenido por esta función se denomina coste o pérdida; la finalidad del método es encontrar los parámetros que minimicen la función de coste [9]. Existen diversas funciones de coste, cada una de las cuales se utiliza dependiendo del problema de aprendizaje: Error Absoluto Medio, Error Cuadrático Medio, Entropía Cruzada Binaria y Entropía Cruzada Categórica. En problemas de clasificación con redes neuronales convolucionales, generalmente se emplea la entropía cruzada binaria como valor de pérdida.

Backpropagation se calcula a partir de las derivadas parciales de cada uno de los parámetros de la red (sus neuronas), con respecto a la función de coste. El objetivo es determinar el porcentaje del error en cada una de las neuronas de la capa, iniciando desde la última capa hasta la primera, para calcular cuánto debe modificarse cada parámetro y así optimizarlo.

El segundo algoritmo clave es el Descenso del Gradiente (*Gradient Descent*), utilizado ampliamente en modelos de regresión lineal para optimizar la función de coste a través del gradiente. En redes neuronales este algoritmo requiere del vector gradiente que es un vector con las derivadas parciales de los parámetros con respecto a la función de coste, las cuales fueron obtenidas por el algoritmo de retropropagación.

El descenso del gradiente estima numéricamente dónde una función genera sus valores más bajos, valiéndose de un optimizador de la función de coste, denominado ratio de aprendizaje, a partir del cual se evalúa cuánto afecta al gradiente la actualización de los parámetros en cada iteración; es decir, cuánto se avanza en cada paso hacia el valor más bajo. La correcta configuración del ratio de aprendizaje es fundamental para que el algoritmo de descenso del gradiente funcione bien.

3.1.3 Redes Neuronales convolucionales

Una red neuronal convolucional es un tipo de red neuronal diseñada para aprovechar la estructura espacial de una imagen [10]. Se caracteriza por aplicar un tipo de capa donde se realiza una operación matemática denominada convolución. Utilizando un filtro o kernel (simétrico de 3 X 3, 5 X 5, o 7 X 7) sobre la imagen original, calcula valores diferentes en los píxeles de imágenes analizados. A cada una de las imágenes generadas a partir de la aplicación de filtros se le denomina mapa de características, las cuales conforman un conjunto de mapas de características.

Los mapas de características son patrones establecidos por la relación de un píxel de imagen con sus píxeles vecinos, los cuales se analizan para ayudar a identificar una imagen. Las convoluciones detectan por ejemplo, cambios de texturas, superficies planas, o cambios de contraste. La potencia de esta red neuronal es la secuencia de la operación donde los parámetros de salida de una capa se convierten en los parámetros de entrada de la siguiente y, en ese sentido, se realizan detecciones sobre las detecciones obtenidas en capas anteriores.

El filtro pasa sobre la imagen original, los valores se multiplican y se suman por el píxel vecino para obtener un nuevo valor en el centro de la matriz.

Una arquitectura de una red neuronal convolucional se representa como un embudo donde la imagen original se va comprimiendo espacialmente, es decir, su resolución se va perdiendo, al mismo tiempo que su grosor aumenta; al final del embudo, el resultado puede entregarse a otra red neuronal totalmente conectada (fully connected) que tomará la decisión sobre lo que se detecta en la imagen.

3.1.3.1 Capas de reducción.

Las capas de reducción de la dimensionalidad o pooling layers, tienen como objetivo reducir el tamaño espacial de las características convolucionadas y reducir con ello el tiempo de cómputo requerido para el procesamiento de datos. Así mismo, extraen las características de alto nivel que son invariantes a rotaciones y traslaciones [11].

Existen dos tipos de capas de reducción: reducción máxima (max-pooling) y reducción promedio (average-pooling); mientras la capa de reducción máxima regresa el valor máximo de la parte de la imagen cubierta por el núcleo, la capa de reducción promedio regresa el promedio de todos los valores de la parte de la imagen cubierta por el núcleo [11]. La Figura 4 muestra el resultado de las capas de reducción descritas.

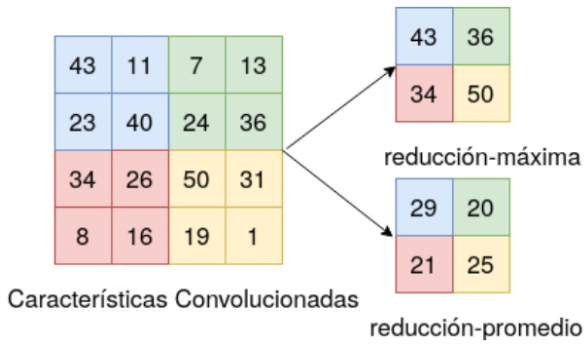


Figura 4. Capas de reducción con paso = 2. Tomado de [9].

3.1.3.2 Capas totalmente conectadas.

Las capas totalmente conectadas o fully connected, se utilizan generalmente en las arquitecturas de redes neuronales convolucionales en la etapa de clasificación, a la salida de las capas convolucionales y capas de reducción. La función de activación Softmax generalmente se utiliza en las capas totalmente conectadas, dado que permite la clasificación de múltiples clases; por el contrario, la función de activación sigmoide se utiliza en las capas totalmente conectadas en tareas de clasificación binaria.

3.1.3.3 Capas residuales.

Las capas residuales o residual network (ResNet), son bloques que alimentan a una capa convolucional que está a dos o tres capas de distancia, con el objetivo de evitar el problema del gradiente que se desvanece en las arquitecturas de redes profundas [11]. Con la implementación de capas residuales los gradientes más grandes se propagan hacia las capas iniciales para que las mismas aprendan de manera tan rápida como las capas finales.

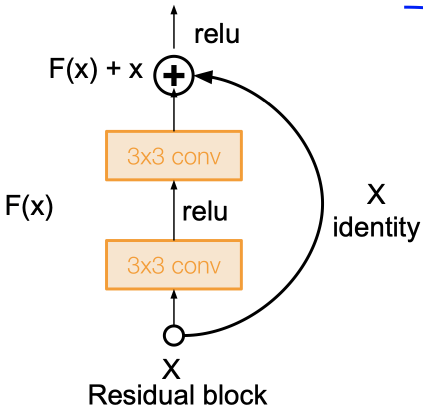


Figura 5. Arquitectura de un módulo residual. Tomada de [9].

3.1.3.4 Capacidad de generalización.

El principal desafío para una red neuronal convolucional es su capacidad para generalizar con nuevos datos de entrada, es decir, con datos diferentes a los conocidos durante el entrenamiento. Los términos de sobreajuste (*overfitting*) y sub ajuste (*underfitting*) hacen referencia a los problemas que tiene la red para generalizar o para mantener un correcto desempeño en presencia de nuevos datos. El sobre ajuste significa que el modelo solo es capaz de predecir bien en presencia de datos idénticos a los conocidos durante el entrenamiento, descartando aquellos que se salen de los rangos establecidos, mientras que el sub ajuste significa que el modelo no tuvo suficientes datos de entrenamiento y, por lo tanto, su capacidad de generalizar es baja y se muestra incapaz de hacer predicciones correctas.

3.1.4 Algoritmos para la detección de objetos basados en redes neuronales profundas

La detección de objetos requiere localizar uno o más objetos en una imagen o vídeo, lo cual es un enfoque diferente a la clasificación de imágenes. La clasificación de imágenes se refiere a predecir un solo objeto y su clase en una imagen, mientras que la detección combina dos enfoques: uno o más objetos en una imagen que se ubican y clasifican. La localización determina dónde se ubican los objetos en una imagen y luego forma un cuadro delimitador alrededor de ellos. La localización también puede verse como un problema de regresión.

Actualmente, la detección de objetos se usa principalmente en aplicaciones como detección de vehículos, detección de peatones, recuento de células sanguíneas, sistemas de videovigilancia en tiempo real, entre otros.

Las redes neuronales convolucionales utilizadas para la detección de objetos se clasifican generalmente en dos tipos: detector de dos etapas, denominado método basado en propuestas de región y comprende algoritmos como R-CNN, Fast R-CNN, Faster R-CNN y Mask R-CNN, y detector de una etapa, denominado clasificación/regresión y comprende algoritmos como SSD (Single Shot Detector) y YOLO (You Only Look Once).

A continuación se realiza una revisión de los algoritmos de una y de dos etapas utilizados en el campo de la detección de objetos con imágenes satelitales, basados en redes neuronales profundas, que han sido referenciados en investigaciones aplicadas recientes [12], [13], [14] y [15].

La detección de objetos o detección de clases de objetos se define como la determinación, a partir de una imagen, de si existen instancias de objetos de características predefinidas, y si las hay, devolver la ubicación espacial y la extensión de cada instancia [16]. La localización de objetos altamente estructurados, como automóviles, paneles solares, viviendas, techos, calles, y de objetos articulados, como seres humanos en bicicleta o asentamientos informales, reúne el interés del presente proyecto aplicado.

3.1.4.1 Modelos de detección por regiones de interés (R-CNN ,Fast R-CNN, Faster R-CNN, RFCN, Mask R-CNN)

Este conjunto de algoritmos para la detección de objetos actúa en dos etapas: una visión del escenario general y luego, un foco a las regiones de interés. Estos modelos de red tiene bajo rendimiento en procesamiento computacional pero alta precisión en la detección de objetos pequeños; la familia de algoritmos R-CNN son redes neuronales convolucionales basadas en regiones.

R-CNN (2013): Esta red neuronal asigna un cuadro delimitador a cada objeto que se encuentra en una imagen. A partir de la imagen de entrada, el algoritmo extrae las regiones en la imagen que potencialmente contienen objetos, realiza un aprendizaje por transferencia a partir de convoluciones previamente entrenadas y, finalmente, clasifica las regiones empleando una máquina de soporte vectorial en un cuadro delimitador [17]. R-CNN fue el primero en su tipo, pero hoy no se utiza porque es costoso en el tiempo y en el espacio y las pruebas son lentas.

Fast R-CNN (2015): Añade un módulo de agrupación de regiones de interés con una capa especial para obtener predicciones de las etiquetas de clase y otra para obtener el cuadro delimitador de cada objeto. Genera propuestas de región procesando la imagen completa, agrupando las características de la CNN, y agregando un módulo de agrupación de región de interés, denominado ROI Pool, el cual funciona extrayendo una ventana de tamaño fijo del mapa de características para obtener las etiquetas de los objetos (clasificador de capas) y el cuadro delimitador (regresión de capas) [17].

Faster R-CNN (2015): Este algoritmo, sucesor de los algoritmos R-CNN y Fast R-CNN, primero pasa la imagen por un red neuronal convolucional para trazar un mapa de características; a continuación, en una red separada, ejecuta un mapa de activación para producir regiones de interés, usando en cada región varias capas completamente conectadas a las salidas, con las coordenadas de la caja de enlace [18]. Este método también utiliza un generador ROI pool externo que recibe el mapa de características de la imagen para que la capa de convolución determine dos salidas: una con la clasificación de los objetos y otra con el cuadro delimitador. El entrenamiento es complejo.

RFCN con ResNet101 (2016): En las redes neuronales de la familia R-CNN se genera primero, una red separada (RPN) que produce las regiones de interés, luego se activa un módulo ROI pool donde se pasa por capas completamente conectas para la clasificación y regresión del cuadro delimitador [19]. En el R-FCN, todavía se tiene la capa (RPN) que produce propuestas de regiones, pero se generan mapas de puntuaciones antes pasar por el módulo ROI pool. Todas las regiones hacen uso del mismo conjunto de mapas de puntuación sensibles para realizar la valoración de la media. El entrenamiento es complejo.

Mask R-CNN (2017): Se basa en algoritmo de detección de objetos R-CNN (2013), Fast R-CNN (2015) y Faster R-CNN (2015). Le otorga prioridad a la máscara de los objetos (predicción de máscara), es decir, a la clasificación de los objetos, usando una interpolación lineal en la capa ROI. Prescinde del cuadro delimitador. Predice una máscara de objeto a partir de una segmentación pixel a pixel, ubicando píxeles exactos de cada objeto en vez de cuadros delimitadores de cada objeto. Reemplaza ROI Pool por ROI Align. Si bien este algoritmo es eficiente para la segmentación de instancias, requiere un gran número de datos para el entrenamiento, por tal motivo es frecuente que se utilicen técnicas de transfer learning, para reutilizar el aprendizaje logrado en tareas similares o relacionadas.

3.1.4.2 Modelos de detección clasificación/regresión o de una sola etapa

YOLOv3 (2018): La arquitectura YOLO (You Only Look Once version 3) es una red neuronal convolucional para la detección de objetos en tiempo real, a partir de videos o imágenes. YOLO predice simultáneamente múltiples cuadros delimitadores y probabilidades de clase para estos cuadros.

La detección de objetos se modela como un problema de regresión a diferencia de los métodos basados en clasificadores. Inicialmente la imagen se divide en una cuadrícula simétrica y para cada una de las celdas de la cuadrícula se predicen: a) los cuadros delimitadores, b) la confianza para cada uno de esos cuadros y c) la probabilidad de ser de una clase. Si bien se ha identificado que YOLO tiene dificultades para detectar objetos pequeños en agrupaciones, el uso acotado de solo 98 cuadros delimitadores en toda la imagen, permite la predicción en tiempo real y es capaz de detectar muchos objetos en una imagen. Todo el código de entrenamiento y de prueba es código abierto y permite la descarga de una variedad de modelos pre entrenados.

SSD Single Shot Detector (2016): Este algoritmo de detección de objetos, dado un mapa de características específicas, aprovecha un conjunto de cajas de anclaje con diferentes radios y escalas para discretizar el espacio de salida de las cajas delimitadoras [20]. Maneja objetos de varios tamaños fusionando las predicciones de múltiples mapas de características con diferentes resoluciones, y añade varias características hasta el final de la red, las cuales son responsables de las predicciones de las clases de objetos en distintas cajas delimitadoras y de sus valores de confianza asociados.

SSD ubica cajas delimitadoras con diferentes tamaños y relaciones de aspecto; a cada cuadro delimitador le calcula una puntuación de clase y 4 desplazamientos por defecto, relativos a la forma original del cuadro delimitador. La red es entrenada con una suma ponderada de pérdida de localización que incluye los parámetros del cuadro de predicción y los parámetros del cuadro correcto y, adicionalmente, realiza compensaciones para el punto central, en el ancho y el largo del cuadro delimitador [20].

Redes como YOLO y SSD al estar estructuradas en una sola etapa prescinden de algoritmos de preprocesamiento, realizan predicciones con menos regiones candidatas y se apoyan en subredes de clasificación completamente convolucionales; si bien esto las hace más rápidas en términos de procesamiento computacional, son menos competitivas en referencia a otras arquitecturas para la detección de objetos pequeños.

3.1.5 Funcionamiento de la red YOLO

YOLO (You Only Look Once) es una red neuronal convolucional que utiliza características de la imagen completa para trazar cuadros delimitadores (bounding boxes), a través de los cuales la red reconoce todos los objetos presentes.

En el algoritmo YOLO, una imagen de entrada se divide en una celda de cuadrícula de *N* x *N*. El número de cuadrículas depende de la complejidad de la imagen. Si el punto central del objeto cae en una cuadrícula, esa cuadrícula tiene la responsabilidad de detectar ese objeto. Cada celda de la cuadrícula realiza la localización y clasificación del objeto. Cada cuadrícula predice el número *B* de los cuadros delimitadores y la puntuación de confianza correspondiente. La puntuación de confianza implica la existencia o la inexistencia del objeto y representa la probabilidad que tiene el objeto de estar en esa celda y de cuán precisa es la predicción. Si no hay ningún objeto en la celda de la cuadrícula, la puntuación de confianza se vuelve cero cuando Pr(Object)=0. Si existe algún objeto en la celda de la cuadrícula, Pr(Objeto)=1.

La confianza (*o*) es el resultado de multiplicar la probabilidad de que el objeto esté en la celda y la medida de intersección sobre la unión (*IOU*). Adicionalmente, cada cuadro delimitador expresa la probabilidad (Pr) de clase para cada una de las clases del Modelo.

Figura 6. Representación de la métrica Intersección sobre la Unión (IOU).

La intersección sobre la unión (*IOU*) expresa la relación entre el área de intersección de dos regiones: el cuadro delimitador del objeto etiquetado (*ground truth bounding box*) y el cuadro delimitador predicho (*predicted bounding box*), sobre el área de la unión de las mismas. Esta métrica se usa para evaluar la precisión de las detecciones realizadas por el modelo y determina la puntuación de confianza de la detección.

La puntuación de confianza de clase calcula la probabilidad de que el cuadro delimitador detectado contenga un objeto de la clase correspondiente. Cada celda en la cuadrícula de la imagen realiza predicciones de las probabilidades de clase *Pr*(clase); estas probabilidades están condicionadas a la presencia de un objeto en la celda. Al final se realiza una única predicción de objeto sin importar el número de cuadros delimitadores. La puntuación de confianza indica la probabilidad de que un objeto esté presente en un determinado cuadro y qué tan bien se ajusta el cuadro predicho al objeto [21].

Cada cuadro delimitador consta de varias predicciones: *x*, *y*, *w*, *h* y confianza (*o*), donde las coordenadas (x, y) representan el centro del cuadro delimitador en relación con los límites de la celda de la cuadrícula; las coordenadas w y h, hacen referencia al ancho y alto, calculado en relación con la imagen completa. Los valores *x*, *y*, *w*, *h* deben ser normalizados en el rango entre 0 y 1.

x: Posición x del centro del cuadro delimitador relativo a la celda de la cuadrícula a la que está asociada.

y: Posición y del centro del cuadro delimitador relativo a la celda de la cuadrícula a la que está asociada.

w: Ancho del cuadro delimitador.

h: Altura del cuadro delimitador.

o: Valor de confianza de que existe un objeto dentro del cuadro delimitador.

Pr: Probabilidades de clase para cada una de las clases del modelo.

Otro parámetro de la red YOLO lo constituye el tamaño de las imágenes que debe ser cuadrado, con el único requisito que sea múltiplo de 32, recibiendo imágenes de 416 X 416, 512 X 512, 640 X 640 y 1280 X 1280 píxeles.

Dado que la arquitectura YOLO es una red pre-entrenada en un conjunto de datos amplio, permite abordar el proceso particular de dominio mediante aprendizaje por transferencia (transfer-learning) para beneficiarse de los patrones y características aprendidos durante el entrenamiento en conjuntos de datos masivos.

3.2. Antecedentes de la Detección de Asentamientos Informales usando imágenes satelitáles VHR

3.2.1 Análisis de la segregación espacial de asentamientos informales en ciudades

latinoamericanas.

De acuerdo con la revisión realizada [22], el uso de imágenes satelitales de muy alta resolución (VHR) para mapear asentamientos informales en América Latina es escaso, principalmente por las limitaciones temporales de las imágenes VHR y por los costos de acceso a las mismas. Usando imágenes de resolución moderada del sensor LANDSAT, mejoradas con enfoque panorámico, [22] analizaron áreas residenciales en las periferias de Bogotá y Sao Paulo para evaluar la segregación espacial, de movilidad y de exposición a peligros ambientales en asentamientos informales frente a otros de clase media y alta.

El método empleado fue de clasificación estratificada, primero mediante clasificación supervisada, con algoritmo de máxima verosimilitud a través del cual se identificaron 8 tipos de cobertura de suelo del área urbana; segundo, mediante una nueva clasificación de las áreas urbanas para detectar asentamientos formales e informales.

Las características atribuidas por el estudio a los desarrollos urbanos formales fueron: lotes de tamaño regular, forma cuadrada, materiales de techo de forma similar, patrón de cuadrícula urbana, calles pavimentadas con aceras, postes de luz, señales de tránsito y suministro eléctrico; mientras que las características atribuidas a los asentamientos informales fueron: tamaño de calle pequeña, estructura urbana orgánica no pavimentada o inexistente, tamaño de parcela, formas y alturas y materiales de techo irregulares.

Una vez clasificadas las áreas con asentamientos formales e informales, aplicaron un enfoque de evaluación de criterios múltiples (MCE) basados en ráster, con el objetivo de identificar áreas de riesgo ambiental por proximidad de ríos y humedales, proximidad de bosques y tipos de pendiente, y medir así las condiciones ambientales peligrosas de los asentamientos. Adicionalmente, usando datos de los sistemas de transporte masivo, midieron la densidad de los equipamientos urbanos de movilidad como paraderos de bus, calles con rutas de autobús y carreteras regionales para medir la movilidad desde los asentamientos informales.

El estudio concluye que el patrón de asentamientos informales en Bogotá y Sao Pablo es similar, con construcciones cercanas a la periferia urbana y de las principales carreteras conectadas con el empleo, la educación y atención médica; concluye además que los asentamientos informales, en el área de estudio, migraron hacia lugares con mejores condiciones ambientales y de movilidad.

3.2.2 Mapeo de asentamientos informales usando redes neuronales convolucionales

El uso de Redes Neuronales Convolucionales (CNN) para la detección de asentamientos informales usando imágenes satelitales VHR se ha probado por [23], entrenando CNN de extremo a extremo para la detección de asentamientos informales en Dar es Salaam, Tanzania.

A partir de 3 imágenes satelitales VHR de 2000 X 2000 pixeles, con 4 bandas multiespectrales y resolución espacial de 0,6 metros, el estudio implementa un método de clasificación basado en parches para dividir las imágenes. Los datos de entrada fueron normalizados y la selección de conjuntos de entrenamiento se realizó a partir de un muestreo estratificado basado en frecuencias de clase para asentamiento formal y asentamiento informal.

Usando la arquitectura propuesta por [24] como base, la red implementada esta compuesta por un bloque convolucional de 2 capas, la primera con 32 unidades en el kernel, de 25 X 25, y la segunda con 64 unidades en el kernel de 17 X 17, emplea la función de activación Unidad Lineal Rectificada (ReLU), y capas de agregación máxima (max pooling) con paso 1. La salida de la capa convolucional final se aplanó en un vector unidimensional para alimentar el siguiente bloque compuesto por capas completamente conectadas (fully connected). La capa de salida fue normalizada con función de activación Softmax para la clasificación final.

Para optimizar la convergencia del modelo durante el entrenamiento se utilizó gradiente estocástico con impulso (SGD momentum), el cual utiliza muestras aleatorias del conjunto de entrenamiento para actualizar los pesos del modelo, agregando un término de impulso a las actualizaciones de los pesos, evitando las oscilaciones del gradiente. Para reducir el sobre ajuste del modelo, se afinaron 3 hiper parámetros: abandono (drop out=0.5), desactivando un porcentaje de neuronas y sus conexiones en las capas oculta; detención anticipada (early stopping), en el punto donde la función de pérdida evaluada en el conjunto de validación no mejora después de determinadas épocas; y regularización L2, que penaliza los pesos grandes del modelo.

Los autores compararon el rendimiento de la clasificación obtenida por la CNN con otros algoritmos como máquinas de soporte vectorial (SVM), usando solo bandas espectrales; SVM mejorado con características de Matriz de concurrencia de nivel de grises (GLCM), usando las 4 bandas originales, en tres conjuntos de entrenamiento con 1080, 2160 y 3060 imágenes. Concluyen que las CNN entrenadas de extremo a extremo pueden aprender de manera efectiva características complejas, jerárquicas y abstractas para la clasificación de asentamientos informales.

3.2.3 Mapeo de asentamientos informales urbanos a escala fina con red de fusión multimodal basada en transformers

A diferencia de los estudios que mapean asentamientos informales basados en parcelas (clasificación de imágenes) o en píxeles de imágenes satelitales (segmentación semántica), enfoque actuales como [25] utilizan información de objetos para mejorar la segmentación semántica de los asentamientos informales. Los autores implementan un modelo experimental para la detección de asentamientos informales urbanos en la ciudad de ShenZhen, China, usando imágenes satelitales VHR y datos de series temporales de los patrones de actividad humana (TDP),

a partir de un ensamble de algoritmos que aprenden por separado las funciones temporales y espaciales, y después fusionan ambas características a través de un transformador que optimiza la clasificación de asentamientos informales. El propósito de esta investigación es demostrar que los datos multimodales mejoran de manera integral el rendimiento de la clasificación.

Para aprender características espaciales el modelo (ResMixer) utiliza redes neuronales convolucionales para extraer características en las capas superficiales y utiliza después una red neuronal Perceptrón Multicapa (MLP-Mixer) para aprender características más diversas en capas más profundas, junto con una estructura para optimizar la clasificación.

El modelo para aprendizaje de características temporales toma una serie temporal de datos de actividad humana como entrada, las cuales son transformadas con el uso de redes neuronales para obtener un mapa de características mejorado, con el formato requerido para ser fusionado. Posteriormente, las características espaciales y temporales pasan a través de una capa transformadora que las fusiona, donde se agrega el factor de clasificación para discriminar si es o no es un asentamiento informal.

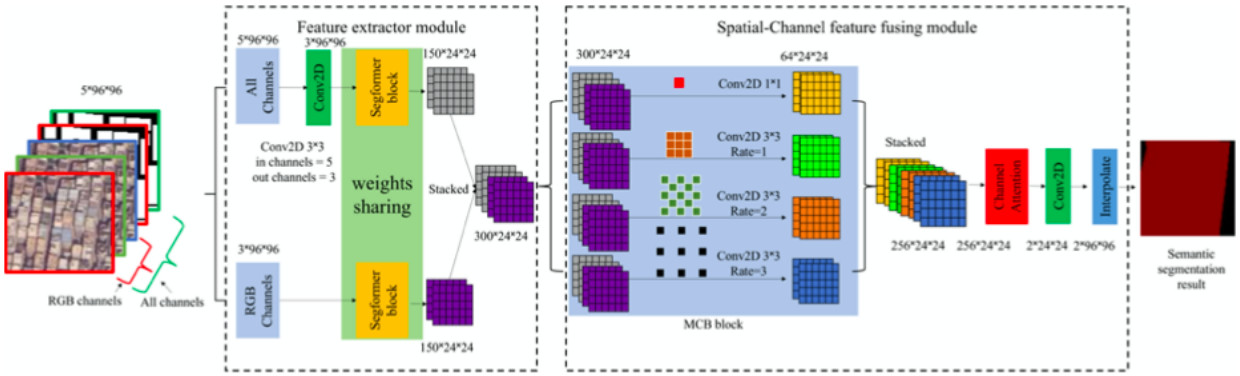


Figura 7. Arquitectura de la red UisNet. Tomado de [22].

El abordaje metodológico de este estudio es complejo en tanto que el modelo propone una red híbrida de fusión de características espacio temporales basada en Transformadores (STNet), que crea una capa de fusión espacio temporal. Lo interesante no es tanto la complejidad del modelo sino la apuesta por mejorar la precisión de la detección de asentamientos informales usando una combinación de distintas modalidades de datos.

3.2.4 Implementaciones de YOLO en detección de objetos usando imágenes VHR

Algunas implementaciones de la red YOLO [21] han comparado el comportamiento del rendimiento de diferentes algoritmos de la familia YOLO, aplicados a la detección de nidos de pájaros y otros objetos en líneas de transmisión eléctrica, mediante imágenes aéreas obtenidas por drones.

Interesa para el proyecto aplicado Detección de Asentamientos Informales usando imágenes satelitales VHR de Santiago de Cali, que [21] y [26] describen en detalle los componentes de la arquitectura YOLO, sus principales hiper parámetros y el entorno de programación necesario para su implementación en tareas de detección de objetos, cuando el conjunto de datos de entrenamiento es pequeño. La apuesta del proyecto aplicado es emplear el enfoque de ajuste fino (fine tuning) para aprovechar los pesos entrenados de YOLO en la detección de objetos, y desarrollar un modelo con capacidad de clasificación de asentamientos informales.

4. BASE DE DATOS CON IMÁGENES VHR DE SANTIAGO DE CALI

4.1 Imágenes Satelitales como fuente de datos

Las imágenes satelitales tienen resolución espacial medida por el tamaño del píxel, resolución espectral medida por la capacidad de discriminar entre longitudes de onda vecinas y número de bandas disponibles, y resolución temporal medida en el tiempo que tarda el sensor en tomar dos imágenes del mismo sitio.

La resolución espacial está relacionada con la calidad de la imagen, o tamaño de un píxel en el suelo. Se considera alta resolución espacial imágenes entre 30 cm y 5 metros por píxel; media resolución espacial imágenes entre 10 y 30 metros por píxel; y baja resolución espacial, imágenes mayores a 60 metros por píxel.

Las imágenes de muy alta resolución o VHR (very high resolution) tienen resolución espacial inferior a 1 metro por píxel. Estas imágenes permiten la identificación de objetos pequeños como personas, realizar análisis de texturas de la superficie como vegetación, suelo o estructuras, y especialmente, la clasificación de objetos en diferentes categorías.

Las bandas espectrales contienen información que permite comprender las propiedades de la superficie terrestre. Están compuestas por longitudes de onda con información específica sobre características particulares como agua poco profunda (B1), agua profunda y atmósfera (B2), vegetación (B3), estructuras, suelo y vegetación (B4), costas y vegetación (B5), temperatura y humedad del suelo (B6) y (B7), alta resolución de imagen a color o pancromática (B8), detección de nubes (B9).

Las imágenes VHR que tienen resolución espectral de 8 bandas o más, se miden en nanómetros (nm) y representan diferentes longitudes de onda de luz visible e infrarroja. Permiten el monitoreo de vegetación, la clasificación de usos del suelo (agrícola, bosques, urbano), y detección de desastres naturales como incendios, inundaciones o deslizamientos de tierra.

4.2 Bases de datos de imágenes etiquetadas

Las bases de datos de imágenes para entrenamiento de modelos de aprendizaje profundo son, por lo general, conjuntos de datos de gran escala con imágenes etiquetadas con información de su contenido. Las principales referencias de bases de datos de imágenes existentes son ImageNet, creada en 2010 con el propósito de fomentar el desarrollo de algoritmos para clasificación de

objetos en imágenes, que contiene 14.000.000 de imágenes etiquetadas manualmente con 1.000 categorías de objetos diferentes; y MS COCO (Microsoft Common Objects in Context), creada en 2014 con el propósito de entrenar modelos de visión por computadora para tareas de detección y segmentación de objetos en imágenes en diferentes categorías, que contiene 330.000 imágenes etiquetadas con 2.5 millones de instancias de objetos pertenecientes 91 categorías diferentes como personas, animales, vehículos, objetos domésticos, lugares, entre otras. La resolución espacial de las imágenes en las bases de datos ImageNet y MS COCO es de 224 X 224 píxeles; no tienen ninguna resolución espectral.

Bases de datos de imágenes como MS COCO, han servido para entrenar modelos de aprendizaje profundo que realizan tareas de clasificación y detección de objetos como la red YOLO. Esta red convolucional fue entrenada para detectar una amplia variedad de objetos en diversos entornos por lo que es empleada para aprovechar sus pesos entrenados en tareas de detección cuando se cuenta con pocos datos de entrenamiento.

4.3 Base de datos con imágenes VHR de Santiago de Cali.

Para el desarrollo de una herramienta de detección de objetos territoriales en imágenes satelitales de Santiago de Cali, se propuso elaborar una base de datos con imágenes de muy alta resolución espacial VHR, etiquetadas de acuerdo con la presencia o ausencia de asentamientos informales.

Para elaborar la base de datos de entrenamiento se realizaron las siguientes tareas: 1) obtener imagen VHR de la ciudad de Cali; 2) recortar las imágenes en secciones homogéneas; 3) corrección y ajuste de las imágenes; 4) extracción de características espaciales de forma, tamaño y ubicación de los objetos, a través del etiquetado manual; 5) generación del conjunto de datos etiquetados. A continuación se describe cada una de las tareas realizadas para elaborar la base de datos de entrenamiento.

4.3.1 Análisis exploratorio de las fuentes de información.

Se propuso contar con imágenes de muy alta resolución espacial para cumplir los requerimientos técnicos del entrenamiento supervisado. Se accedió a 3 conjuntos de imágenes de Santiago de Cali, en bandas de espectro visible RGB (red, green, blue), que permiten identificar y etiquetar suficientes objetos urbanos, incluida la clase de interés: asentamiento informal. La Tabla 1 describe las características técnicas de las imágenes identificadas y accedidas para el desarrollo del proyecto Detección de fenómenos territoriales usando imágenes VHR de Santiago de Cali; las figuras 7, 8 y 9 corresponden a las imágenes VHR de referencia.

4.3.1.1 Fuente Mosaico Satelital_2016_2do_semestre_mc

La Figura 7 corresponde a una imagen satelital de toda el área urbana y rural de Santiago de Cali, de muy alta resolución espacial, propiedad de la Alcaldía de Santiago de Cali. Esta imagen del año 2016 sirvió como línea base para la identificación de los Asentamientos Humanos de Desarrollo Incompleto (AHDI) existentes en la ciudad, en el marco de la formulación de la Política Pública de Mejoramiento Integral del Hábitat, adoptada mediante Acuerdo Municipal No.0411 de 2017.

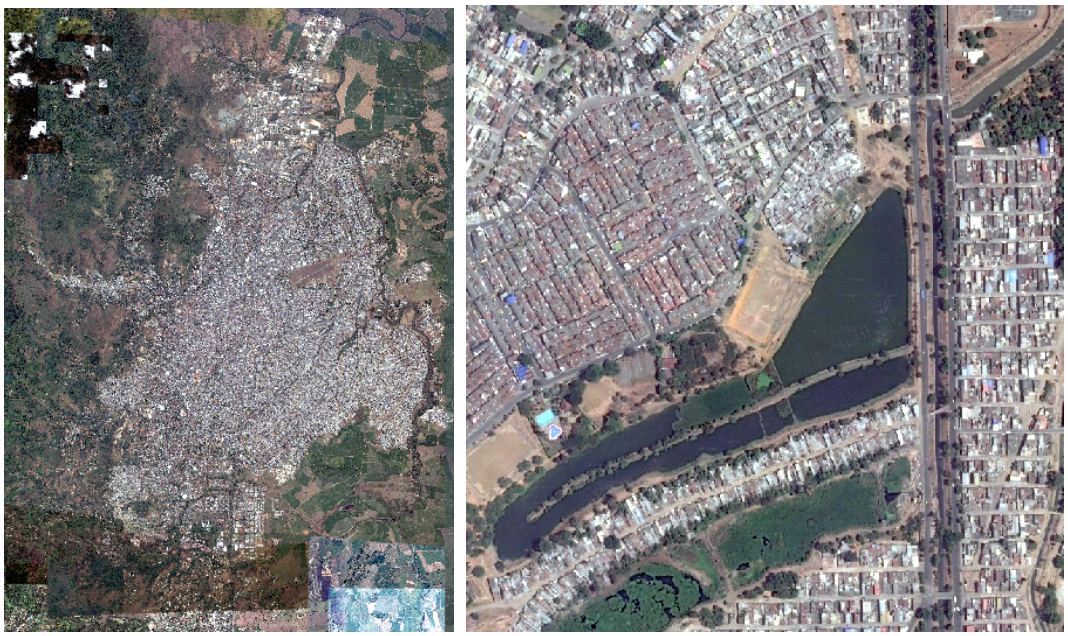


Figura 8. Imagen Satelital Santiago de Cali, 2016.

4.3.1.2 Fuente Ortofotomosaico Cali

La Figura 8 corresponde a una aerofotografía del borde perimetral de Santiago de Cali, de muy alta resolución espacial, propiedad de la Alcaldía de Santiago de Cali. Esta imagen del año 2020, recorre el contorno urbano y la zona rural colindante donde se presenta la mayor dinámica de crecimiento de los asentamientos informales en la ciudad de Cali.

Figura 9. Aerofotografía de bordes, 2020.

4.3.1.3 Fuente Zona_cortada_2_Oriente.

La Figura 9 corresponde a la sección oriental, urbana y rural de Santiago de Cali (R1C1), de foto mosaico compuesto por 9 imágenes satelitales de alta resolución espacial, propiedad de la Alcaldía de Santiago de Cali. La descarga se realizó en 2022, desde la plataforma Secure Watch de Maxar Technologies.

Figura 10. Imagen satelital sensor Maxar, 2022.

A partir de la valoración de propiedades de las imágenes como la resolución espacial y de atributos como representación de la clase de interés en diferentes topografías, se escogió la imagen Ortofotomosaico Cali (2020) como fuente de información para la construcción de la base de datos de entrenamiento.

Características de las imágenes identificadas para proyecto Detección de Fenómenos Territoriales en Santiago a partir de imágenes VHR.										
No.	Nombre de la imagen	Tamaño píxel	Resolución espacial	Número de bandas	Tipo	Formato	Sistema de Referencia de Coordenadas	Peso del archivo	Año	Descripción de la imagen
1.	**Mosaico_satelital_2016_2do_semestre_mc**	0.29 metros	Muy Alta	3	RGB	GeoTIFF	MAGNA-SIRGAS / Cali urban grid	23.25 GB	2016	Cobertura completa zona rural y urbana de Santiago de Cali.
2.	**ORTOFOTOMOSAICO CALI**	0.1 metros	Muy Alta	4	RGB	GeoTIFF	MAGNA-SIRGAS / Cali urban grid	105.64 GB	2020	Ortofotomosaico de bordes de Santiago de Cali con temporalidad espacial de 4 meses entre cada toma
2.1	Cali_Primera_Toma							26.66 GB		
2.2	Cali_Segunda_Toma							25.90 GB		
2.3	Cali_Tercera_Toma							26.96 GB		
2.4	Cali_Cuarta_Toma							26.12 GB		
3.	**Zona_cortada_2_oriente**	3.23 metros	Alta	3	RGB	GeoTIFF	EPSG:4326 - WGS 84	780 MB	2022	Imagen satelital de sección oriental de Santiago de Cali que incluye zona urbana y zona rural.
3.1	zona_cortada_2_oriente_BROWSE							588 KB		
3.2	zona_cortada_2_oriente_R1C1							164.69 MB		
3.3	zona_cortada_2_oriente_R1C2							145.34 MB		
3.4	zona_cortada_2_oriente_R2C1							183.53 MB		
3.5	zona_cortada_2_oriente_R2C2							154.29 MB		
3.6	zona_cortada_2_oriente_R3C1							53.55 MB		
3.7	zona_cortada_2_oriente_R3C2							46.98 MB		
3.8	zona_cortada_2_oriente_R3C3							31.04 MB		

Tabla 1. Imágenes de Santiago de Cali accedidas.

4.3.2 Preprocesamiento de las imágenes

Utilizando la herramienta QGis (versión 3.32 Lima)[1], se realizó el análisis exploratorio y tratamiento de la imagen fuente. Primero, se creó una capa de información (shapefile) con cuadrícula homogénea sobre la imagen completa, con 1344 celdas, cada una equivalente a 640 metros cuadrados. La Figura 10 ilustra la capa con la grilla creada sobre la imagen Ortofotomosaico Cali y un acercamiento a las celdas correspondientes al territorio de la Comuna 1.

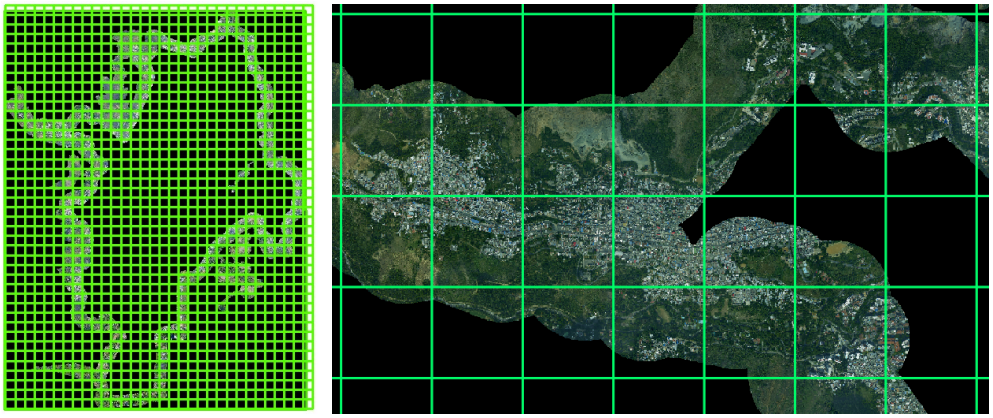


Figura 11. Grilla inicial sobre imagen Ortofotomosaico Cali. Elaboración propia.

[1] QGis es un sistema de información geográfico libre y de código abierto que permite aplicar técnicas de preprocesamiento de imágenes compuestas por píxeles y la superposición de capas de información (shape files), operando desde un computador de escritorio.

Se creó una segunda capa de información con grilla depurada sobre aquellas celdas con imágenes relevantes, asignando un identificador único a las celdas seleccionadas y descartando el resto, tal como se observa en la Figura 11. De esta manera se identificaron 239 celdas con la información relevante para realizar el recorte de la imagen fuente.

Figura 12. Grilla con ID de celda sobre imagen Ortofotomosaico Cali. Elaboración propia.

Antes de realizar el recorte de cada celda de imagen se ajustó el sistema de coordenadas geográficas de la imagen original para asegurar que las celdas de la grilla tengan las mismas dimensiones en coordenadas geográficas que en píxeles. En cada una de las celdas de la grilla, la herramienta QGis calcula las coordenadas geográficas de la esquina inferior izquierda, las cuales se utilizan para extraer el valor de la imagen original correspondiente a la ceda.

Se obtuvieron 239 imágenes de dimensiones X:6406, Y:6372, en Formato GeoTiff. La Figura 12 ilustra la visualización de la imagen reconstruida a partir de los recortes realizados.

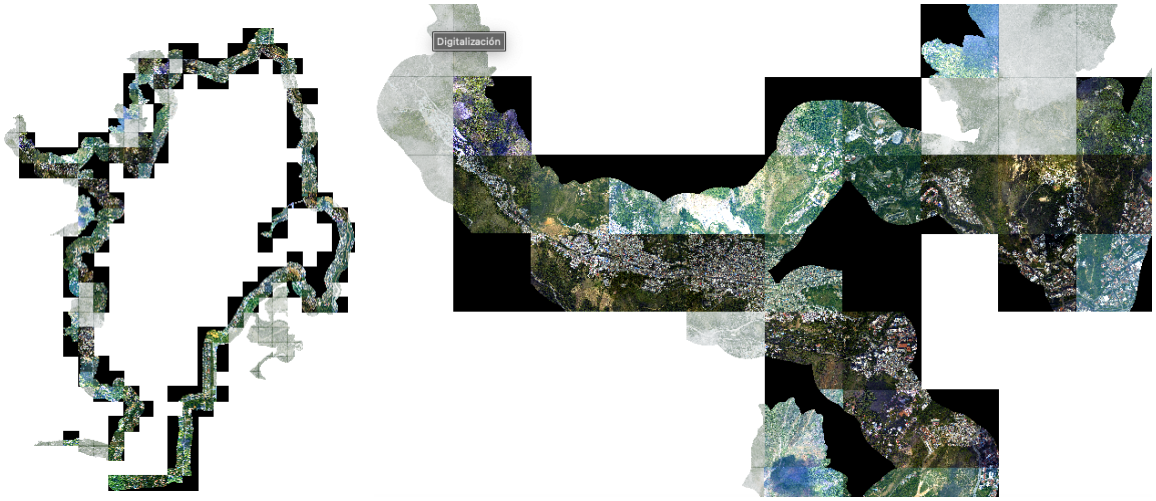
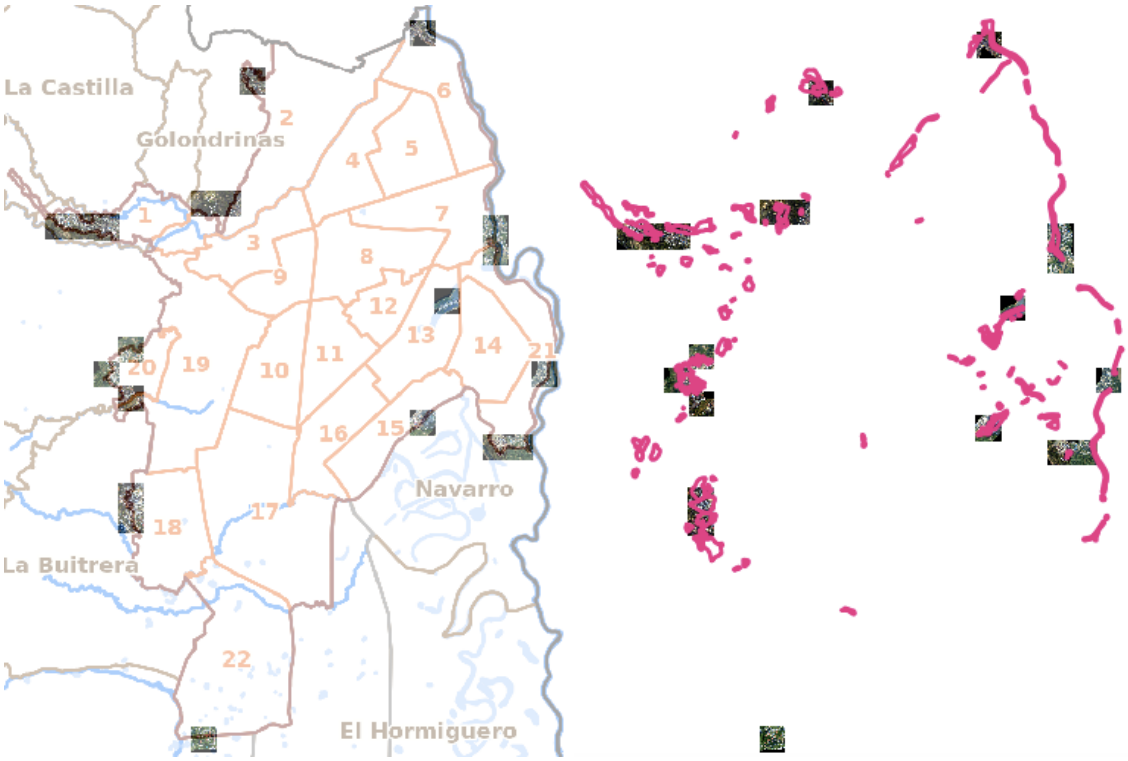


Figura 13. Imagen reconstruida a partir de recortes. Elaboración propia.

De este ejercicio se obtuvo el siguiente resultado: 121 recortes con imágenes en condiciones óptimas, 70 recortes de imágenes con pérdida de resolución espacial, 34 imágenes con saturación de luz que requiere ajuste de color, 8 recortes con imágenes incompletas, 3 imágenes faltantes y 2 recortes con imágenes no relevantes.

Tomando como base los Asentamientos Humanos de Desarrollo Incompleto (AHDI) de Cali[2] de la Figura 13 y el análisis de expertos [27], se identificó que los asentamientos informales en la ciudad se localizan principalmente en la periferia urbana de las comunas 1, 18 y 20 sobre los cerros occidentales de Cali, las comunas 6, 7 y 21 colindantes con el Jarillón del río Cauca en sector nororiental y las comunas 13, 14 y 15 en el oriente de la ciudad en el Distrito de Aguablanca. También se identificaron AHDI en los límites de la zona urbana de la comuna 2 y en corregimientos de la zona rural como Golondrinas y Navarro.



[2] La capa de información con polígonos AHDI corresponde a la línea base trazada por la Política Pública de Mejoramiento Integral del Hábitat De Santiago de Cali, adoptada mediante Acuerdo municipal No.0411 de 2017. El Artículo 4 de Política Pública de Mejoramiento Integral de Hábitat (2017), define los Asentamientos Informales o Asentamientos Humanos de Desarrollo Incompleto (AHDI), como "Asentamientos humanos precarios que concentran hogares del área urbana o rural con tenencia irregular del suelo, con precariedad de sus viviendas y sin acceso o con acceso restringido a la infraestructura de movilidad urbana, servicios públicos domiciliarios y equipamientos básicos y complementarios."

Figura 14. Superposición 20 imágenes VHR de Cali con capa comuna y capa AHDI.

Tomando como base esta fuente de información, se seleccionaron 20 imágenes VHR con representación amplia de la clase de interés, en la zona urbana y rural de Cali, con diferentes características morfológicas de asentamientos y topográficas del terreno. La Figuras 14 y 15 muestran la correspondencia entre las 20 imágenes VHR seleccionadas con los polígonos AHDI y las comunas y corregimientos de la ciudad.

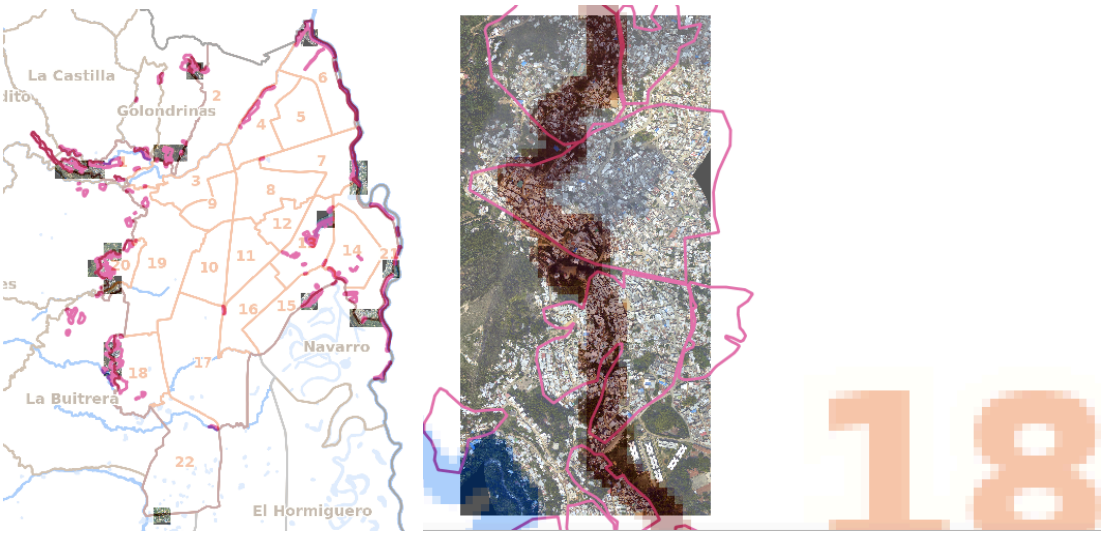


Figura 15. Superposición de 20 imágenes con capas AHDI, comuna y corregimientos

4.3.2.1 Ajuste final de imágenes VHR al formato y tamaño requeridos

Sobre cada una de las 20 imágenes VHR se creó una capa de información (*shapefile*) con cuadrícula de 3 X 3 para obtener 9 segmentos. Cada segmento de imagen se recortó individualmente conservando la información geográfica asociada y ajustando el tamaño de píxel de 0,1 metros a 0,25 metros, con la finalidad de obtener imágenes homogéneas de dimensión 1280 X 1280 píxeles, acordes con el parámetro máximo de entrada de la red YOLO.

Finalmente, cada uno de los recortes de imagen se exportó a formato PNG y se obtuvo un conjunto de 180 imágenes con los requerimientos técnicos de tamaño, peso y formato para iniciar el proceso de etiquetado manual en herramienta de software especializado.

4.3.3 Etiquetado manual del set de imágenes

El etiquetado tiene el objetivo de asignar categorías descriptivas que permiten clasificar el contenido de las imágenes. Las etiquetas proporcionan la información semántica con el contenido de las imágenes para su organización en un archivo, para la indexación en bibliotecas de imágenes, y fundamentalmente, para el entrenamiento de modelos de aprendizaje automático en tareas de clasificación y detección de objetos.

Las etiquetas representan los objetos territoriales presentes en las imágenes. Además de la clase de interés denominada asentamiento informal, se establecieron las siguientes categorías de objetos territoriales para el etiquetado: asentamiento formal, asentamiento rural, infraestructura (puentes, postes, equipamientos urbanos de gran calado), entorno natural (bosques y ríos), zona de cultivo, espacio público (zonas verdes, parques, plazoletas, canchas de futbol).

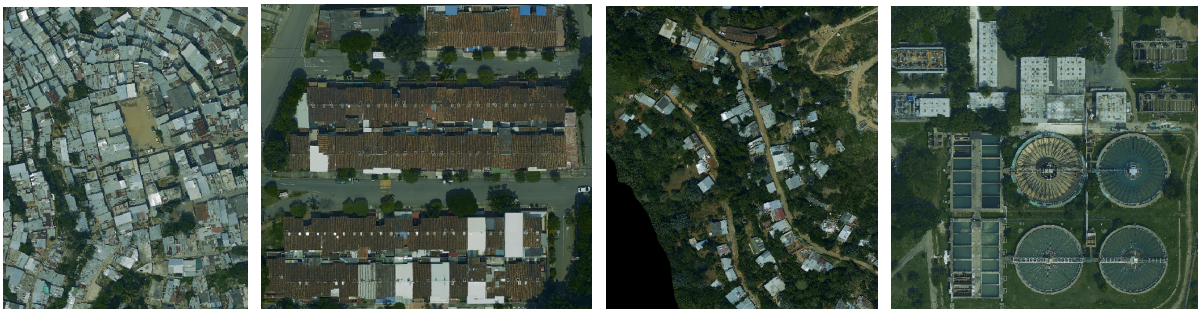


Figura 16. Objetos territoriales presentes en imágenes.

El proceso de etiquetado se realizó de manera manual, revisando cada una de las 180 imágenes a la luz de la información disponible sobre AHDI para la clase de interés, utilizando la herramienta SageMaker Ground Truth de Amazon Web Services (AWS).

Primero se configuró el trabajo de etiquetado. Este proceso requiere que se establezca una conexión directa con el repositorio del conjunto de imágenes y se defina una ruta de almacenamiento para los archivos con información de los objetos etiquetados en formato JSON. También permite configurar la tarea de etiquetado en sesiones automatizadas, de grupo o privada. El proceso finaliza con la selección del tipo de etiquetado, en este caso a través de cajas delimitadoras (*bounding boxes*).

En sesión de trabajo de etiquetado se definieron las etiquetas para 8 objetos territoriales según la siguiente nomenclatura: asentamiento informal (asen_infor), asentamiento formal (asen_for), asentamiento rural (asen_rural), entorno natural (natural), espacio público (es_público), infraestructura (infraes), zona de cultivo (cultivo), otro urbano (otro_urb) y otro_rural (otro_rural). Durante el proceso de etiquetado la herramienta asigna un color a cada una de las etiquetas, tal como se muestra en la Figura 16 donde se aprecia el ejercicio de etiquetado manual en una muestra del conjunto de imágenes.

Figura 17. Imágenes con etiquetas de las diferentes clases.

4.3.4 Atributos de la base de datos de imágenes etiquetada

El formato de salida de un conjunto de imágenes etiquetadas en la herramienta SageMaker Ground Truth es JSON Lines. Cada línea del archivo de salida contiene un objeto JSON con la siguiente estructura:

{“ruta a la fuente de almacenamiento de la imagen”, “nombre de carpeta de almacenamiento”: {tamaño de la imagen: [{“alto”: 1280, “ancho”: 1280, “profundidad”: 3 canales}], “anotaciones”:

[{“id_clase”: , “cordenadas x, y, h, w”}, {“id_clase”: , “cordenadas x, y, h, w”}, {“id_clase”: , “cordenadas x, y, h, w”}, {“id_clase”: , “cordenadas x, y, h, w”}…]}, “ruta del metado de la imagen”: {“objetos”: [{“confianza”: 0}, [{“confianza”: 0}, [{“confianza”: 0}…], “mapa de clases”: {“7”: “Espacio_público”, “5”: “Otro_rural”, “3”: “Entono_natural”…}, “tipo”: “nombre de la herramienta”, “anotación manual humana”, “fecha de creación”, “nombre del trabajo”}}

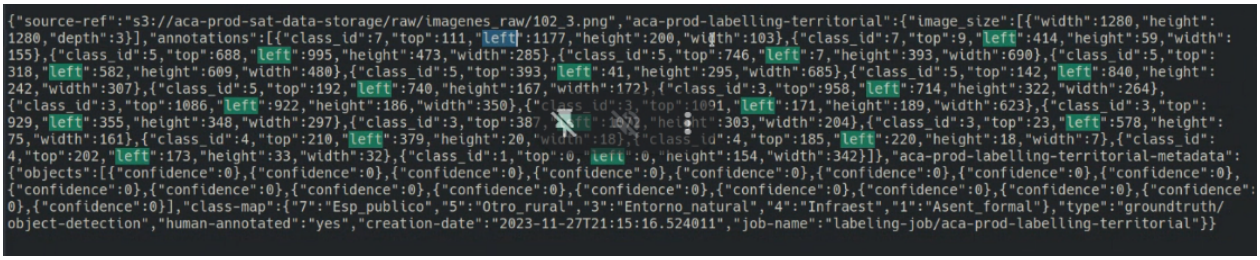


Figura 18. Imagen etiquetada en formato JSON.

Los archivos JSON con información de la imágenes etiquetadas reposan en repositorio de Amazon s3, para ser accedidos desde la herramienta SageMaker en etapa de modelado.

El tamaño de la base de datos con 166 imágenes etiquetadas es de 615 MB, con un tamaño promedio por imagen de 3,7 MB, en formato PNG sin información geográfica asociada.

Elaborar una base de datos con imágenes etiquetadas requirió transformar una imagen de gran tamaño, escala y formato geoespacial en un conjunto de imágenes de menor tamaño, manteniendo la resolución espacial en el rango VHR. En el preprocesamiento las imágenes pierden su resolución original y son despojadas de atributos de información como son las coordenadas geográficas propias de las imágenes de satélites y sensores remotos.

El foco principal del tratamiento de las imágenes estuvo en la identificación de segmentos de imagen ampliamente representativos de la clase de interés, a la luz del conocimiento experto disponible sobre el fenómeno territorial que dio origen al proyecto de ciencia de datos. La naturaleza de los datos espaciales requiere del uso de una herramienta para el análisis de información geográfica, que permitiera obtener un conjunto de imágenes en formato plano con el tamaño y peso adecuados para iniciar el entrenamiento del modelo.

La base de datos con imágenes etiquetadas es un resultado valioso en tanto que constituye el primer conjunto de datos con identificación de fenómenos territoriales en la Alcaldía de Santiago de Cali.

5. ENTRENAMIENTO DE LA RED YOLOv5

5.1 Familia YOLO

YOLO v5 es un modelo de la familia de los modelos de visión por computadora YOLO (You Only Look Once), usado comúnmente para la detección de objetos. Desde el lanzamiento de la red YOLO en el año 2015, se han creado nuevas versiones y actualizaciones para mejorar la efectividad de la herramienta en la detección de objetos en imágenes y video. A partir de YOLOv4 los nuevos modelos fueron introducidos por diferentes autores que modificaron la arquitectura medular de la red y optimizaron sus hiperparámetros.

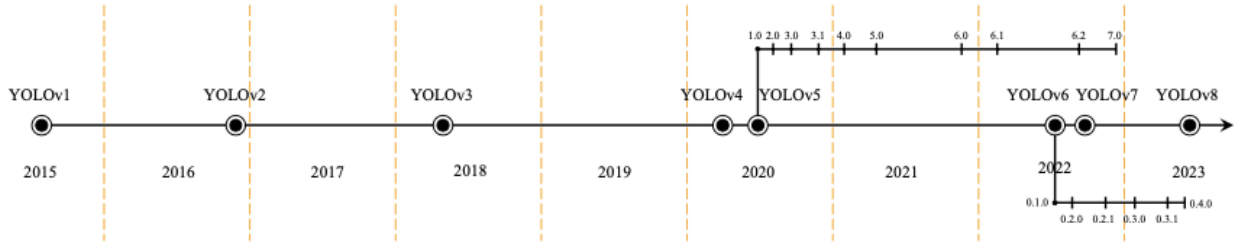


Figura 19. Línea de tiempo modelos YOLO. Tomado de [25].

En 2020, Ultralytics presentó YOLOv5 en cinco tamaños diferentes: nano, small, medium, large y extralarge. La red YOLOv5 se encuentra disponible desde el repositorio GitHub UltraLytics YOLOv5 [26], donde se encuentra la documentación completa y se puede descargar los pesos previamente entrenados para tareas de detección de objetos. Dado que utiliza una única red para la detección y la clasificación de objetos, ha sido optimizada para realizar las dos tareas de forma conjunta [28].

Para el entrenamiento del modelo *Detección de fenómenos territoriales en Santiago de Cali*, se implementaron las variantes YOLOv5l6 y YOLOv5x6, los cuales aceptan imágenes de entrada de tamaño 1280 píxeles.

5.2 Arquitectura de YOLOv5

La arquitectura de la red YOLOv5 es completamente convolucional (*fully convolucional network*). Está compuesta por columna vertebral (*Backbone*), donde se realiza la primera extracción de características de las imágenes de entrada; cuello (*Neck*), que conecta la columna vertebral con el cabezal, donde se combinan y fusionan características de diferentes escalas y niveles de

abstracción para mejorar la precisión, y cabezal (Head), donde se realizan las predicciones de la detección de objetos [29].

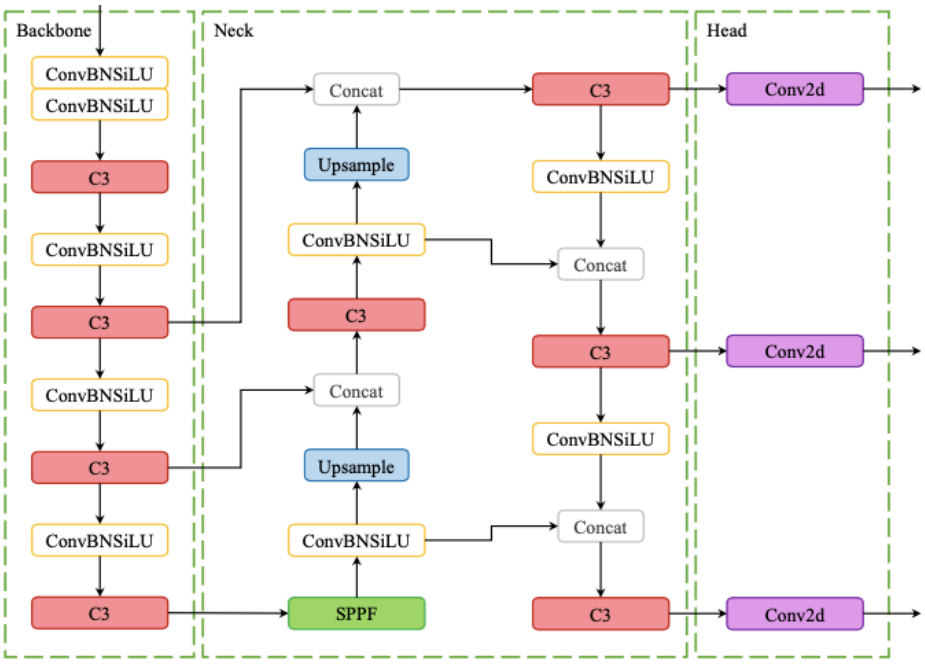


Figura 20. Arquitectura YOLOv5. Tomado de [25].

5.2.1 Backbone

La columna vertebral de YOLOv5, se denomina DarkNet53-CSP. Este componente intercala capas convolucionales con bloques C3, que son módulos fusionados con la estrategia CSP (*Cross Stage Partial*), donde se divide el mapa de características de la capa base en dos partes y luego se fusiona a través de una jerarquía entre capas, permitiendo el manejo adecuado de los gradientes redundantes y mejorando la eficiencia en la transferencia de información entre bloques residuales y capas convolucionales [28]; el módulo *Bottleneck* o cuello de botella, incluido dentro del bloque C3, mitiga el problema de los gradientes redundantes que afectan a las redes completamente convolucionales e implican cálculos de inferencia intensivos en recurso computacional, por lo que se generan menos parámetros y menos operaciones de coma flotante por segundo (*FLOPS*), mejorando la velocidad de inferencia [30].

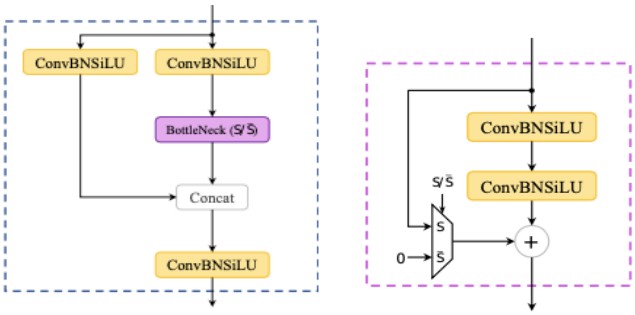


Figura 21. Arquitectura del bloque C3 y estructura Bottleneck *CSP de Darknet53. Tomado de [25].*

Todas las capas de la columna vertebral DarkNet53-CSP usan la función de activación SiLU (*Sigmoid Linear Unit*), que ayuda a la red a aprender funciones complejas, entrenar más rápido con menos recursos computacionales, y mejora la precisión, cobertura y velocidad del modelo.

5.2.2 Neck

El cuello de YOLO v5 conecta la columna vertebral con el cabezal de la red. Está compuesto por estructuras SPPF y CSP-PAN. Mientras que el módulo SPPF permite que la red capture características en diferentes niveles y escalas de abstracción, acoplando 3 capas de agregación máxima entre capas convolucionales para detectar objetos de diferentes tamaños [31], CSP-PAN (Conectividad de paso de punto de conexión), permite que los bloques de la red compartan información entre sí de forma paralela, conectando las capas de entrada, de extracción de características y de fusión de la red, permitiendo la agregación de funciones [28].

Tanto la columna vertebral como el cuello de la red, consta de muchas capas convolucionales que están estrechamente interconectadas y concatenadas [28] y emplean la función de activación SiLU.

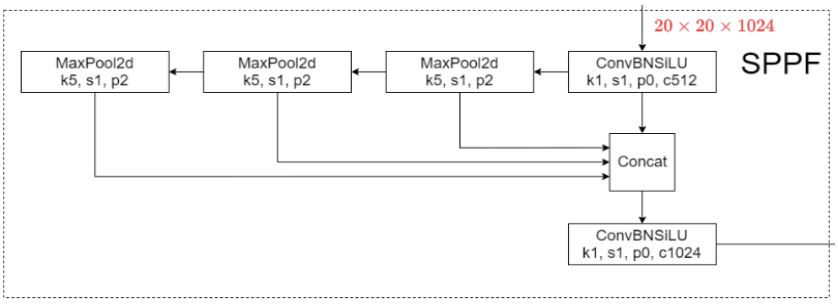


Figura 22. Estructura SPPF del cuello YOLOv5. Tomado de [29].

5.2.3 Head

El cabezal de la red está compuesto por tres capas convolucionales que predicen respectivamente la localización de los cuadros delimitadores (*x, y, w, h*), las puntuaciones de confianza y las clases de objetos.

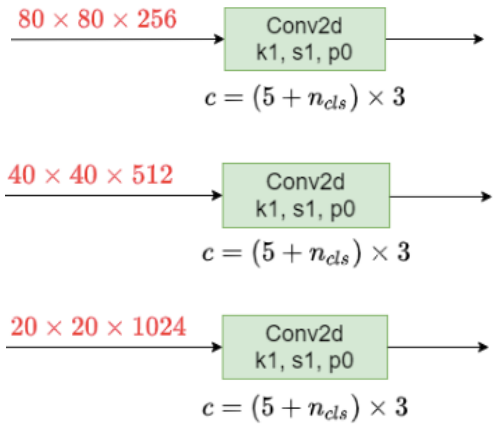


Figura 23. Capas convolucionales de salida del cabezal YOLOv5. Tomado de [29].

5.2.4 Aumento de Datos

YOLOv5 introduce durante el entrenamiento un cargador de datos que aumenta los datos en línea mediante tres tipos de aumentos: escala, ajustes de espacio y color y aumento de mosaicos. Estas técnicas de aumento de datos mejoran la capacidad del modelo para generalizar y reducir el sobreajuste.

El aumento de datos en mosaico combina cuatro imágenes en mosaicos de proporciones aleatorias, con el propósito de mejorar la precisión en la detección de objetos pequeños. El aumento de datos por copia aleatoria recorta parches de una imagen para pegarlos en otra imagen elegida al azar y generar así nuevas muestras de entrenamiento. También realiza transformaciones aleatorias por medio de rotación, escalado y corte aleatorio de las imágenes y cambios aleatorios en el tono, saturación y el valor de las imágenes [32].

5.2.5 Función de pérdida

La función de pérdida en la red YOLO se calcula como una combinación de tres componentes de pérdida individuales [26]. Pérdida de clases (*BCE*), mide el error de la tarea de clasificación; si no hay objeto la pérdida es igual a cero. Pérdida de objetividad (*BCE*), calcula el error al detectar si un objeto está presente en una celda de la cuadrícula o no. Pérdida de ubicación (*CIoU*), mide el error de localización del objeto dentro de la celda de la cuadrícula.

La función de pérdida general se representa mediante la siguiente ecuación:

$$Loss = \lambda 1 L_{class} + \lambda 2 L_{obj} + \lambda 3 L_{loc}$$

Las pérdidas de objetividad de las capas de predicción se ponderan mediante contrapesos con valores preestablecidos de 4.0, 1.0 y 0.4, de este modo se garantiza que las predicciones a diferentes escalas contribuyan al cálculo de la pérdida total.

$$L_{obj} = 4.0 * L_{small\_obj} + 1.0 * L_{medium\_obj} + 0.4 * L_{large\_obj}$$

5.2.6 Aprendizaje por transferencia

Para implementar técnicas de aprendizaje por transferencia, las capas de la columna vertebral de YOLO se pueden congelar, estableciendo sus gradientes en cero antes de empezar el entrenamiento. De esta manera, una parte de los pesos iniciales de la red se congela mientras el resto de los pesos se utiliza para calcular la pérdida, los cuales son actualizados por la función de optimización [33].

5.2.7 Hiperparámetros

5.2.7.1 Inicializar hiperparámetros

YOLOv5 cuenta con 30 hiperparámetros que se utilizan para diferentes configuraciones de entrenamiento, los cuales vienen optimizados para su correcta inicialización. La documentación recomienda iniciar con los siguientes valores predeterminados, optimizados para el entrenamiento COCO de YOLOv5 desde cero.

```
lr0: 0.01 # initial learning rate (SGD=1E-2, Adam=1E-3)
lrf: 0.01 # final OneCycleLR learning rate (lr0 * lrf)
momentum: 0.937 # SGD momentum/Adam beta1
weight_decay: 0.0005 # optimizer weight decay 5e-4
warmup_epochs: 3.0   # warmup epochs (fractions ok)
warmup_momentum: 0.8 # warmup initial momentum
warmup_bias_lr: 0.1  # warmup initial bias lr
box: 0.05 # box loss gain
cls: 0.5  # cls loss gain
cls_pw: 1.0 # cls BCELoss positive_weight
obj: 1.0 # obj loss gain (scale with pixels)
obj_pw: 1.0 # obj BCELoss positive_weight
iou_t: 0.20 # IoU training threshold
```

```
anchor_t: 4.0 # anchor-multiple threshold
              # anchors: 3  # anchors per output layer (0 to ignore)
fl_gamma: 0.0 # focal loss gamma (efficientDet default gamma=1.5)
hsv_h: 0.015  # image HSV-Hue augmentation (fraction)
hsv_s: 0.7 # image HSV-Saturation augmentation (fraction)
hsv_v: 0.4 # image HSV-Value augmentation (fraction)
degrees: 0.0   # image rotation (+/- deg)
translate: 0.1 # image translation (+/- fraction)
scale: 0.5 # image scale (+/- gain)
shear: 0.0 # image shear (+/- deg)
perspective: 0.0 # image perspective (+/- fraction), range 0-0.001
flipud: 0.0 # image flip up-down (probability)
fliplr: 0.5 # image flip left-right (probability)
mosaic: 1.0 # image mosaic (probability)
mixup: 0.0  # image mixup (probability)
copy_paste: 0.0 # segment copy-paste (probability)
```

5.2.7.3 Métrica mAP

La métrica mAP (*mean Average Precision*) se utiliza para evaluar el rendimiento de los modelos de detección de objetos. Se calcula como el promedio de la precisión media (*AP*) para cada clase de objeto. A su vez, la precisión media (*AP*) se calcula como el área bajo la curva (*AUC*) de la Precisión en función del Recall; entendida la Precisión como la proporción de objetos detectados que están correctamente etiquetados, y el Recall como la proporción de objetos correctamente etiquetados que fueron detectados. Un modelo con alto mAP es un modelo preciso y que detecta muchos objetos.

La curva ROC representa la tasa de verdaderos positivos (o instancias de objetos que se clasifican correctamente), respecto de la tasa de falsos positivos (o instancias de objetos que se clasifican incorrectamente como positivos). Comúnmente usada para evaluar el rendimiento de modelos de clasificación binaria, la curva ROC establece umbrales de clasificación.

El área bajo la curva (*AUC*) es una medida de la capacidad de los modelos de clasificación binaria para distinguir entre las clases positivas y negativas. El AUC se calcula como el área bajo la curva ROC. Un AUC de 0,5 indica que el modelo no es mejor que una predicción aleatoria, mientras un AUC cercano a 1 indica que el modelo clasifica correctamente.

Dado que un modelo de detección predice la clase y su localización en la imagen, el umbral IoU considera que una predicción es correcta únicamente si el valor de IoU es mayor a 0.5 y la clase predicha es correcta, notándose como mAP0.5 (o mAP@50); en general, el mAP se calcula para diferentes umbrales de IoU de 0.50, 0.55, 0.60… 0.95) [34].

5.3 Implementación de YOLO v5

La implementación de la red YOLO v5 consta de cinco etapas: configuración inicial, descarga del dataset etiquetado, entrenar el modelo YOLOv5 personalizado, y hacer inferencias con el modelo creado.

Para el entrenamiento e inferencia de los modelos se habilitó un entorno de programación en instancia de Amazon SageMaker Studio, con 4 vCPU, 1GPU y 16 GiB de memoria RAM (*ml.g4dn.xlarge*) para computación acelerada.

5.3.1 Configuración inicial

Como paso inicial se clonó el repositorio y se instalaron los requisitos de YOLOv5 en un entorno de Python >=3.8.0 y PyTorch >=1.8. Adicionalmente, por el entorno de programación utilizado, se importaron las librerías que permiten el uso del servicio de almacenamiento s3 de AWS.

```
[2]: !pip install -qr yolov5/requirements.txt
     import os
     import boto3
     import glob
     import datetime

     s3_resource = boto3.resource('s3')
     s3_client = boto3.client('s3')
```

Figura 24. Configuración inicial YOLOv5 en entorno de ejecución SageMaker Studio de AWS.

5.3.2 Descarga del conjunto de datos etiquetado

YOLOv5 requiere que el conjunto de datos etiquetado se transforme al formato .yalm. Se escribió una ruta desde Amazon s3 para cargar los datos, y para dividir el conjunto de datos en set de entrenamiento y de validación en el formato requerido, desde un fichero en la instancia de programación.

```
[3]: dataset_s3_uri = "s3://aca-prod-sat-data-storage/stage/dataset/"
     labels = ['Asent_informal', 'Asent_formal', 'Asent_rural', 'Entorno_natural', 'Infraest', 'Otro_rural', 'Otro_urbano', 'Esp_pu
```

Download the dataset

```
[4]: def split_s3_path(s3_path):
         path_parts=s3_path.replace("s3://","").split("/")
         bucket=path_parts.pop(0)
         key="/".join(path_parts)
         return bucket, key

     def upload_folder_to_s3(local_folder_path, s3_bucket_name, s3_prefix=''):
         for root, dirs, files in os.walk(local_folder_path):
             for file in files:
                 local_file_path = os.path.join(root, file)
                 s3_object_key = os.path.join(s3_prefix, os.path.relpath(local_file_path, local_folder_path))

                 try:
                     s3_client.upload_file(local_file_path, s3_bucket_name, s3_object_key)
                     print(f'Subido: {local_file_path} como {s3_object_key}')
                 except NoCredentialsError:
                     print("No se encontraron credenciales de AWS")
```

```
[5]: bucket,dataset_name = split_s3_path(dataset_s3_uri)
     bucket,dataset_name
```

Now let's add these data sources to the data library in the yolov5 folder for our model to train

```
[6]: with open("yolov5/data/custom-model.yaml", 'w') as target:
         target.write("path: ../{}\n".format(dataset_name))
         target.write("train: images/train\n")
         target.write("val: images/validation\n")
         target.write("names:\n")
         for i, label in enumerate(labels):
             target.write("  {}: {}\n".format(i, label))

     with open('yolov5/data/custom-model.yaml') as file:
         lines = file.readlines()
         for line in lines:
             print(line)
```

Figura 25. Carga, separación del conjunto de datos y transformación al formato YOLO

5.3.3 Entrenar el modelo YOLOv5 personalizado

A continuación se descargaron los pesos entrenados de los modelo a implementar, en este caso Yolov5x6 y Yolov5l6 desde el repositorio web de UltraLytics GitHub y se transfirieron al directorio ejecutando la siguiente línea de código:

```
!python yolov5/train.py --workers 4 --img 1280 --batch 8 --epochs 200 --data
yolov5/data/custom-model.yaml --weights yolov5x6.pt --weights yolov5l6.pt --
cache
```

Los resultados del entrenamiento se guardan en fichero creado en la instancia de SageMaker Studio en la ruta: modelo_eventos_territoriales/yolov5/runs/train

```
[15]: fecha_actual = datetime.datetime.now()
      fecha_actual_str = fecha_actual.strftime("%Y-%m-%d %H:%M:%S")
      train_results_path = 'yolov5/runs/train/exp20'
      s3_prefix = f'stage/yolov5/runs/train/{fecha_actual_str}'
      upload_folder_to_s3(train_results_path, bucket, s3_prefix)
```

Figura 26. Almacenamiento de resultados de entrenamiento en fichero.

Realizado el entrenamiento de los modelos YOLOv5x6 y YOLOv5l6 se obtuvieron los siguientes resultados:

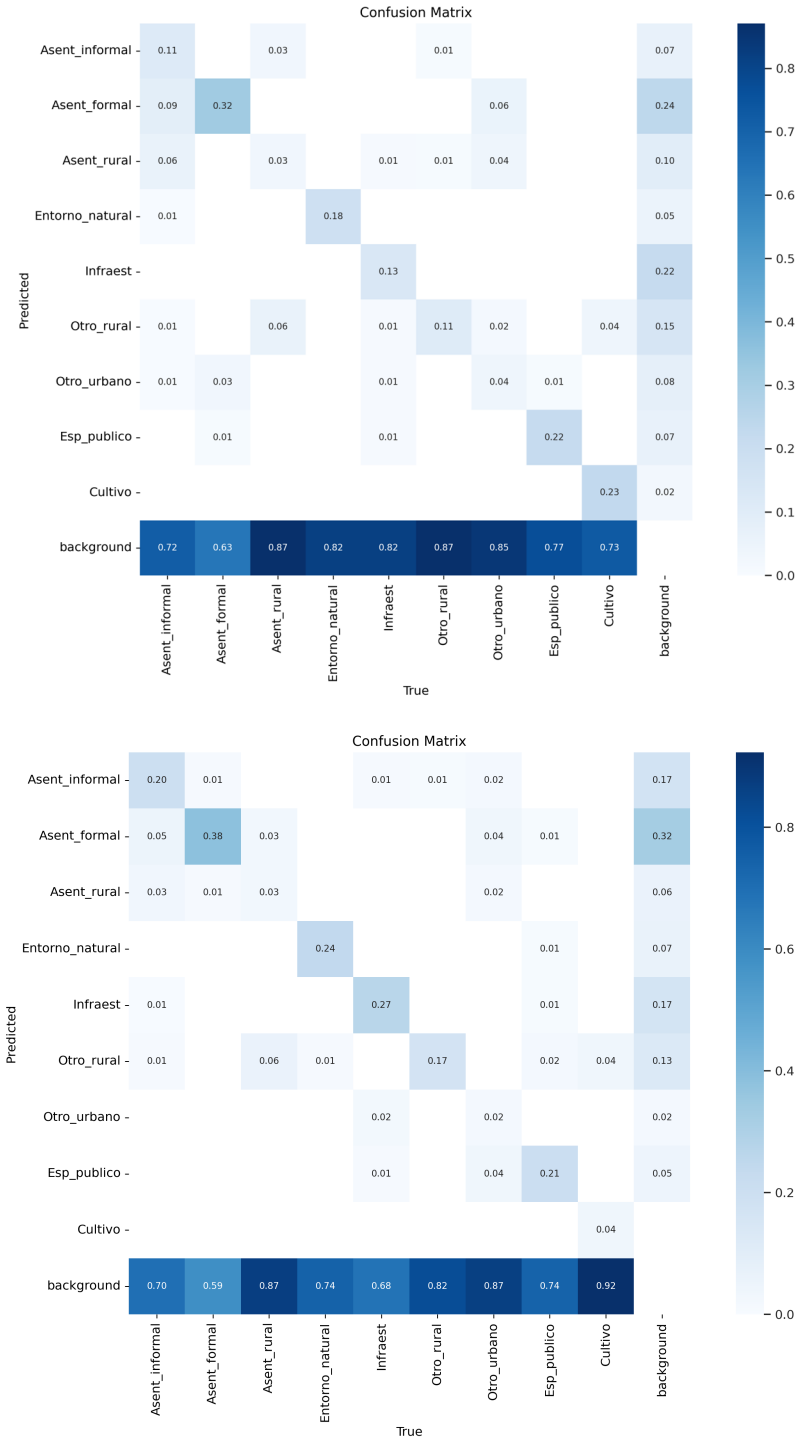


Figura 27. Matriz de confusión entrenamiento YOLOv5x6 y YOLOv5l6.

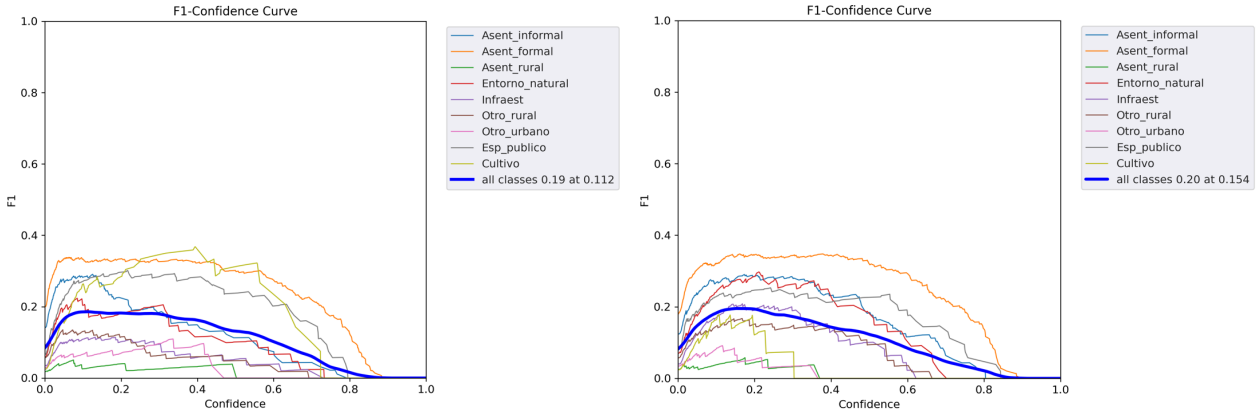


Figura 28. Curva de confianza F1 entrenamiento YOLOv5x5 vx YOLOv5l6.

La Matriz de confusión y la Curva de confianza F1 mostraron mejor correlación entre instancias reales y predichas en el modelo YOLOv5l6, para la clase de interés, asentamiento informal, y también para las clases: asentamiento formal, entorno natural, infraestructura y otro rural.

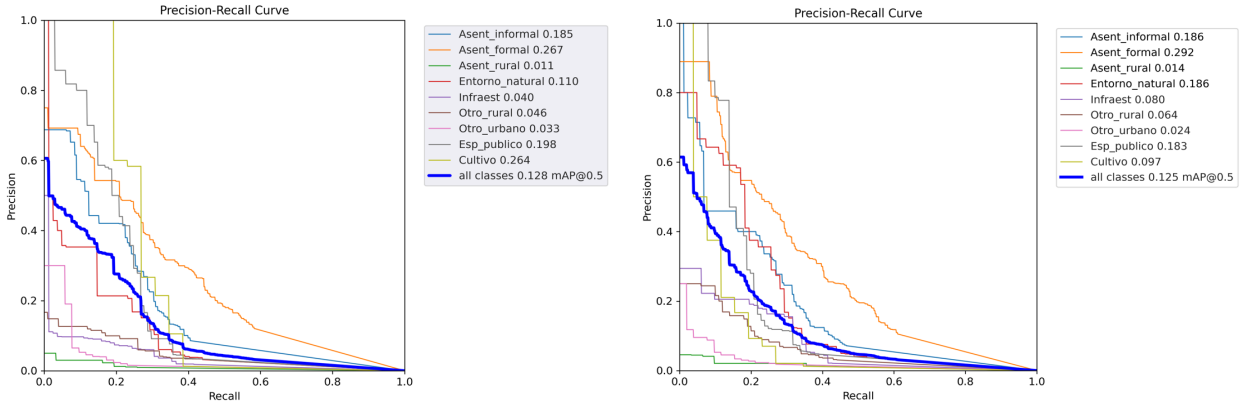


Figura 29. Curva de Precisión-Recuperación entrenamiento YOLOv5x5 vx YOLOv5l6.

5.3.4 Hacer inferencias con el modelo implementado

Se definió un umbral de confianza de 0.5 y un valor de intersección sobre la unión (IoU) de 0.6 para la predicciones correctas, evaluando el rendimiento de los modelos en el conjunto de datos de validación, a través de la siguiente línea de código:

```
!python yolov5/val.py --weights yolov5/runs/train/exp20/weights/best.pt --data
yolov5/data/custom-model.yaml --img-size 1280 --conf 0.001 --iou 0.6 --task val
```

Los resultados de la validación quedan almacenados en fichero creado dentro en la instancia de SageMaker Studio en la ruta: modelo_eventos_territoriales/yolov5/runs/val

```
[38]: fecha_actual = datetime.datetime.now()
      fecha_actual_str = fecha_actual.strftime("%Y-%m-%d %H:%M:%S")
      detect_results_path = 'yolov5/runs/val/exp3'
      s3_prefix = f'stage/yolov5/runs/val/{fecha_actual_str}'
      upload_folder_to_s3(detect_results_path, bucket, s3_prefix)
```

Figura 30. Script de almacenamiento de resultados de prueba en fichero.

Finalmente, se definió un umbral de confianza mayor al 50% para las detecciones correctas, y se ejecutó la línea de código correspondiente para acoger los pesos entrenados de mejor rendimiento durante el entrenamiento para realizar la detección.

```
!python yolov5/detect.py --weights yolov5/runs/train/exp20/weights/best.pt --
img 1280 --conf  0.5 --source "stage/dataset/images/validation"
```

Los resultados de la detección se guardan en fichero creado dentro en la instancia de SageMaker Studio en la ruta: modelo_eventos_territoriales/yolov5/runs/detect

```
[16]: fecha_actual = datetime.datetime.now()
      fecha_actual_str = fecha_actual.strftime("%Y-%m-%d %H:%M:%S")
      detect_results_path = 'yolov5/runs/detect/exp10'
      s3_prefix = f'stage/yolov5/runs/detect/{fecha_actual_str}'
      upload_folder_to_s3(detect_results_path, bucket, s3_prefix)
```

Figura 31. Script de almacenamiento de resultados de detecciones en fichero.

5.4 Resultado de los modelos YOLOv5x6 y YOLOv5l6

A continuación se presentan de forma paralela y comparativa el resultado del test de los modelos entrenados, correspondiente a los modelos YOLOv5x6 en la columa izquierda y YOLOv5l6 en la columna derecha.

5.4.1 Matriz de confusión

La matriz de confusión permite comparar las predicciones realizadas por el modelo contra las etiquetas reales de los datos de prueba.

Para la clase de interés *Asentamiento informal* y para otras clases como *Asentamiento formal*, *Entorno natural*, *Infraestructura* y *Otro rural*, el modelo YOLOv5l6 mostró mayores niveles de precisión en las detecciones; mientras que para la clase *Cultivo*, el modelo YOLOv5x6 realizó más predicciones correctas. Las clases *Otro urbano* y *Asentamiento rural* mostraron nula capacidad de clasificación, mientras que la clase *Espacio público alcanzó* nivel similar de predicciones en ambos modelos.

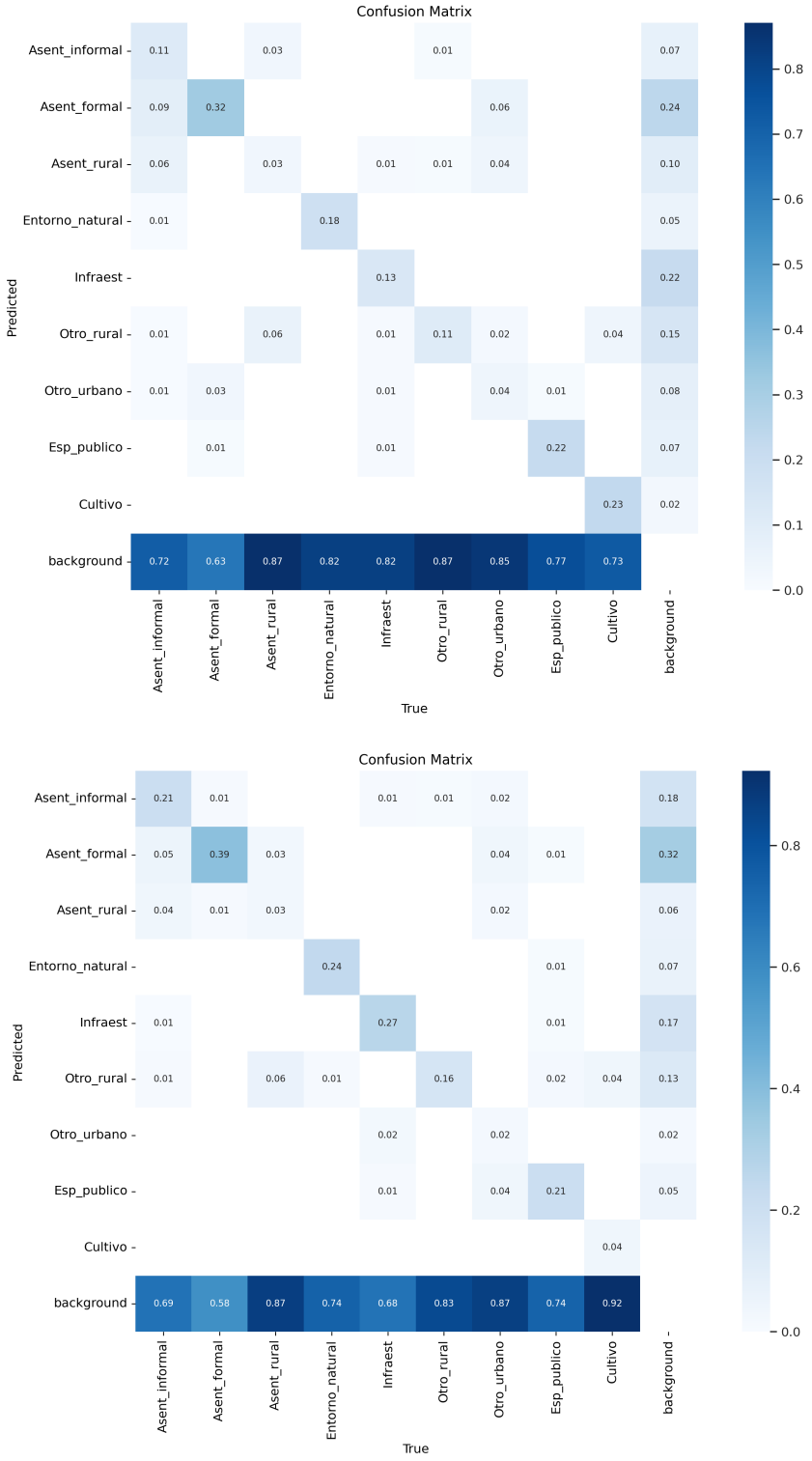


Figura 32. Matriz de confusión test YOLOv5x6 vs YOLOv5l6.

5.4.2 Curva de confianza *F1*

La curva de confianza F1 es una representación gráfica de la métrica F1 score en función de la confianza del modelo. Cada punto de la curva representa el puntaje F1 para una confianza específica.

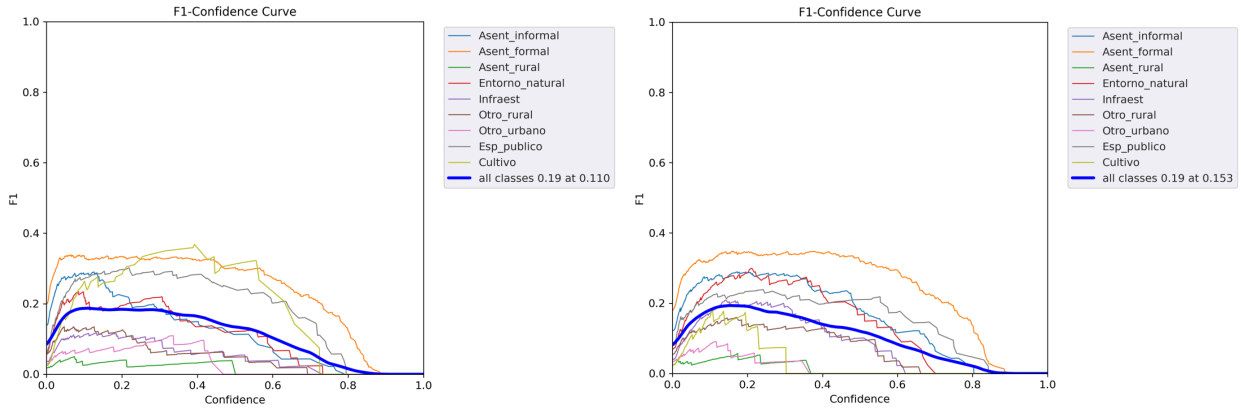


Figura 33. Curva de confianza F1 test YOLOv5x6 vs YOLOv5l6.

Las gráficas comparadas de los modelos YOLOv5x6 y YOLOv5l6 indican con claridad que en ambos modelos, clases como: *Otro urbano* y *Asentamiento rural*, tienen una confianza inferior al 50% y por lo tanto no son aceptables para la detección. Por el contrario, las clases *Asentamiento formal*, *Asentamiento informal*, *Espacio público*, *Entorno natural*, *Infraestructura* y *Espacio público*, tienen una confianza mayor al 60% de que las detecciones son correctas, aunque alcanzan bajos niveles de puntaje F1.

La curva de confianza F1 para todas las clases es similar en los dos modelos comparados, sin embargo, para la clase de interés *Asentamiento informal*, el modelo YOLOv5l6 indica mayor puntaje para la métrica F1, en niveles de confianza cercanos al 80% alcanzados en los dos modelos.

5.4.3 Curva *Precision-Recall*

Mientras que la métrica precisión mide la proporción de todos los objetos detectados que son verdaderos positivos, la métrica recuperación (*Recall*) mide la proporción de objetos verdaderos positivos que son correctamente clasificados por la herramienta de detección.

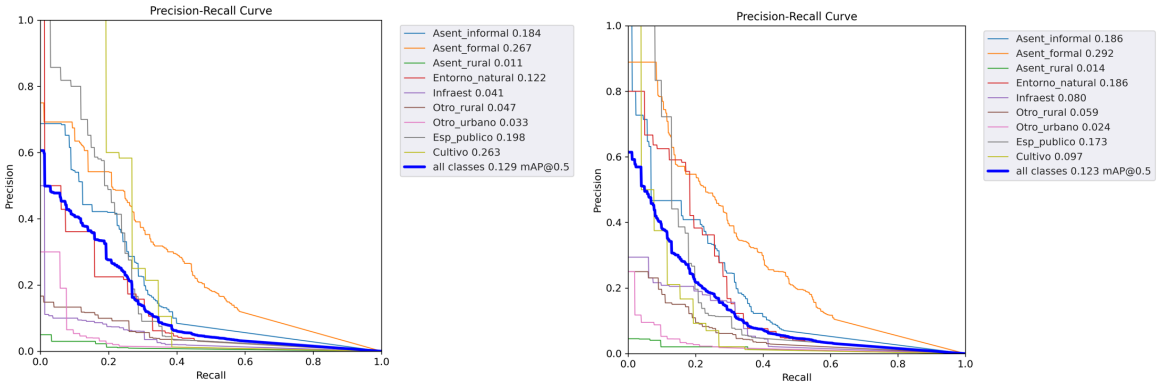


Figura 34. Curva de Precisión-Recuperación test YOLOv5x5 vs YOLOv5l6.

En general, la curva *Precision-Recall* para ambos modelos testeados indican niveles similares de la métrica promedio de la precisión media mAP:50 de 12,9% y 12,3% respectivamente, dentro del umbral de confianza definido para las predicciones correctas.

Mientras que el modelo YOLOv5x6 mostró mayor sensibilidad para detectar los objetos presentes de las clases *Asentamiento formal*, *Cultivo*, *Espacio público* y *Asentamiento informal*, en su orden; el modelo YOLOv5l6 lo fue para detectar los objetos presentes en las clases *Asentamiento formal*, *Asentamiento informal* y *Entorno natural*.

La clase de interés *Asentamiento informal* mostró mejor relación Precisión-Recall en el modelo YOLOv5l6.

5.4.4 Curva de confianza *Precision*

La curva de confianza precisión es una representación gráfica de la métrica *Precision* en función de la confianza del modelo. La precisión indica la proporción de objetos detectados que son verdaderos positivos. Se espera que en la medida que aumenta el umbral de confianza aumente, en consecuencia, la precisión del modelo.

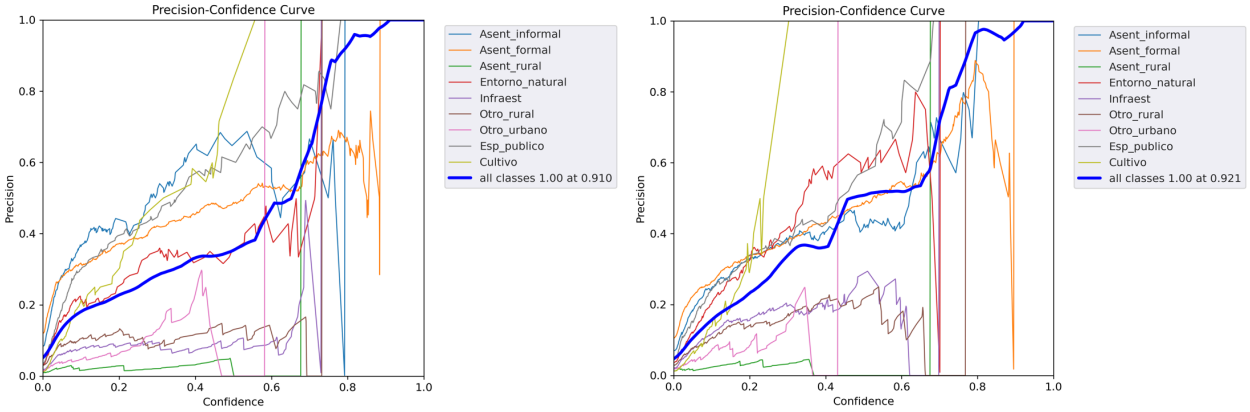


Figura 35. Curva de Precisión test YOLOv5x5 vs YOLOv5l6.

En la gráfica curva de Precisión, el comportamiento observado para todas las clases tiene un patrón similar en los modelos YOLOv5x6 y YOLOv5l6, evidenciando que las clases *Asentamiento formal* y *Asentamiento informal* presentan los mayores niveles de confianza en la precisión de las detecciones. En la misma vía, las clases *Otro urbano* y *Asentamiento rural* no alcanzan el umbral de confianza necesario para la clasificación y presentan bajo nivel de precisión en la detección. Las clases *Espacio público*, *Entorno natural*, *Infraestructura* y *Otro rural* son detectados con precisión por ambos modelos dentro del umbral de confianza mayor al 60%.

5.4.5 Curva de confianza *Recall*

La curva de confianza recuperación es una representación gráfica de la métrica *Recall* en función de la confianza del modelo. Se espera que en la medida que aumenta la confianza también aumente el *Recall*, lo cual indicaría que existe mayor probabilidad de que el modelo detecte correctamente un objeto.

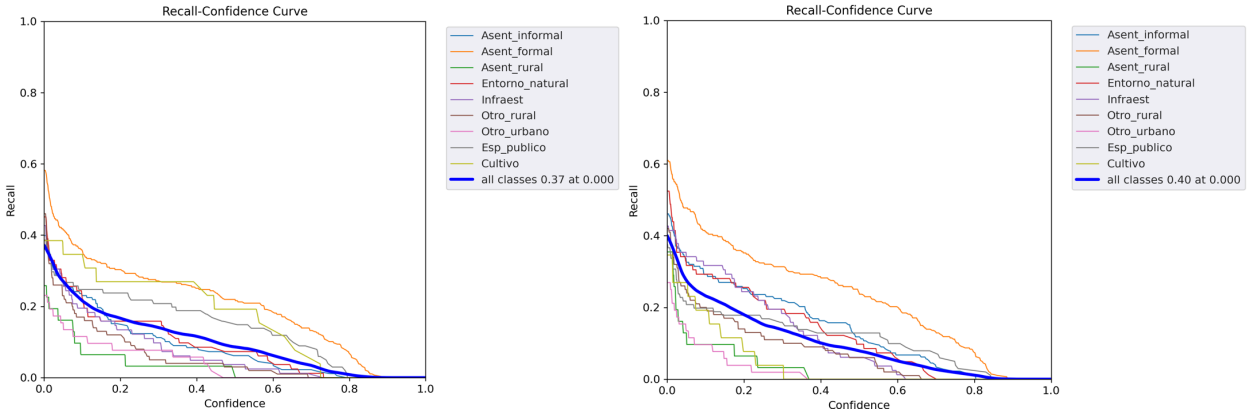


Figura 36. Curva de Recuperación test YOLOv5x5 vs YOLOv5l6.

Las gráficas de confianza recuperación muestran que las clases *Asentamiento formal*, *Asentamiento informal*, *Espacio público*, *Entorno natural*, *Otro rural* e *Infraestructura*, se detectan correctamente en ambos modelos en niveles de confianza mayores al 60%, siendo el modelo YOLOv5l6 más sensible para la clasificación correcta de las clases Asentamiento formal, Asentamiento informal y Espacio público en niveles de confianza mayores al 80%.

5.4.5 Valores de pérdida

Los valores de pérdida de objetividad, pérdida de clase y pérdida de caja indican si el modelo predice con precisión la confianza de los objetos, las clases de los objetos, y la ubicación y el tamaño de las cajas delimitadoras respectivamente. Si el modelo YOLOv5 tiene una pérdida ponderada baja, presentará un buen rendimiento en la detección de objetos en imágenes.

Como se observa en la figura 37, los valores de la función de pérdida durante el entrenamiento y validación de los modelos YOLOv5x6 y YOLOv5l6 muestran una tendencia similar hacia el

sobreajuste en los valores de *obj_loss*, *cls_loss* y *box_loss*, manteniendo métricas muy cercanas en los valores mAP:50 entre el 10% y el 12% y de mAP:50:95 entre el 3% y el 4%.

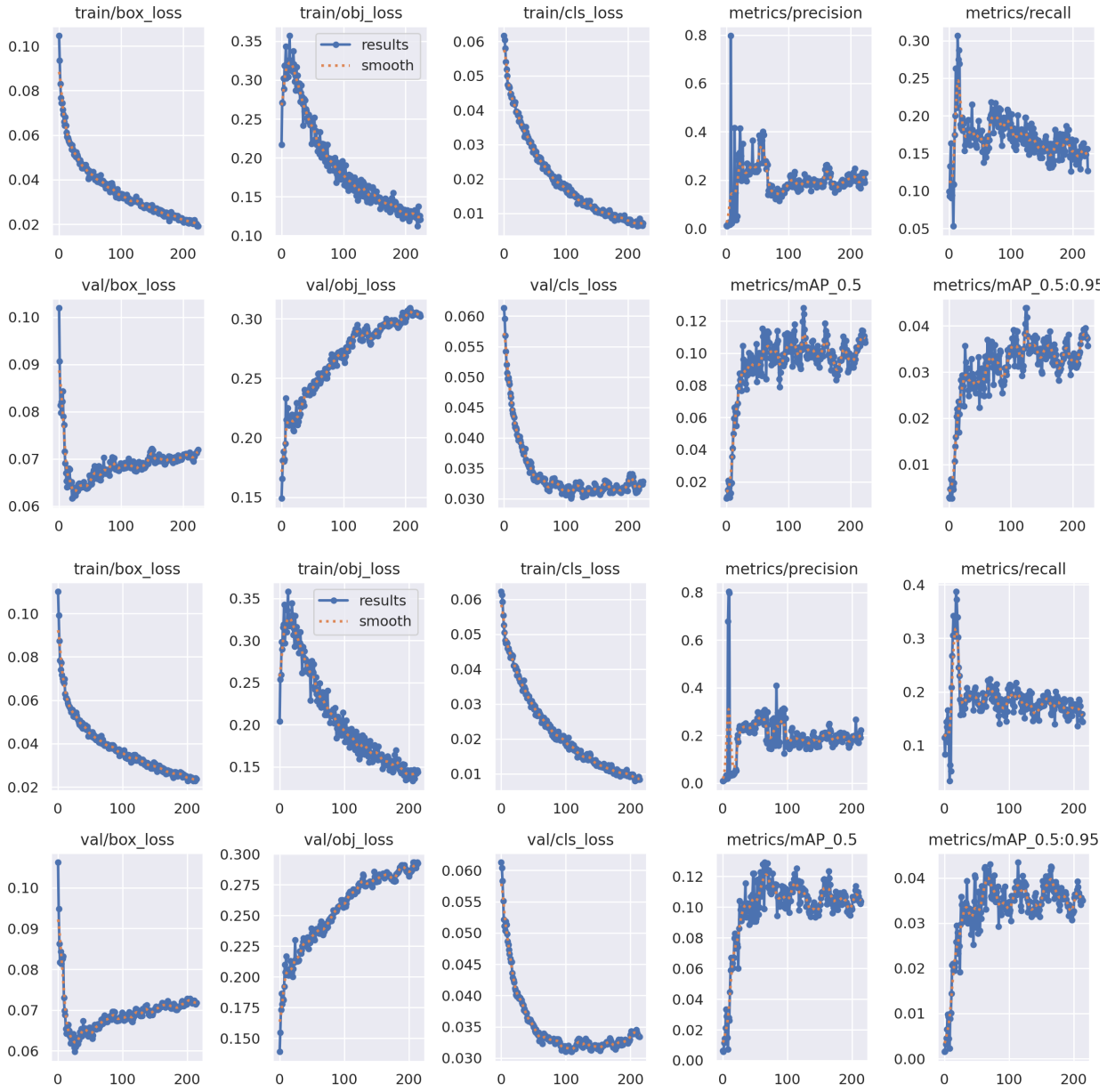


Figura 37. Valores de la función de pérdida y métrica mAP, YOLOv5x5 vs YOLOv5l6.

5.4.6 Resultados del test en modelo YOLOv5l6 de mejor rendimiento.

Para la clase *Asentamiento informal*, el resultado de las pruebas de validación del modelo YOLOv5l6 muestran niveles de precisión del 30,8% y de recuperación del 27%, mientras el promedio de la precisión media mAP:50 fue de 18,6%; la clase *Asentamiento formal*, mostró valores de precisión del 31,8% y de recuperación del 37,3%, con el valor más alto de mAP:50 del 29,2%.

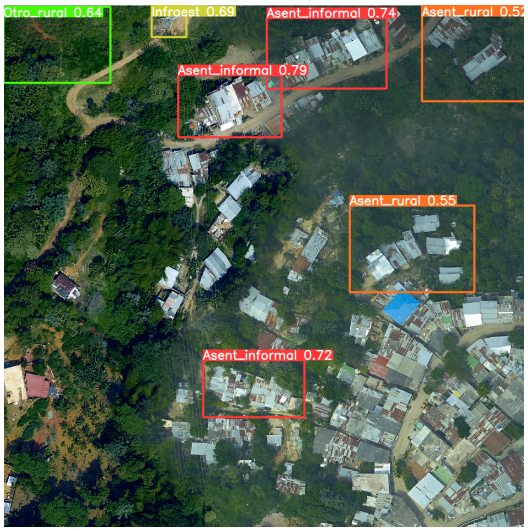
```
                 Class     Images  Instances          P          R      mAP50   libpng warning: iCCP: known incorrec
t sRGB profile
                 Class     Images  Instances          P          R      mAP50
                   all         34        939      0.205        0.2      0.123     0.0433
          Asent_informal         34        178      0.308       0.27      0.186     0.0581
            Asent_formal         34        287      0.318      0.373      0.292     0.0977
             Asent_rural         34         31     0.0367     0.0968     0.0137    0.00428
         Entorno_natural         34         82      0.269      0.268      0.186      0.057
                Infraest         34         82       0.16      0.291       0.08     0.0265
             Otro_rural          34        100      0.149       0.17     0.0585     0.0146
            Otro_urbano          34         52     0.0631     0.0385     0.0242    0.00946
            Esp_publico          34        101      0.302      0.178      0.173      0.078
                Cultivo          34         26      0.243      0.115     0.0966     0.0438
Speed: 2.3ms pre-process, 108.9ms inference, 32.7ms NMS per image at shape (32, 3, 1280, 1280)
Results saved to yolov5/runs/val/exp3
```

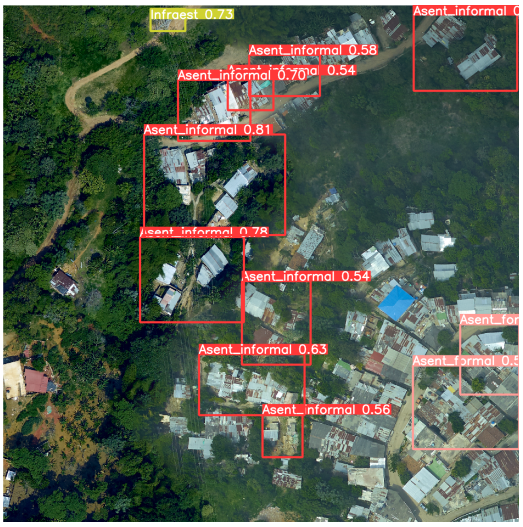
Figura 38. Resultados del test de validación YOLOv5l6.

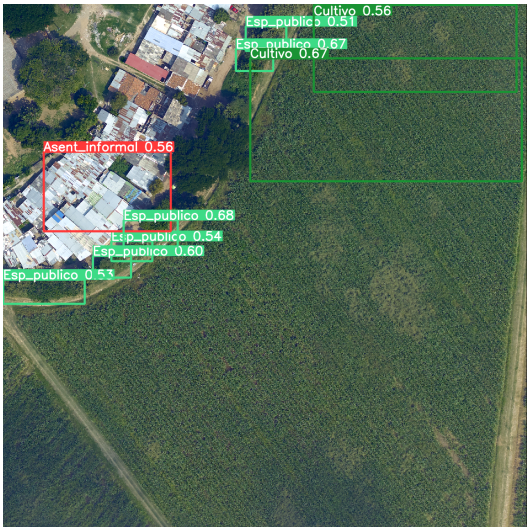
5.4.6 Inferencia en imágenes de prueba

Al comparar las inferencias realizadas en 34 imágenes de prueba, especialmente de la clase de interés *Asentamiento informal*, se evidencia que el modelo YOLOv5l6 realizó mayores y mejores detecciones. A continuación se presenta una muestra de las detecciones realizadas por ambos modelos, YOLOv5x6 en la columna izquierda y YOLOv5l6 en la columna de la derecha.

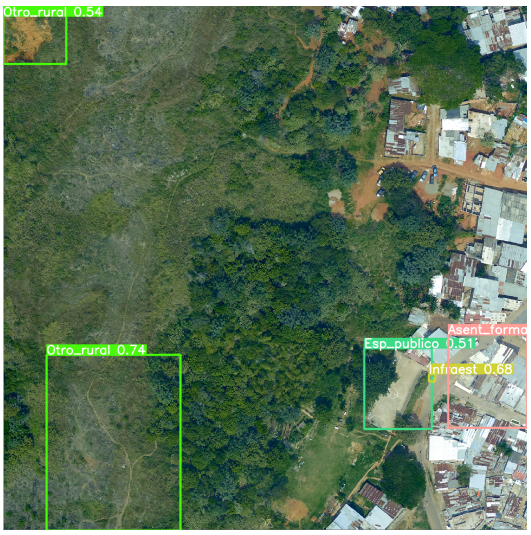
Las detecciones de instancias de objetos en las inferencias realizadas por los modelos mostraron que YOLOv5l6 es más sensible y preciso para detectar la clase de interés *Asentamiento informal*, así como para detectar las clases *Asentamiento formal*, *Entorno natural* y *Otro rural*, mientras que el modelo YOLOv5x6 se mostró más sensible para detectar las clases *Cultivo*, *Espacio público*, *Infraestructura* y *Asentamiento rural*.

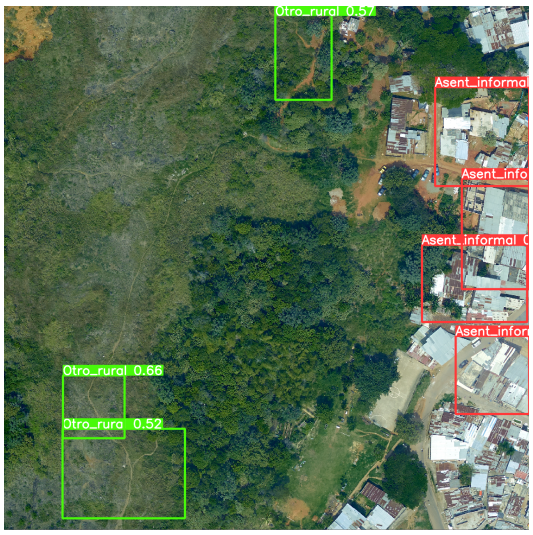

Otro_rural 0.64
Infraest 0.69
Asent_informal 0.74
Asent_rural 0.52
Asent_informal 0.79
Asent_rural 0.55
Asent_informal 0.72

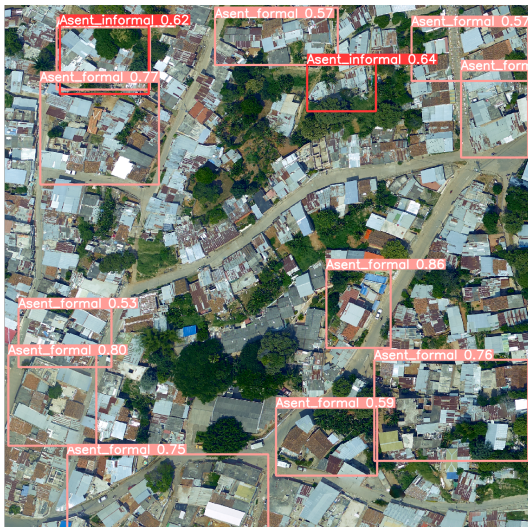

Infraest 0.73
Asent_informal 0.58
Asent_informal 0.70
Asent_informal 0.81
Asent_informal 0.54
Asent_informal 0.63
Asent_informal 0.56

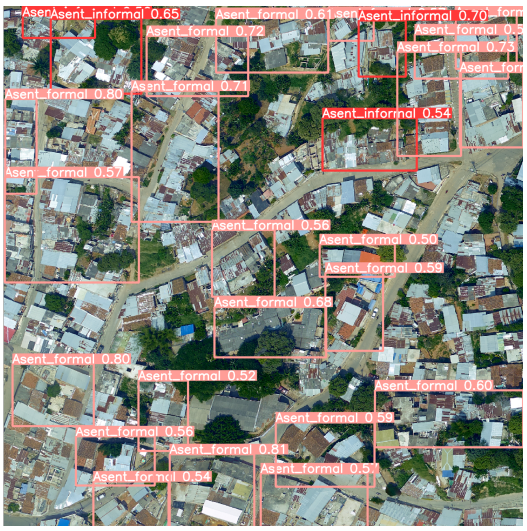

Cultivo 0.56
Esp_publico 0.51
Esp_publico 0.67
Cultivo 0.67
Asent_informal 0.56
Esp_publico 0.68
Esp_publico 0.54
Esp_publico 0.60

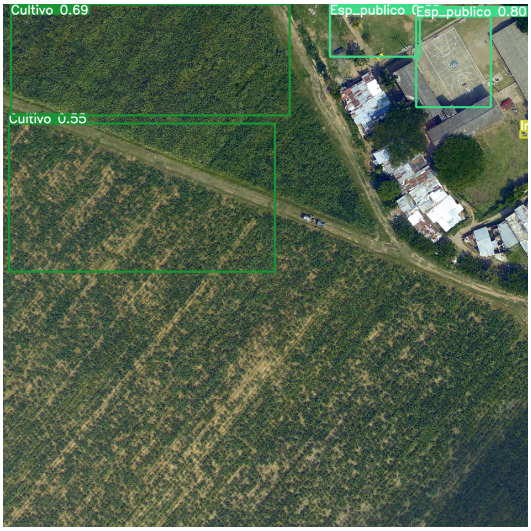

Asent_informal 0.71
Asent_informal 0.60
Asent_informal 0.69
Asent_informal 0.73

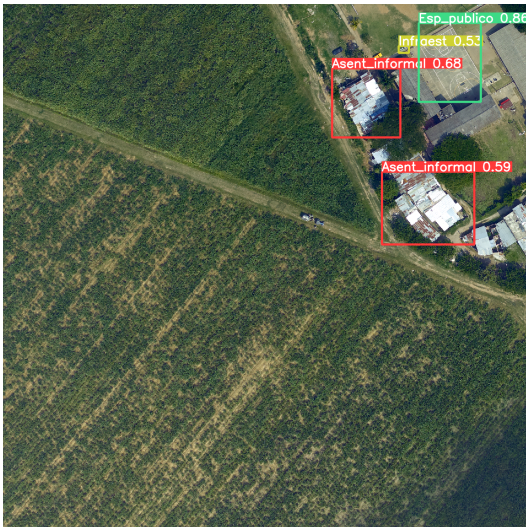

Otro_rural 0.54
Otro_rural 0.74
Esp_publico 0.51
Infraest 0.68

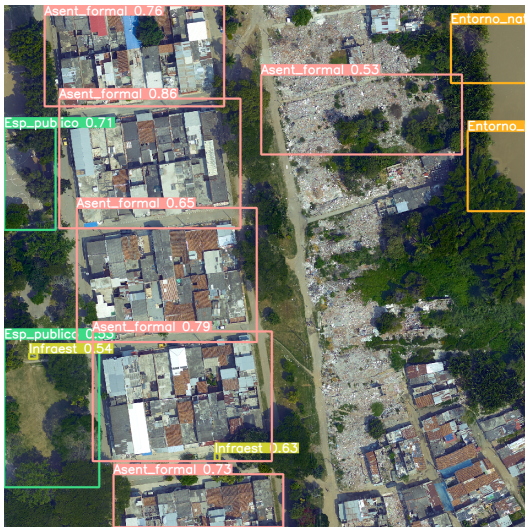

Otro_rural 0.57
Otro_rural 0.66
Otro_rural 0.52

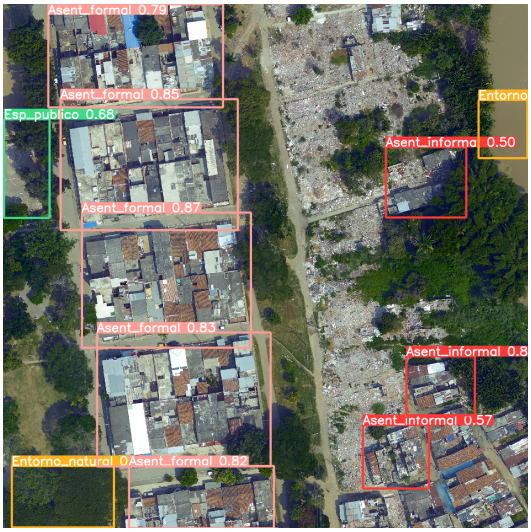

Asent_informal 0.62
Asent_formal 0.57
Asent_informal 0.64
Asent_formal 0.77
Asent_formal 0.86
Asent_formal 0.53
Asent_formal 0.80
Asent_formal 0.76
Asent_formal 0.59
Asent_formal 0.75


Asent_informal 0.65
Asent_formal 0.72
Asent_formal 0.80
Asent_informal 0.54
Asent_formal 0.56
Asent_formal 0.50
Asent_formal 0.59
Asent_formal 0.68
Asent_formal 0.80
Asent_formal 0.52
Asent_formal 0.60
Asent_formal 0.56
Asent_formal 0.81


Cultivo 0.69
Esp_publico 0.80
Cultivo 0.55


Esp_publico 0.86
Infraest 0.53
Asent_informal 0.68
Asent_informal 0.59


Asent_formal 0.76
Asent_formal 0.53
Asent_formal 0.86
Esp_publico 0.71
Asent_formal 0.65
Asent_formal 0.79
Infraest 0.54
Infraest 0.63
Asent_formal 0.73

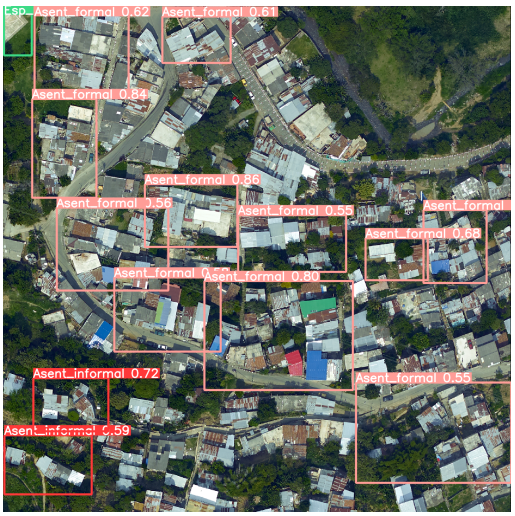

Asent_formal 0.79
Asent_formal 0.85
Esp_publico 0.68
Asent_informal 0.50
Asent_formal 0.87
Asent_formal 0.83
Asent_informal 0.57
Asent_formal 0.82


Asent_informal 0.7
Asent_rural
Otro_rural 0.53


Asent_informal 0.5
Asent_rural 0.50
Otro_rural 0.59
Asent_informal 0.52


Asent_rural 0.61


Otro_rural 0.56
Otro_rural 0.69
Asent_informal 0.68


Asent_formal 0.62
Asent_formal 0.61
Asent_formal 0.84
Asent_formal 0.86
Asent_formal 0.55
Asent_formal 0.55
Asent_formal 0.68
Asent_formal 0.80
Asent_informal 0.72
Asent_formal 0.55


Asent_formal 0.59
Asent_formal 0.65
Asent_formal 0.59
Asent_formal 0.65
Asent_formal 0.65
Asent_informal 0.58
Asent_informal 0.54
Asent_informal 0.66
Asent_formal 0.51
Asent_informal 0.53
Asent_informal 0.68
Asent_informal 0.63
Asent_informal 0.72
Asent_informal 0.66

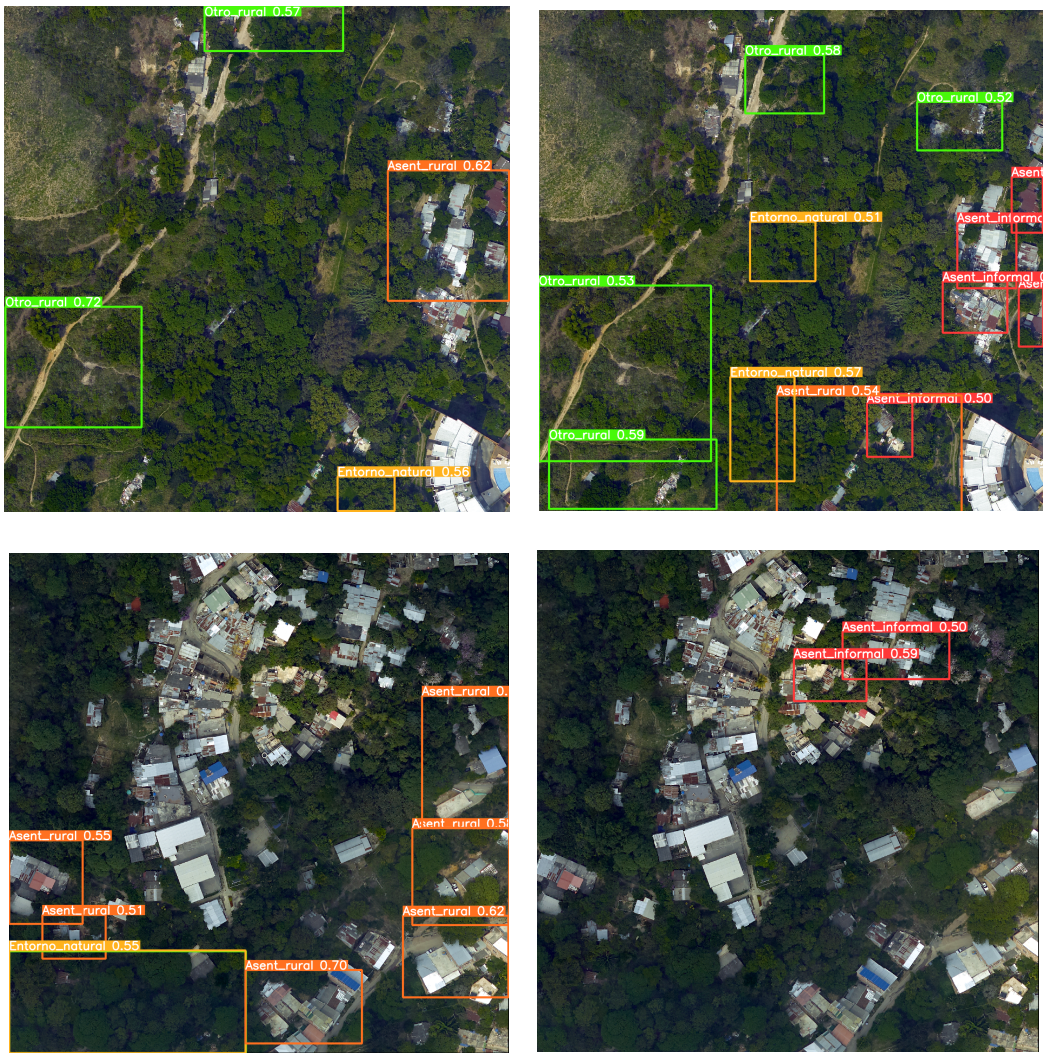


Figura 39. Detecciones de clase Asentamiento informal YOLOv5x5 vx YOLOv5l6.

5.5 Exploración de técnicas para mejorar detección de la clase de interés

Una vez identificado que el modelo YOLOv5l6 presentó los mejores resultados usando la totalidad de sus pesos preentrenados, se exploró una técnica de ajuste fino (*fine tuning*) para probar una mejora de rendimiento en la herramienta de detección para la clase de interés *Asentamiento informal*.

Específicamente se congelaron 10 capas convolucionales y bloques C3 de la columna vertebral (*backbone*) de YOLOv5, con el propósito permitirle al modelo actualizar los pesos en su capas superiores y adaptarse al conjunto de datos de entrenamiento, manteniendo los hiperparámetros de inicialización.

```
!python yolov5/train.py --workers 4 --img 1280 --batch 8 --epochs 200 --freeze
10 --data yolov5/data/custom-model.yaml --weights yolov5l6.pt –cache
```

5.5.1 Congelar capas del backbone de YOLOv5l6

Para la clase de interés *Asentamiento informal* y para la clase *Asentamiento Formal*, el resultado del entrenamiento con capas congeladas no mejoró la capacidad de detección de la herramienta; por el contrario, disminuyeron las métricas *precisión*, *recall* y *mAP:50* respecto de los niveles alcanzados por el modelo YOLOv5l6 con todos los pesos pre entrenados.

Para las clases: *Espacio público* y *Entorno natural*, disminuyeron las métricas precisión y mAP:50, mejorando la recuperación; la clase *Cultivo* fue la única que mejoró todas sus métricas con la técnica aplicada.

Clase	Modelo YOLOv5	Metrics/ Precision	Metrics/ Recall	Metrics/ mAP:50	Metrics/ mAP50:95
Asentamiento informal	l6	0.308	0.27	0.186	0.0581
	l6_freeze	0.305	0.219	0.176	0.0502
Asentamiento formal	l6	0.318	0.373	0.292	0.0977
	l6_freeze	0.276	0.331	0.258	0.092
Espacio público	l6	0.302	0.178	0.173	0.078
	l6_freeze	0.248	0.257	0.165	0.073
Cultivo	l6	0.243	0.115	0.0966	0.0438
	l6_freeze	0.276	0.115	0.123	0.0341
Entorno natural	l6	0.269	0.268	0.186	0.057
	l6_freeze	0.197	0.305	0.144	0.0351

Tabla 2. Métricas de clase entrenamiento YOLOv5l6 y YOLOv5l6 freeze

En todas las clases disminuyó la métrica mAP:50:95, manteniendo bajo el nivel de precisión en la detección de objetos territoriales durante el entrenamiento del modelo

Esta técnica si bien redujo el tiempo de entrenamiento del modelo, se mostró con menor capacidad para detectar la clase de interés, mostrando niveles de mAP:50 inferiores a los obtenidos durante el entrenamiento previo, donde se alcanzaron niveles de mAP:50 para la clase *Asentamiento informal* de 18,6%, para la clase *Asentamiento formal* de 29,2% y para todas las clases en general de 12,3%.

5.5.2 Aumento de imágenes

Por medio del aumento de imágenes se crean nuevos ejemplos de entrenamiento a partir de los existentes, aplicando cambios sobre la imagen original. Esta técnica es usada en modelos de redes neuronales profundas para mejorar resultados en la clasificación y detección de objetos y para reducir el sobreajuste [35].

YOLOv5 incluye dentro de sus parámetros de inicialización, 13 relacionados con el aumento de imágenes, los cuales fueron mencionados en el punto 5.2.7.1. Estos hiperparámetros configuran el aumento de datos por saturación, rotación, escala, perspectiva, parches y mosaicos durante el proceso de entranamiento del modelo, los cuales se imprimen en la ejecución del script como train_batch, tal como muestra en la siguiente Figura 40.

Figura 40. Aumento de imágenes en el train_batch YOLOv5l6.

Igualmente, YOLOv5 se integra por defecto con la biblieteca de Python Albumentations para el aumento de imágenes, lo que permite al modelo durante el entrenamiento, agregar nuevos canales de aumento de imégenes.

Los resultados obtenidos en los modelos YOLOv5x6, YOLOv5l6 y YOLOv5l6_freeze aplicaron la técnica de aumento de imágenes durante su entrenamiento, por lo que no es fue posible mejorar el rendimiento en la detección de la clase de interés.

Otras técnicas de ajuste fino como son aumentar el número de iteraciones durante el entrenamiento o ajustar la tasa de prendizaje durante el aprendizaje tampoco fueron aplicadas porque YOLOv5 ha sido optimizado para aplicar *early stopping* después de 100 épocas sin mejoras de rendimiento y para ajustar el *learning rate* inicial durante el entrenamiento (lr0, lr1 y lr2).

6. EVALUACIÓN DE LA CAPACIDAD DE DETECCIÓN YOLOv5

6.1 Métricas de detección

La métrica mAP:50:95 se emplea para evaluar rendimiento de modelos de detección de objetos en imágenes. Esta se calcula como la media de las áreas bajo la curva (AUC) de los valores de la precisión media (AP), a dos umbrales de confianza en la detección (IoU): 0.5 y 0.95.

Mientras la precisión (AP:50) representa el porcentaje de detecciones del modelo que tienen intersección sobre la unión (IoU) con el objeto real de al menos el 0.5; la precisión (AP:95) representa el porcentaje de detecciones del modelo que tienen intersección sobre la unión (IoU) con el objeto real de al menos el 0.95.

Los valores de la métrica mAP:50:95 indican entonces la capacidad del modelo para la detectar de forma precisa los objetos presentes en las imágenes. A mayor valor mAP, mayor capacidad de detección precisa.

Sin embargo, un criterio para evaluar la capacidad de un modelo de detección de objetos en imágenes por medio de la métrica mAP:50 y mAP:50:95, debe considerar el tipo de objetos que se propone clasificar y la base de datos utilizada para el entrenamiento, ya que estos elementos determinan la magnitud de la tarea a realizar.

6.2 Criterios de evaluación

6.2.1 Clasificación de objetos territoriales

A diferencias de otras instancias de objetos como son personas, animales, automóviles, células o señales de tránsito, un evento territorial como *asentamiento informal* representa una instancia de objeto compleja, compuesta por múltiples elementos como son estructuras de diferentes elevaciones y texturas, vegetación, vías y una disposición en el espacio, de tipo orgánica cuando está localizada zona de montaña o de tipo ortogonal cuando está localizada en zona plana.

De acuerdo con investigaciones recientes [27], los asentamientos informales en la ciudad de Cali ocupan un total 606,4 hectáreas, el 55% de las cuales está localizada dentro del perímetro urbano y el restante 45% en zona rural.

Se puede afirmar entonces que este fenómeno territorial, abordado como objeto terrestre, representa un desafío para la tarea de detección mediante herramientas de aprendizaje profundo.

Dado que se trata de un fenómeno de ocupación del suelo de calado histórico, amplia escala y en continua expansión, que va desde asentamientos consolidados en proceso de legalización que cuentan con infraestructura de servicios públicos[3], hasta asentamientos denominados "invasiones", de aparición reciente y localizadas en zonas de protección ecológica, la detección y clasificación de este fenómeno territorial requiere de un ejercicio de delimitación más exhaustivo que responda a problemas de negocio concretos. Por ejemplo, identificar asentamientos informales rurales en zona de protección de cuencas hídricas o ambientales; asentamientos informales en sectores de consolidación urbanos; asentamientos informales en zona de riesgo natural. Esta dilimitación de la clase de interés permitirá que se identifiquen mejor las características espaciales de este objeto terrestre.

Otras clases de objetos, denominados en este estudio *fenómenos territoriales*, como *asentamiento formal*, *entorno natural* o *espacio público* se encuentran en el mismo nivel de complejidad y desafio para la tarea de clasificación. Por lo mismo, deben acotarse sus atributos para detectar por ejemplo, un tipo de espacio público particular bien sean canchas de fútbol, parques, o andenes.

El desafío principal consiste en afinar la verdad terrestre (*ground truth*). En este caso, puede afirmarse que mejorar los niveles de detección de la clase de interés y demás clases de objetos territoriales, pasa necesariamente por la revisión experta del proceso de etiquetado manual de las clases de interés, previo al ejercicio de entrenamiento de modelos.

6.2.2 Base de datos de imágenes de Santiago de Cali

La base de datos elaborada con 166 imágenes etiquetadas de Santiago de Cali es reducida para el entrenamiento de una herramienta de detección que debe aprender características complejas de un objeto terrestre compuesto por varios elementos. Tanto por el número de imágenes disponibles como por el número de instancias de objetos de la clase de interés, la base de datos requiere ser ampliada para contar con mayor número de imágenes.

Los bajos niveles de precisión en la detección de la clase de interés y de las otras clases de fenómenos territoriales no se puede subsanar aplicando técnicas de *data augmentation*, sino que requiere de ampliar la disponibilidad de imágenes.

Ampliar la disponibilidad de imágenes VHR se puede realizar de dos maneras: primero, segmentando las imágenes existentes a escala más pequeña, en dimensiones de 256 X 256 píxeles

[3] Según noticia publicada por el diario El País, sectores como Alto Polvorines, La Arboleda, Pampas del Mirador y Brisas de las Palmas, considerados Asentamientos Humanos de Desarrollo Incompleto, fueron legalizados como barrios por la alcaldía de Santiago de Cali el 18 de noviembre de 2023, en el marco de la Política Pública de Mejoramiento Integral de Hábitat. "¿Por qué la Alcaldía de Cali legalizó ocupaciones históricas?" El País. 25 de noviembre de 2023. [Online]: https://www.elpais.com.co/cali/por-que-la-alcaldia-de-cali-legalizo-ocupaciones-historicas-en-la-comuna-18-2549.html

como en la base de datos ImageNet, o en dimensiones 320 X 320 píxeles, como en la base de datos MS COCO; en el primer caso se multiplicaría por 5 y en el segundo por 4 la disponibilidad de imágenes. Sin embargo, YOLOv5 acepta imágenes de entrada de 640 X 640 píxeles, por lo tanto podría ampliarse al doble la base de datos de imágenes de Santiago de Cali existente. Segundo, procesando imágenes existentes en formato GeoTIFF para disponibilizarlas en el formato requerido como entrada del modelo. Las dos opciones son necesarias para contar con suficientes datos de entrenamiento de modelos de detección de objetos terrestre en imágenes satelitales.

6.3 Evaluación de resultados modelo YOLOv5l6 en detección de fenómenos territoriales

El valor de la métrica (mAP:50) de precisión media en la detección para todas las clases fue de 12,3%, mientras que para la clase de interés *Asentamiento Informal* fue del 18,6%. El rendimiento general del modelo para umbrales de confianza en la detección (mAP:50:95) fue del 4,33%, mientras que para la calse de interés es del 5,81%.

La mayor sensibilidad y precisión del modelo se logró para la clase *Asentamiento formal*, que alcanzó una métrica de precisión media (mAP:50) del 29,2% y global (mAP:50:95) de 9,77%

Clase	Imágenes	Instancias	Metrics/ Precision	Metrics/ Recall	Metrics/ mAP:50	Metrics/ mAP50:95
Asentamiento informal	34	178	0.308	0.27	0.186	0.0581
Asentamiento formal	34	287	0.318	0.373	0.292	0.0977
Espacio público	34	101	0.302	0.178	0.173	0.078
Entorno natural	34	82	0.269	0.268	0.186	0.057
Infraestructura	34	82	0.16	0.291	0.08	0.0265
Cultivo	34	26	0.243	0.115	0.0966	0.0438

Tabla 3. Métricas de confianza en la detección de clases de objetos territoriales YOLOv5l6

La Tabla 3 muestra que los resultados alcanzados en la métrica mAP, en cada una de las clases, están directamente relacionados con el número de instancias de objetos de verdad terrestre en las imágenes testeadas por el modelo. De ese modo se explica que los valores más altos de mAP se encuentre en las clases con mayor representación de instancias, en este caso, *asentamiento formal* y *asentamiento informal*.

Sin embargo, para las clases *Infraestructura* y *Cultivo*, la relación entre número de instancias y valor de mAP no es coincidente. Esto se explica porque la clase con menor número de instancias, *cultivo*, es identificada por el modelo como una trama regular y uniforme, mientras que la clase

con más instancias, *infraestructura*, identifica diferentes tipos de objetos como torres de conducción eléctrica urbanos y rurales, postes de luz urbanos, incluso puentes. Se colige que, a mayor homegeneidad en los atributos de verdad terrestre de un objeto mejora la precisión de las detecciones.

En general, los resultados alcanzados con YOLOv5l6 para la detección de fenómenos territoriales fueron bastante modestos, comparados con los valores mAP:50 y mAP:50:95 alcanzados por modelos YOLOv5 en conjuntos de imágenes de entrenamiento de gran escala, como los que se indican en la Tabla 4.

Conjunto de datos	mAP50	mAP50:95
COCO	93,60%	60,10%
VOC2007	88,90%	47,40%
PASCAL VOC 2012	87,50%	44,10%

Tabla 4. Valores óptimos alcanzados por YOLOv5 en distintos conjuntos de datos

La evaluación de rendimiento del modelo YOLOv5 implementado indica que la vía para mejorar la precisión en la detección automática de fenómenos territoriales en imágenes es delimitar exhaustivamente la verdad terrestre de la clase de interés y aumentar el conjunto de imágenes de Santiago de Cali disponibles para ampliar la base de datos con un mayor número de instancias de objetos etiquetados.

7. CONCLUSIONES Y TRABAJOS FUTUROS

7.1 Conclusiones

Este proyecto de máster en ciencia de datos implementó un modelo de redes neuronales completamente convolucionales llamado YOLOv5 (*You Only Look Once versión 5*), para la detección de fenómenos territoriales, entendidos como espacios físicos donde ocurren dinámicas de interacción sociales, culturales y ambientales, de especial interés para el ordenamiento del territorio.

Se propuso abordar el reto como un problema de clasificación de objetos en imágenes, aprovechando fuentes de inforaciόm de muy alta resolución (VHR) espacial y algoritmos de aprendizaje profundo pre entrenados en conjuntos de imágenes de gran escala para aprovechar los pesos optimizados en la detección de múltiples categorías de objetos.

Se definió como clase de interés *asentamientos informales*, que desde la perspectiva institucional se denominan Asentamientos Humanos de Desarrollo Incompleto (AHDI), cuyas características principales son la precariedad de las viviendas, la tenencia irregular del suelo y la carencia de infraestructura de servicios públicos y equipamientos básicos. Como base de conocimiento experto sobre este fenómeno territorial se acogieron las descripciones y los análisis físico-espaciales de AHDI realizados [27] en el marco de la revisión y actualización de la Política Pública municipal de Mejoramiento Integral del Hábitat en 2020.

El primer reto fue elaborar una base de datos con imágenes etiquetadas de fenómenos territoriales con suficiente representación de la clase de interés. Se transformó una ortofotografía de gran escala, en 166 imágenes en formato PNG, de tamaño 1280 X 1280 píxeles, etiquetadas con 7 instancias de objetos territoriales. El principal desafío abordado fue el procesamiento de información geográfica asociada a la imagen fuente y su transformación necesaria para cumplir con los parámetros de entrada de una red neuronal.

Un aprendizaje del ejercicio realizado, que incluyó tareas de segmentación de la imagen fuente, selección de imágenes con atributos geográficos acordes con la representación de la clase de interés y con el conocimiento experto sobre el fenómeno, transformación de formato de las imágenes seleccionadas, segmentación de las imágenes al tamaño aceptado por el modelo, y el etiquetado manual de todas las instancias de objetos de acuerdo con las variables de respuesta, fue que el conjunto de imágenes requerido para el entrenamiento de una herramienta de detección de objetos debe ser amplio en número de imágenes y de instancias de objetos, y

requiere de un ejercicio exhaustivo para acotar más la verdad terrestre de cada uno de los fenómenos territoriales.

El segundo reto fue implementar la red YOLO, popular para la detección de objetos en movimiento en el campo de *computer vision*, pero sin antecedentes de uso documentados para la clasificación de fenómenos territoriales, como los asentamientos informales. La principal motivación de implementar la red YOLO fue compensar la insuficiencia de datos con el aprendizaje acumulado en los pesos del modelo durante su entrenamiento y testeo en conjuntos masivos de imágenes para la detección de cientos de instancias de objetos. Adicionalmente, se trata de un modelo optimizado para el aprendizaje profundo con pocos requerimientos para su implementación.

Si bien los resulados obtenidos muestran baja capacidad del modelo para la detectar de forma precisa los fenómenos territoriales presentes en las imágenes, generan un impacto considerable en dos sentidos: el primero, como herramienta piloto de inteligencia artificial para apoyar tareas de planificación del territorio, muestra la pertinencia de los proyectos de ciencia de datos en la administración pública municipal orientados a extraer valor de una fuente de información no convencional como son las imágenes VHR. El segundo, como camino que integra el interés por el ordenamiento del territorio, el análisis de datos geoespaciales y el desarrollo de soluciones tecnológicas basadas en ciencia de datos, donde encuentro un campo profesional por construir.

7.2 Trabajos futuros

Los resultados de implementación del proyecto *Detección de Asentamientos informales usando imágenes VHR de Santiago de Cali* permite avizorar la oportunidad de mejora de la herramienta de detección en dos líneas de trabajo. La primera, que ya se ha mencionado, es ampliar el conjunto de imágenes y afinar la base de datos de fenómenos territoriales con mayor número de instancias de objetos; la segunda, es entrenar modelos más avanzados de la familia YOLO. Esas dos líneas de trabajo deben estar orientadas por el propósito de desplegar la herramienta de detección en un sistema de monitoreo de eventos territoriales en la Alcaldía de Santiago de Cali.

Llegar al despliegue de una solución tecnológica que permita realizar inferencias en imágenes satelitales de alta resolución espacial y permanente disponibilidad temporal es un horizonte posible para la administración municipal, si el piloto de la herramienta desarrollada muestra mejores niveles de precisión y confianza en la detección de los fenómenos territoriales de interés.

7.2.1 Ampliar conjunto de imágenes y afinar base de datos etiquetada para eventos territoriales

Aunque existen repositorios con aerofotografías e imágenes satelitales del territorio en distintas dependencias municipales, se requiere de herramientas se software geográfico y servidores

robustos para cargar y manipular estos archivos. Disponer de un conjunto de imágenes etiquetadas con fenómenos territoriales representa un ejercicio pionero en la Alcaldía de Cali.

Un trabajo futuro es ampliar este conjunto de imágenes existente a partir de la imagen Ortofotomosaico Cali (2020). Este trabajo implica procesar nuevamente los 239 segmentos de la imagen fuente, indicados en el punto 4.3.2, cada uno de dimensiones 6.400 píxeles X 6.400 píxeles, en formato GeoTiff. De este modo podría ampliarse el conjunto de imágenes a 2.151 de dimensiones 1.280 píxeles X 1.280 píxeles en formato PNG.

Una alternativa para aumentar aún más el conjunto de imágenes disponibles para entrenamiento de modelos de detección es segmentar las imágenes a dimensiones 640 píxeles X 640 píxeles, tamaño que también es admitido como parámetro de entrada para los modelos de la familia YOLO. De este modo, se multiplicaría por 4 el número del conjunto de imágenes.

A partir de un conjunto amplio de imágenes crudas, es posible movilizar a las dependencias con énfasis en el desarrollo territorial como son el Departamento Administrativo de Planeación, el Departamento Administrativo de Gestión del Medio Ambiente, Secretaría de Vivienda y Hábitat, y Secretaria de Gestión del Riesgo, Emergencia y Desastres, para que elaboren sus propias bases de datos con imágenes etiquetadas de acuerdo con los fenómenos territoriales de su interés.

Disponer de un conjunto de imágenes suficiente de Santiago de Cali y de bases de datos de imágenes con instancias de objetos etiquetados van a promover nuevos proyectos de ciencia de datos centrados en aprendizaje profundo que desarrollen modelos predictivos y prescriptivos para la planificación, monitoreo y control del territorio, y para el diseño de acciones de respuesta oportunas.

7.2.1 Probar detección con modelo YOLOv8

Los modelos YOLO son versátiles, eficientes, fáciles de usar, permiten personalizar sus hiperparámetros y ofrecen modelos pre entrenados. Bien sea que se disponga de un entorno computacional como el de AWS o de cualquier otro proveedor de servicios especializados para machine learning, YOLO ha configurado un ecosistema de herramientas de sotware libre, compatible con las grandes plataformas, que pueden ejecutarse desde la nube o con recusos on-premise.

YOLOv5 y YOLOv8 son modelos creados en 2020 y 2022 respectivamente por Ultralytics. Ambos modelos admiten imágenes de 640 y 1.280 píxeles y están disponibles en versiones nano, medium, large y extra large; igualmente son modelos rápidos para la detección de objetos en tiempo real, segmentación de instancias y clasificación de imágenes que admiten una gama amplia de formatos para imagen y video.

La prinicipal motivación para implementar YOLOv8 en trabajos futuros es porque la documentación existente apunta a mayores niveles precisión mAP en la detección de objetos y mayor eficiencia computacional como se muestra en la Figura 41 que compara las curvas de resultados para diferentes modelos de la familia YOLO.

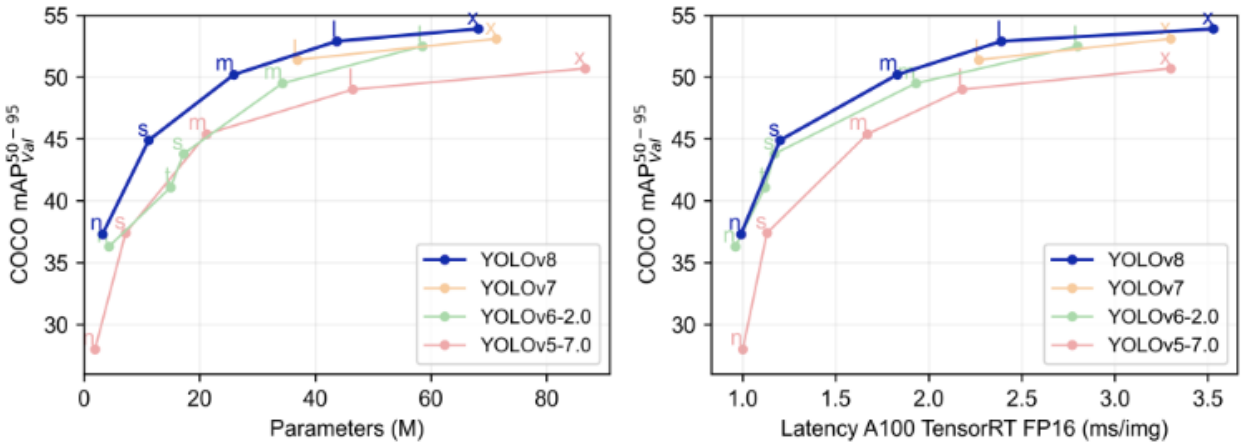


Figura 41. YOLOv8 comparado con otros modelos YOLO. Tomado de [36].

Se colige que la familia YOLO, especialmente YOLOv8, tiene algoritmos potentes para desarrollar detectores que aborden problemas de clasificación de fenómenos territoriales, que entrenados con bases de datos más extensas, tanto en número de imágenes como de instancias de objetos etiquetados, pueden mejorar los resultados obtenidos.

8. REFERENCIAS BIBLIOGRAFICAS

[1] BID, «MAIIA: Código para el desarrollo,» 2022. [En línea]. Available: https://code.iadb.org/es/herramientas/maiia . [Último acceso: 17 Septiembre 2022].

[2] Ministerio de Vivienda, «En los últimos 30 años ciudades y municipios de Colombia han crecido de manera informal,» Ministerio de Vivienda, 12 Diciembre 2018. [En línea]. Available: https://minvivienda.gov.co/sala-de-prensa/en-los-ultimos-30-anos-ciudades-y-municipios-de-colombia-han-crecido-de-manera-informal#:~:text=Los%20municipios%20que%20encabezan%20el,Soledad%20y%20Florencia%2C%20entre%20otras. [Último acceso: 17 noviembre 2022].

[3] G. d. C. -. DNP, «Bases del Plan Nacional de Desarrollo 2018-2022. Pacto por Colombia, pacto por la equidad.,» 2018. [En línea]. Available: https://ccong.org.co/files/867_at_BasesPND2018-2022.pdf. [Último acceso: 19 Septiembre 2022].

[4] BLUE radio, «Más de 1400 personas desalojan un lote invadido en el oriente de Cali.,» 6 octubre 2022. [En línea]. Available: https://www.bluradio.com/blu360/pacifico/mas-de-1-400-personas-desalojan-un-lote-invadido-en-el-oriente-de-cali-rg10. [Último acceso: 25 octubre 2022].

[5] Dirección de Censos y Demografía. DANE, «Predicción del IPM censal usando aprendizaje de máquinas e imágenes satelitales.,» DANE, Bogotá, 2020.

[6] K. Dubovikov, «What You Can Do with Data Science,» de *Managing Data Science: Effective Strategies to Manage Data Science Projects and Build a Sustainable Team.*, Packt Publishing, 2022.

[7] A. Geron, «The Machine Learning Landscape,» de *Hands-On Machine Learning with Cckit learn, Keras and TensosFow. Concepts, tools and techniques to Built Intelligent Systems*, O´Reilly, 2022, pp. 3-37.

[8] C. Santana, «DotCSV: ¿Qué es una red neuronal? Parte 1 y 2,» 19 marzo 2018. [En línea]. Available: https://www.youtube.com/watch?v=MRIv2IwFTPg. [Último acceso: 4 marzo 2023].

[9] S. Kostadinov, «Understanding Backpropagation Algorithm,» 2019. [En línea]. Available: https://towardsdatascience.com/understanding-backpropagation-algorithm-7bb3aa2f95fd . [Último acceso: 4 marzo 2023].

[10] C. Santana, «DotCSV: ¡Redes neuronales convolucionales! ¿Cómo funcionan?,» 12 noviembre 2020. [En línea]. Available: https://www.youtube.com/watch?v=V8j1oENVz00. [Último acceso: marzo 4 2023].

[11] E. R. Silva, «Tutorial: Entrenamiento de la Red Neuronal Convolucional YOLO para objetos propios,» Universidad Nacional Autónoma de México [Tesis de Maestría],

Ciudad de México, 2020.

[12] J. W. a. B. S. Y. Li, «Comparison of two target detection algorithms based on remote sensing images,» de *International Conference on Computer Information Science and Artificial Intelligence (CISAI)*, Kunming, China, 2021.

[13] L. M. J. W. a. H. x. W. Liu, «Detection of Multiclass Objects in Optical Remote Sensing Images,» *Geoscience and Remote Sensing Letters,* vol. 16, nº 5, pp. 791-795, 2019.

[14] R. Z. W. Z. S. S. y. N. W. J. Zhou, «APS-Net: An Adaptive Point Set Network for Optical Remote Sensing Object Detection,» *Geoscience and Remote Sensing Letters,* vol. 20, pp. 1-5, 2023.

[15] Y. L. y. L. He, «An Improved Object Detection CNN Module for Remote Sensing Images,» de *IEEE International Geoscience and Remote Sensing Symposium*, Kuala Lumpur, Malasia, 2022.

[16] X. Y. Y. H. Z. W. H. Y. G. C. Zhang, «Object class detection: A survey,» *ACM Computing Surveys,* vol. 46, nº 1, 2013.

[17] Surflaweb, «Qué es R-CNN, Fast R-CNN, Faster R-CNN y Mask R,» 6 febrero 2021. [En línea]. Available: https://www.youtube.com/watch?v=C-kBqPIZGo8. [Último acceso: 6 marzo 2023].

[18] S. H. K. G. R. Y. S. J. Ren, «Faster R-CNN: Towards Real Time Object Detection with Region Proposal Networks,» *IEEE Transactions on Pattern Analysis and Machine Intelligence. Institute of Electrical and Electronics Engineers,* vol. 39, nº 6, pp. 1137-1149, 2017.

[19] J. L. Y. H. K. Y. S. J. Dai, «R-FCN: Object detection via region-based fully convolutional networks,» *Advances in neural information processing systems,* pp. 379-387, 2016.

[20] H. R. G. D. Y. H. H. David Ameijeiras Sánchez, «Revisión de algoritmos de detección y seguimiento de objetos con redes profundas para videovigilancia inteligente,» *Revista Cubana d eCiencias Informáticas,* vol. 14, nº 3, 2020.

[21] N. Cassi Sertutxa, «Detección de componentes y nidos de pájaros en las líneas de transmisión aérea mediante técnicas de Deep-Learning,» E.T.S.I. Telecomunicación (UPM) [Tesis de Maestría], Madrid, 2023.

[22] S. Morales, «Analyzing Segregation of Informal Residents in Latin American Cities Periphery Using Remote Sensing,» *Revista Cartográfica,* nº 106, p. 77–97, 2023.

[23] C. P. J. R. B. a. A. S. N. Mboga, «Detection of informal settlements from VHR satellite images using convolutional neural networks,» de *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, Fort Worth, TX, USA, 2017.

[24] C. P. a. C. G. J. R. A. Bergado, «A deep learning approach to the classification of sub-decimeter resolution aerial images,» p. 1516–1519, 2016.

[25] F. L. W. H. J. Y. J. L. a. L. W. R. Fan, «Fine-Scale Urban Informal Settlements Mapping by Fusing Remote Sensing Images and Building Data via a Transformer-Based Multimodal Fusion Network,» *IEEE Transactions on Geoscience and Remote Sensing,* vol. 60, pp. 1-16, 2022.

[26] y. I. J. S. Rozada, «Estudio de la arquitectura YOLO para la detección de objetos mediante Deep Learning,» E.T.S.I. Telecomunicación (UV) [Tesis de maestría], Valladolid, 2021.

[27] F. F. A. C. E. Q. L. R. D. L. Urrea, «Una Mirada a los Asentamientos informales de Cali. Análisis de los datos Sisben III - 2019,» CIDSE - Universidad del Valle, Santiago de Cali, 2020.

[28] M. F. J. A.-Y. Y. N. H. Jani, «Model Compression Methods for YOLOv5: A Review,» 21 julio 2023. [En línea]. Available: https://arxiv.org/abs/2307.11904. [Último acceso: 8 noviembre 2023].

[29] G. Jocher, «YOLOv5 by Ultralytics (Version 7.0) [Computer software],» 22 noviembre 2022. [En línea]. Available: https://doi.org/10.5281/zenodo.3908559 . [Último acceso: 8 noviembre 2023].

[30] C. L. H. Y. I. W. Y. C. P. H. J. Wang, «CSPNet: A New Backbone that can Enhance Learning Capability of CNN,» 27 noviembre 2019. [En línea]. Available: arXiv:1911.11929. [Último acceso: 9 noviembre 2023].

[31] G. Jocher, « Comprender la implementación de SPP y SPPF # 8785,» Ultralytics, 29 julio 2022. [En línea]. Available: https://github.com/ultralytics/yolov5/issues/8785. [Último acceso: 14 noviembre 2023].

[32] Ultralytics, «Resúmen de Arquitectura. Técnicas de Aumento de datos. YOLOv5.,» Ultralytics, 12 noviembre 2023. [En línea]. Available: https://docs.ultralytics.com/yolov5/tutorials/architecture_description/#2-data-augmentation-techniques. [Último acceso: 20 noviembre 2023].

[33] Ultralytics, «Transferir aprendizaje con capas congeladas,» Ultralytics, 12 noviembre 2023. [En línea]. Available: https://docs.ultralytics.com/yolov5/tutorials/transfer_learning_with_frozen_layers/#__codelineno-3-13 . [Último acceso: 20 noviembre 2023].

[34] A. Geron, «Deep Computer Vision Using Convolutional Neural Networks,» de *Hands-On Machine Learning with Cckit learn, Keras and TensosFow. Concepts, tools and techniques to Built Intelligent Systems*, O´Reilly, 2022, pp. 479-535.

[35] E. D. C. a. B. Z. a. D. M. a. V. V. a. Q. V. Le, «AutoAugment: Learning Augmentation Policies from Data,» arXiv:1805.09501 , 2018.

[36] R. Soviético, «YOLOv8: Guía completa para la detección de objetos de última generación,» AprenderOpenCV, 10 enero 2023. [En línea]. Available: https://learnopencv.com/ultralytics-yolov8/. [Último acceso: 18 noviembre 2023].

[37] Ultralytics, «Evolución de Hiperparámetros. Definir Aptitud. YOLOv5,» Ultralytics, 12 noviembre 2023. [En línea]. Available: https://docs.ultralytics.com/yolov5/tutorials/hyperparameter_evolution/#before-you-start . [Último acceso: 20 noviembre 2023].