Trabajo final.

Aplicación de modelos de machine learning para la segmentación e identificación de árboles de cítricos.

[1]Felipe, Ceballos M.,[2]Brayan, C. Gaviria B.

*[1]*fceballosm@eafit.edu.co*, [2]* *bcgaviriab@eafit.edu.co*

Universidad EAFIT, Medellín
Antioquia, Colombia
noviembre de 2025

Introducción

En la agricultura moderna, el monitoreo preciso de los cultivos resulta esencial para optimizar la productividad y el manejo fitosanitario. En el caso de los cítricos, la identificación y segmentación de árboles individuales es una tarea que demanda alta inversión de tiempo y mano de obra especializada, un recurso cada vez más escaso en las zonas rurales.

Ante esta limitación, los modelos de aprendizaje automático y visión por computador se presentan como una alternativa eficiente para automatizar la recopilación y análisis de información agrícola. El uso de imágenes obtenidas mediante vehículos aéreos no tripulados (UAVs) o satélites, junto con modelos supervisados basados en redes neuronales convolucionales (CNN), permite realizar tareas de clasificación y segmentación con alta precisión, reduciendo la dependencia del trabajo manual.

Estudios recientes, como el de Tariku et al. (2024), demostraron que la combinación de técnicas avanzadas de preprocesamiento de imágenes (ESRGAN, CLAHE y balance de blancos) con arquitecturas VGG-16 y clasificadores SVM puede alcanzar precisiones superiores al 97% en la identificación de especies vegetales a partir de imágenes UAV. De manera complementaria, el plugin TreeEyed (Ruiz-Hurtado et al., 2025) integra modelos como DeepForest y Mask R-CNN dentro de QGIS, facilitando la detección y segmentación automática de árboles individuales.

La aplicación de estos enfoques en cultivos de cítricos permitiría mantener un control más constante del cultivo, optimizar los recursos disponibles y mitigar la escasez de mano de obra, posibilitando que uno o dos operarios, apoyados en inteligencia artificial, realicen tareas tradicionalmente efectuadas por equipos numerosos.

Objetivo general

Desarrollar y entrenar un modelo de aprendizaje profundo como DeepForest, para la identificación y segmentación automática de árboles de cítricos a partir de imágenes capturadas con vehículos aéreos no tripulados (UAVs), con el fin de generar una herramienta extrapolable al entorno QGIS que permita automatizar el proceso de detección de las copas de los árboles y facilitar el monitoreo espacial del cultivo.

Metodología

El proceso metodológico se centró en la generación de un conjunto de datos (dataset) destinado al entrenamiento de un modelo de detección de copas de árboles basado en DeepForest, utilizando imágenes aéreas RGB de plantaciones de cítricos obtenidas mediante drones. El flujo de trabajo incluyó las etapas de preprocesamiento de imágenes, alineación de datos vectoriales, generación de tiles, y creación de anotaciones en formato compatible con aprendizaje profundo.

1. Adquisición y preparación de datos
Se utilizaron ortomosaicos RGB en formato .tif, generados por fotogrametría a partir de vuelos UAV, y shapefiles (.shp) con polígonos de copas de árboles delineadas manualmente, que sirvieron como verdad terreno. Ambos conjuntos fueron reproyectados a un mismo sistema de referencia espacial (CRS) para garantizar la correcta superposición entre datos vectoriales y ráster.

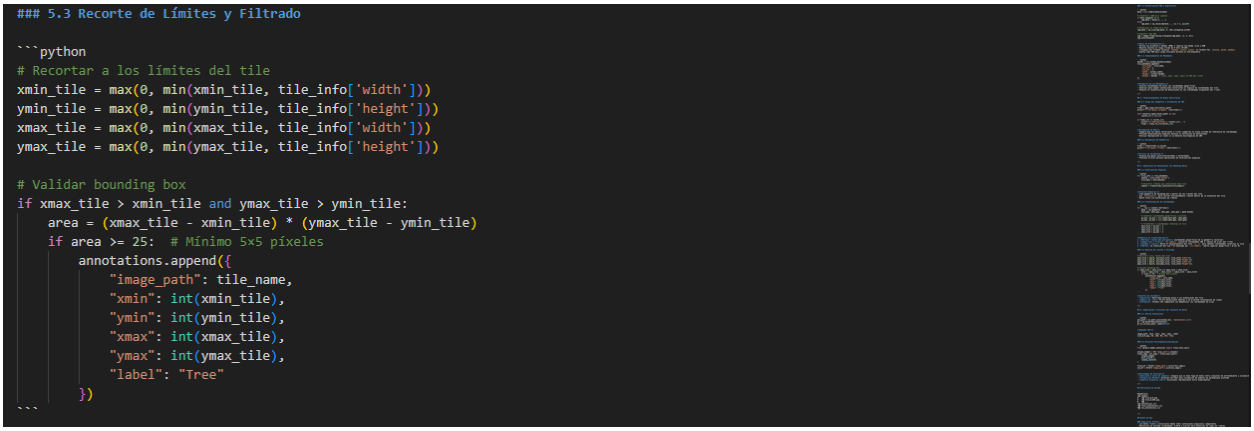
2. Generación de tiles con solapamiento controlado
Los ortomosaicos se segmentaron en subimágenes o tiles de 900×900 píxeles, empleando un solapamiento de 250 píxeles entre ventanas consecutivas. Esta estrategia permitió capturar completamente los árboles ubicados en los bordes de los tiles, evitando la fragmentación de copas. Cada tile se exportó como imagen PNG normalizada a 8 bits, conservando metadatos espaciales para su trazabilidad.

3. Procesamiento y filtrado de datos vectoriales
Las geometrías vectoriales fueron validadas para eliminar polígonos malformados y posteriormente intersectadas con los límites espaciales de cada tile. Para cada intersección, se calcularon las coordenadas de los límites (xmin, ymin, xmax, ymax) en unidades de píxeles relativas al tile, generando las anotaciones tipo bounding box necesarias para el entrenamiento supervisado.

4. Control de calidad y exportación del dataset
Se aplicó un umbral mínimo de 25 píxeles² para descartar anotaciones pequeñas o ruidosas. Los datos resultantes se consolidaron en un archivo CSV con el formato estándar de DeepForest (image_path, xmin, ymin, xmax, ymax, label). Posteriormente, se realizó una división estratificada del conjunto en 80% entrenamiento y 20% validación, asegurando que no existieran tiles compartidos entre ambos subconjuntos.

5. Estructura final del dataset

El conjunto de datos resultante incluyó una carpeta images/ con los tiles ráster y tres archivos: annotations.csv, train_annotations.csv y val_annotations.csv. Esta estructura garantiza compatibilidad directa con los métodos de entrenamiento de DeepForest y permite la futura extrapolación del modelo entrenado a entornos geoespaciales como QGIS, para automatizar la segmentación de árboles a partir de imágenes UAV.



````
### 5.3 Recorte de Límites y Filtrado

```python
# Recortar a los límites del tile
xmin_tile = max(0, min(xmin_tile, tile_info['width']))
ymin_tile = max(0, min(ymin_tile, tile_info['height']))
xmax_tile = max(0, min(xmax_tile, tile_info['width']))
ymax_tile = max(0, min(ymax_tile, tile_info['height']))

# Validar bounding box
if xmax_tile > xmin_tile and ymax_tile > ymin_tile:
    area = (xmax_tile - xmin_tile) * (ymax_tile - ymin_tile)
    if area >= 25:  # Mínimo 5x5 píxeles
        annotations.append({
            "image_path": tile_name,
            "xmin": int(xmin_tile),
            "ymin": int(ymin_tile),
            "xmax": int(xmax_tile),
            "ymax": int(ymax_tile),
            "label": "Tree"
        })
```
````

Figura 1: Código donde se muestra las especificaciones para el recorte de limites y filtrado (autoría propia).

El modelo se entrenó utilizando el dataset generado a partir del pipeline de preprocesamiento, compuesto por tiles de imágenes UAV y sus respectivas anotaciones en formato de bounding boxes. Se empleó la librería DeepForest, implementada sobre PyTorch, configurada para tareas de detección de objetos en imágenes RGB.

El entrenamiento se realizó sobre una división 80/20 de los datos (entrenamiento/validación), utilizando transfer learning a partir de pesos preentrenados en bosques templados. Se ajustaron parámetros como el número de épocas, la tasa de aprendizaje y el tamaño de lote, optimizando el desempeño mediante evaluación continua del F1-score, precisión y recall.

Durante la validación, el modelo fue capaz de detectar y segmentar copas de árboles de cítricos con alta correspondencia espacial respecto a las anotaciones originales, evidenciando su capacidad de generalización sobre nuevas imágenes UAV. Finalmente, el modelo entrenado fue exportado en formato .pth, permitiendo su futura integración en entornos geoespaciales como QGIS mediante un plugin similar a TreeEyed, para automatizar la identificación y segmentación de árboles directamente desde imágenes aéreas.

Resultados

Como resultado del proceso de preprocesamiento descrito, se obtuvo un dataset estructurado y listo para el entrenamiento del modelo, compuesto por mosaicos ráster divididos en tiles de 900×900 píxeles y sus respectivas anotaciones en formato de bounding boxes. Este conjunto permitió representar de manera precisa la distribución espacial de las copas de los árboles de cítricos dentro del área de estudio.

En la fase inicial de revisión visual del dataset, se observó que cada imagen contenía en promedio entre 5 y 7 árboles individuales, claramente delimitados a partir de las anotaciones vectoriales originales. Durante las primeras pruebas de inferencia con el modelo entrenado, se evidenció que el modelo lograba detectar entre 4 y 5 copas por imagen, mostrando una correspondencia visual alta con las anotaciones de referencia y confirmando la capacidad del modelo para identificar la mayoría de los individuos presentes.

Estos resultados preliminares indican que el proceso de generación del dataset fue adecuado para capturar la variabilidad del dosel en las plantaciones de cítricos, y que el modelo comenzó a generalizar correctamente las características morfológicas de las copas, sentando las bases para etapas posteriores de ajuste fino y validación cuantitativa.

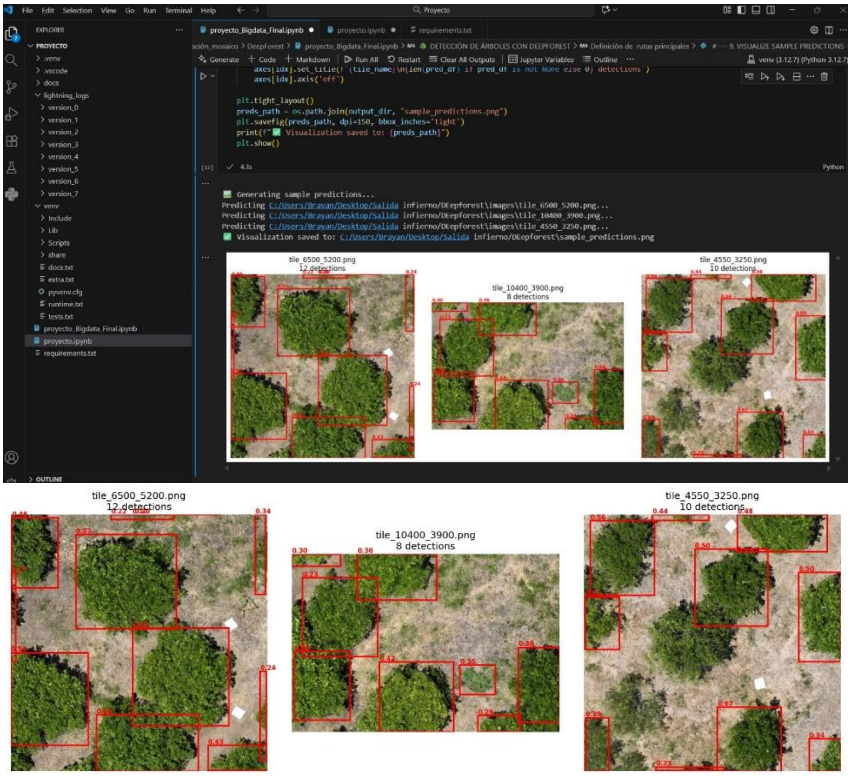


Figura 2: Muestra de como el modelo realiza el reconocimiento de los arboles y la cantidad que identifica por imagen (autoría propia).

Durante la fase de entrenamiento, se configuró el modelo DeepForest para ejecutarse con cinco épocas de ajuste utilizando los conjuntos de datos generados en la etapa de preprocesamiento. Sin embargo, se observó que la primera época proporcionaba el mejor desempeño general del modelo, alcanzando valores de precisión y recall más estables en comparación con las iteraciones posteriores.

Análisis de resultados

Al evaluar el desempeño del modelo, se realizó tanto un análisis visual como estadístico. A partir de la inspección directa de las imágenes procesadas, se evidencia que el modelo logra identificar entre el 80% y el 90% de los árboles presentes en las escenas. Esto significa que, de manera práctica, la detección de copas es consistente y funcional, dado que la mayoría de los árboles visibles en la imagen son correctamente marcados mediante bounding boxes. Esta observación respalda que, en términos de percepción visual, el modelo alcanza un rendimiento adecuado en la tarea de detección.

Sin embargo, cuando se revisan los resultados desde una perspectiva estadística, se observa una situación diferente. El valor promedio de confianza del modelo fue de 38%, lo cual indica que, aunque el modelo logra identificar gran parte de los árboles, no lo hace con un nivel de seguridad elevado. Es decir, el modelo "detecta" los árboles, pero no está completamente seguro de que sus predicciones sean correctas. Esto sugiere que el modelo aún necesita fortalecerse, ya sea mediante un ajuste de hiperparámetros, la inclusión de mayor cantidad de datos de entrenamiento o la incorporación de datos locales más representativos.

Conclusiones

En conclusión, la implementación de modelos de Machine Learning como DeepForest ha demostrado ser útil para la segmentación y detección de árboles en imágenes obtenidas por UAV, permitiendo ubicar los árboles y reconocerlos dentro del dataset. Sin embargo, es importante señalar que el rendimiento del modelo puede mejorarse significativamente. Para ello, es recomendable aumentar el tamaño del dataset, incorporando más imágenes que reflejen la diversidad de los árboles en cuanto a tamaño de copa, forma y densidad foliar. Esto permitiría al modelo detectar incluso árboles con condiciones particulares, como aquellos con copas muy despejadas, que en algunos casos se confundieron con el suelo.

Asimismo, aunque el modelo ha mostrado una precisión considerable, es necesario realizar pruebas adicionales con datasets distintos y más extensos, así como optimizar el entrenamiento del modelo, para garantizar su robustez y capacidad de generalización en distintos escenarios y condiciones de vuelo.

Referencias

Ruiz-Hurtado, A. F., Pérez Bolaños, J., Arrechea-Castillo, D. A., & Cardoso, J. A. (2025). *TreeEyed: A QGIS plugin for tree monitoring in silvopastoral systems using state of the art AI models*. SoftwareX, 29, 102071. **https://doi.org/10.1016/j.softx.2025.102071**

Tariku, G., Ghiglieno, I., Simonetto, A., Gentilin, F., Armiraglio, S., Gilioli, G., & Serina, I. (2024). *Advanced image preprocessing and integrated modeling for UAV plant image*

classification. Drones, 8(11), 645. **https://doi.org/10.3390/drones8110645**

Saurav. (2024, diciembre 25). *Drone detection using deep learning: A VGG16-based approach for object localization*. Medium. Recuperado de **https://medium.com/@1032211306/drone-detection-using-deep-learning-a-vgg16-based-approach-for-object-localization-056b743af44a**