

Caso 2: Detección de piscinas y análisis de tamaño en imágenes satelitales

1. Propuesta de solución basada en IA

El objetivo es de la propuesta desarrollar un sistema automatizado basado en distintas tecnologías, capaz de detectar de forma precisa la existencia o ausencia de casas en una dirección determinada y, en los resultados positivos, poder garantizar la presencia o no de piscina, así como devolver el tamaño de la misma. Además, se busca minimizar los falsos positivos causados por la posibilidad de detección de lagos y ríos. El flujo de trabajo incluye:

1. Detección de casas en la dirección indicada mediante el modelo **YOLOv8**.
2. En las imágenes con casas detectadas, se procede a la segmentación de todas las regiones de la imagen mediante **Segment Anything Model (SAM)**.
3. Filtrado de únicamente regiones que podrían ser piscinas mediante un análisis de color y forma con **OpenCV**.
4. Medición del tamaño de las piscinas (**SAM**).
5. Comprobación de que la piscina detectada es real y no un lago o río mediante el uso de **OpenStreetMap API**.
6. Creación de una función final que devuelva un JSON estructurado con la información relevante.

2. Técnicas/modelos a implementar y justificación

- **YOLOv8**: Se trata de una CNN para detección de objetos de forma eficiente y rápida, la cual funciona bien en imágenes satelitales sin necesidad de entrenamiento adicional. Podría obtener buenos resultados detectando casas, ya que estas suelen ser estructuras muy definidas con bordes marcados, lo que facilita su detección.

Enlace 1: [Documentación YOLOv8](#)

- **Segment Anything Model (SAM) de Meta AI**: Al igual que la tecnología anterior, SAM no necesita entrenamiento previo. Segmenta automáticamente cualquier objeto y permite extraer la "máscara" exacta de la piscina, lo que facilita su posterior medición utilizando esta misma tecnología.

Enlace 2: [Documentación SAM](#)

- **OpenCV**: Con OpenCV podemos detectar piscinas basándonos en el color (azul, turquesa) y la geometría de la misma (rectangular o circular), lo que ayuda al reconocimiento de la piscina, pero también a descartar posibles masas de agua naturales (formas irregulares).

Enlace 3: [Documentación OpenCV](#)

- **OpenStreetMap API:** llamadas a una API de datos geospaciales con el objetivo de detectar si las coordenadas que hemos señalado como “piscina” corresponden a zona de lago o río registrada para su descarte. Esta propuesta funciona como segundo filtro para poder hacer una mejor diferenciación entre piscina y zona de agua natural.

3. Modificación de parámetros y posibilidad de fine-tuning

En principio no se considera necesario realizar fine-tuning en esta propuesta, ya que todos los modelos seleccionados son preentrenados y suelen generalizar bien para este tipo de problemas. Si los resultados no fueran los esperados por ciertas situaciones (casas con techos poco definidos, piscinas con formas irregulares, cubiertas o aguas no limpias), podríamos considerar ciertas modificaciones antes de proceder a realizar entrenamientos:

- Ajustar parámetros de confianza en YOLOv8.
- Probar distintos umbrales en SAM.
- Refinar los filtros de color y forma en OpenCV.
- Ajustar el radio de búsqueda para aumentar la precisión en OpenStreetAPI.

Si probadas varias modificaciones se siguieran sin obtener buenos resultados, se podría considerar fine-tuning para YOLOv8 con imágenes específicas. El proceso incluiría los siguientes pasos:

- Reunir un dataset de imágenes satelitales bien etiquetadas en formato YOLO (bounding boxes).
- Ajustar un 80/20 de los datos para train y test, organizando el dataset en carpetas (images, labels).
- Entrenar del modelo (carga y configuración de hiperparámetros).
- Evaluar el rendimiento del modelo.

4. Métricas

En cada parte del proceso será necesario aplicar unas métricas concretas para saber si en cada momento se está ejecutando cada acción de forma exitosa:

- Evaluación en la detección de casas: Accuracy, Recall, F1-Score.
- Evaluación de la Segmentación con SAM: IoU, Dice Coefficient y MAE.
- Evaluación del filtro de color y forma (OpenCV): Accuracy, Tasa de Falsos Positivos.
- Evaluación del filtro de OpenStreetMap API: Accuracy en descarte de lagos/ríos.

5. Justificación del posible éxito

El enfoque propuesto tiene altas probabilidades de obtener buenos resultados gracias al uso de modelos preentrenados como YOLOv8 y SAM, que permiten detectar casas y segmentar imágenes sin necesidad de entrenar modelos desde cero.

Además, la combinación de distintas técnicas optimiza el flujo de trabajo: YOLOv8 se encarga de identificar las casas, SAM segmenta las piscinas, OpenCV ayuda a filtrar falsos positivos y OpenStreetMap permite descartar lagos y ríos.

Finalmente, al ser una solución basada en modelos flexibles y datos geoespaciales, es escalable y adaptable a diferentes ubicaciones y condiciones sin necesidad de grandes ajustes, lo que la hace viable para su aplicación en escenarios reales.