

Carrera de Especialización en Inteligencia Artificial

Proyecto:

Aimbot para videojuegos basado en YoloV5s y desplegado en GCP

Responsable:

Carlos Jorge Pallares Urieles

Contenido:

- Descripción del problema
- Análisis exploratorio de conjunto de datos
- Tarea de visión a resolver
- Experimentos a realizar
- Resultados y conclusiones

Descripción del problema



Asistencia de direccionamiento con previo reconocimiento: al tener que realizar acciones inmediatas con dispositivos mecánicos, el tiempo de reacción de una persona para llegar a un accionamiento puede ser lento. Ejemplo:

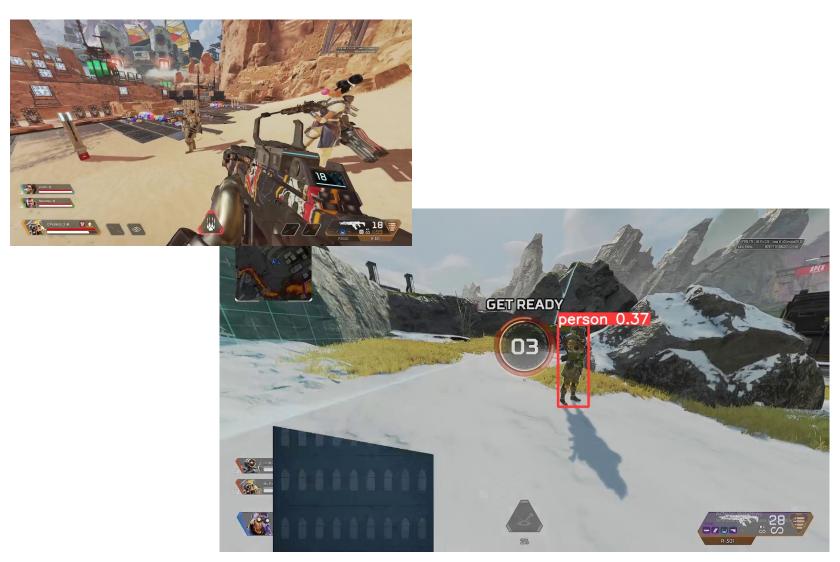
Supongamos la evaluación de pastillas de uranio para un reactor, en muchos casos como en Conuar, el retiro de estas pastillas es totalmente manual, y se quiere automatizar dicho proceso.

Todo dependerá del cuadro a cuadro que registre una cámara actualmente.

Análisis exploratorio



Especialización en Inteligencia Artificial



Análisis exploratorio



Especialización en Inteligencia Artificial

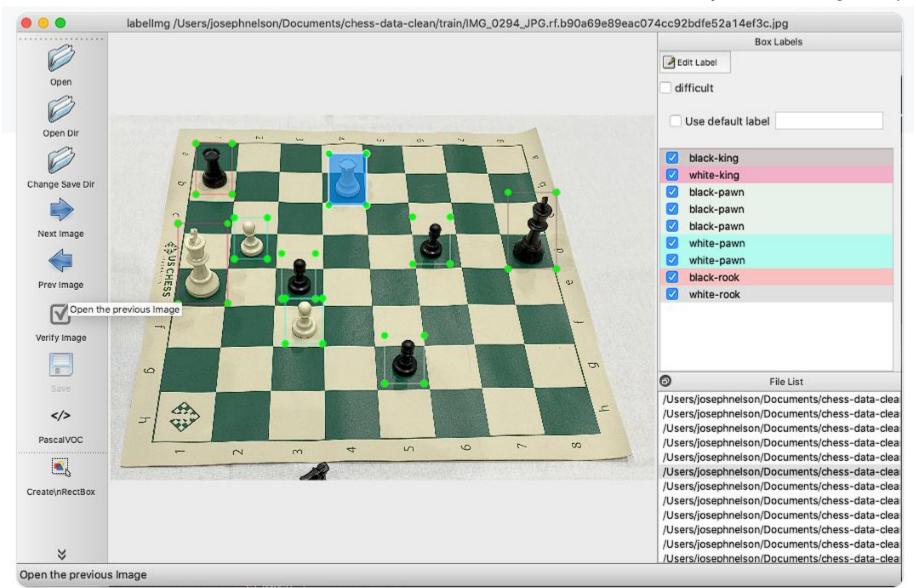
Auto-Orient	Edit
Isolate Objects	Edit
Resize	Edit
Stretch to 640×480	
Auto-Adjust Contrast	Edit
Using Histogram Equalization	
Modify Classes	Edit
0 remapped, 4 dropped	

Crop 0% Minimum Zoom, 20% Maximum Zoom	Edit
Hue	Edit
Between -15° and +15°	
Blur	Edit
Up to 2px	Luit
Bounding Box: Brightness	Edit
Between -32% and +32%	
Bounding Box: Blur	Edit
Up to 5.75px	Luit
Bounding Box: Noise	Edit
Up to 5% of pixels	Euli

Análisis exploratorio



Especialización en Inteligencia Artificial



Tarea de visión a resolver



Sistema de tracking de enemigos en tiempo real a través de captura de pantalla desplegado en Cloud. Las herramientas utilizadas fueron:

- Transfer learning, YoloV5s
- Roboflow
- MSS (Multiple Screen Shot)
- PyAutoGUI
- Docker
- Google Cloud

Tarea de visión a resolver



Dataset:

- 169 fotos de entrenamiento
- 103 fotos de validación
- Tamaño fijo de 1280x720 píxeles

Parámetros de YoloV5s:

- Imágenes de 640x480 píxeles
- 4 de batch
- 10 epochs
- PyAutoGUI

Resultados y conclusiones



Rendimiento:

- Resultado final de MAP50 = 0.915.
- Utilizar epochs mayores a 20 con transfer learning para 1 clase y pocas fotos incurre en sobre ajuste del modelo.
- Tiempo de entrenamiento, 38 segundos por epochs, un total de 6 minutos y 20 segundos.

Parámetros de YoloV5s:

- Imágenes de 640x480 píxeles
- 4 de batch
- 10 epochs
- Entrenado en Google Colab

Resultados y conclusiones



Latencia:

- 10 segundos por cuadro al usar GCP, problema en carga de modelo.
- 1 segundo por cuadro al usar servidor local.

Costos GCP:

 Importante no exceder el número de instancias al usar Google Run.