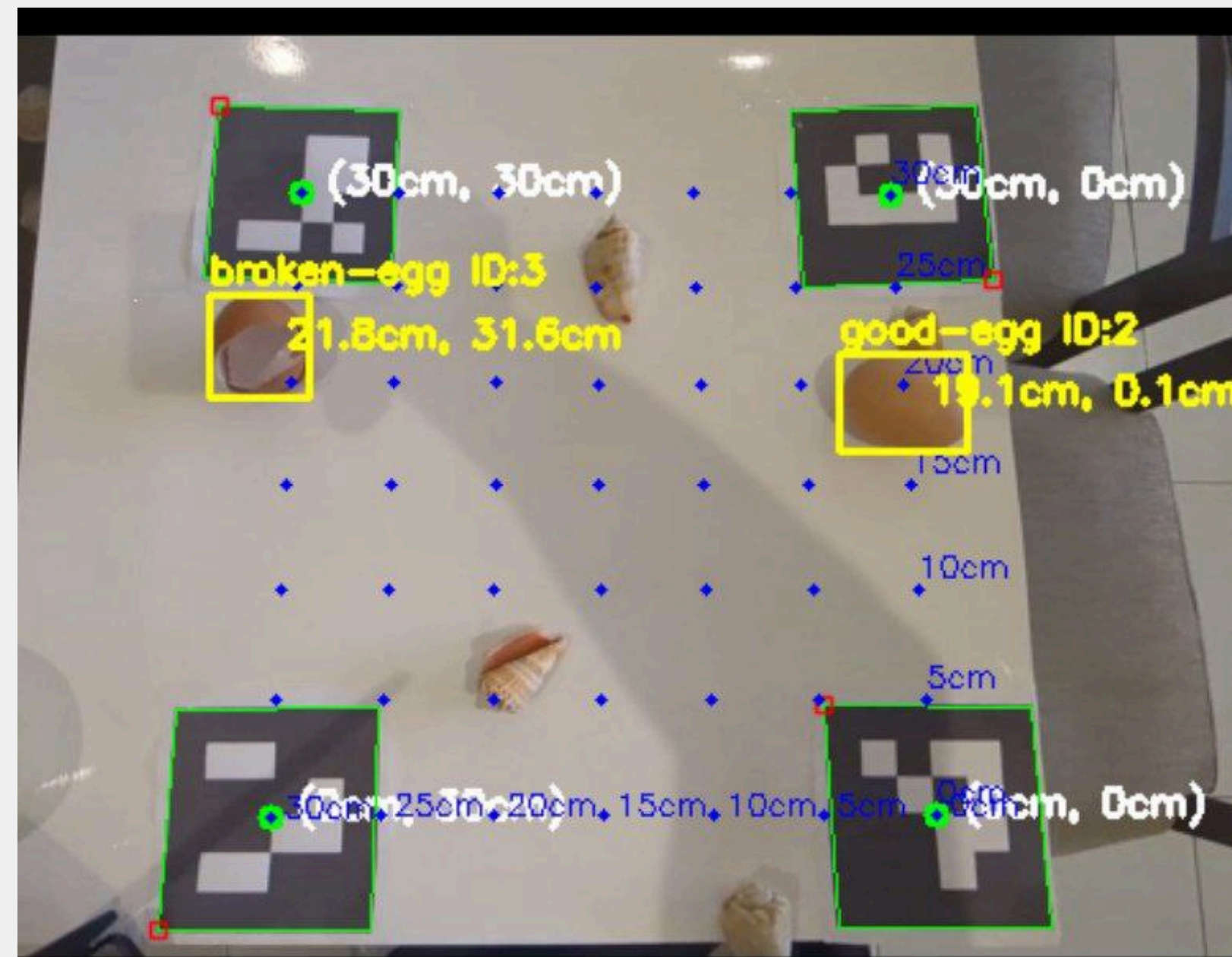


“Egg Detection” usando YOLOv8

Lucas Pin
Carlos Benitez



“Egg Detection” usando
YOLOv8

El proyecto

inteligencia Artificial

Este proyecto constituye la vision de un sistema robótico orientado a retirar huevos defectuosos en una línea de empaquetado. El objetivo principal es detectar y clasificar huevos en dos categorías: broken egg y good egg, además de obtener las coordenadas para ubicarlos en el espacio de trabajo. En las pruebas preliminares usamos YOLOv8 Nano, con datos clasificados mediante Roboflow. Para mejorar la posición, complementamos con marcadores Aruco que sirven como referencia para escalar la ubicación de los huevos.

El proyecto también incorpora la optimización de parámetros con Optuna y un sistema que organiza automáticamente el dataset con marcas de tiempo, facilitando el análisis y evaluación.



O P T U N A



“Egg Detection” usando
Yolov8

Flujo de Trabajo del Proyecto

inteligencia Artificial

**1-) Obtención
del Dataset.**

**2-) Elección
del modelo
base.**

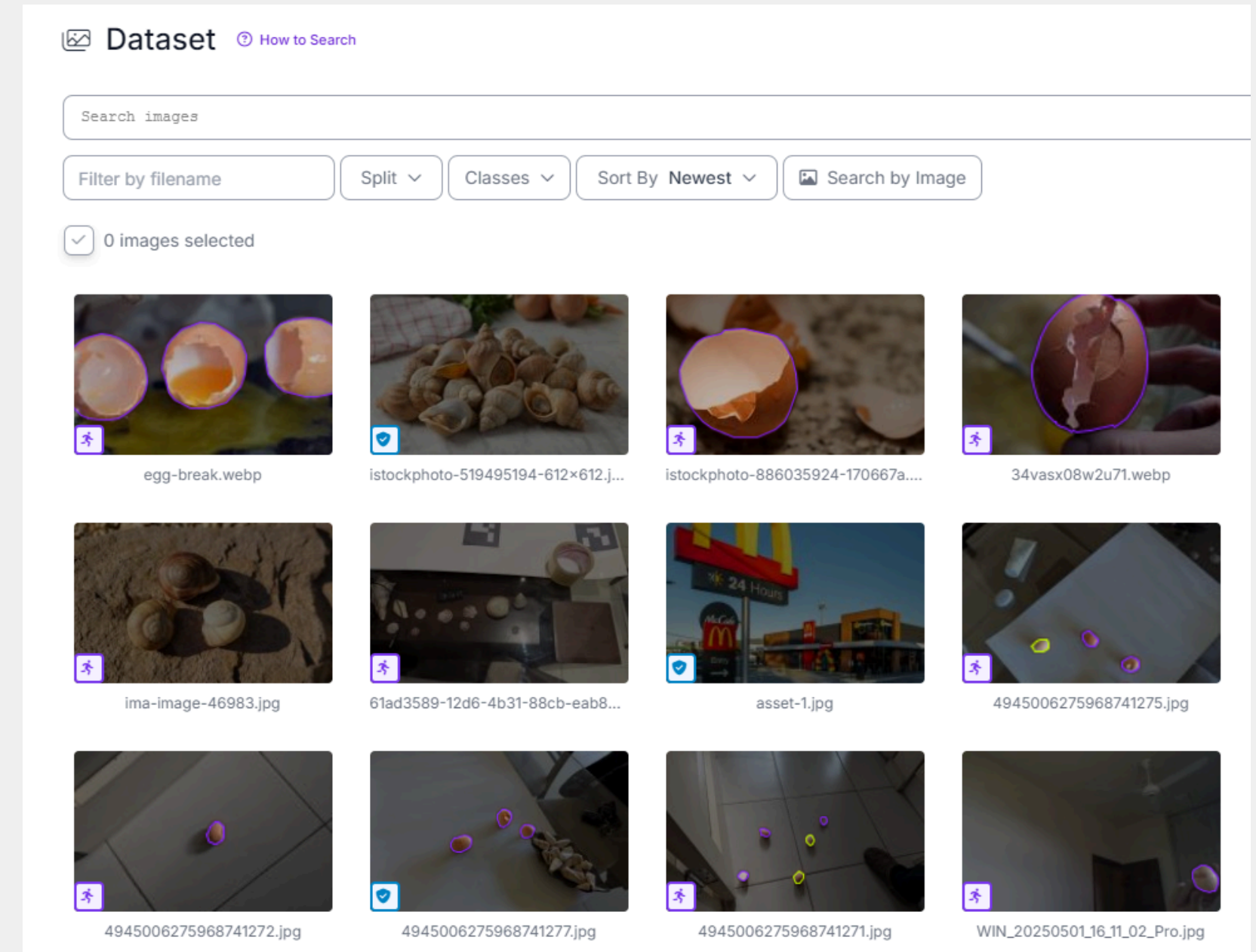
**3-) Ajuste de
Hyperparamet
ros.**

**4-) Implementación
(Inferencia
en tiempo
real)**

“Egg Detection” usando
Yolov8

Obtención del Dataset

Para obtener el Dataset, se utilizó la plataforma Roboflow, este cuenta con su propia comunidad donde se pueden encontrar gran variedad de imágenes de donde partir. Aun así se requirió de una limpieza y etiquetado manual del dataset obtenido, eliminando imágenes mal etiquetadas y subiendo nuevas imágenes propias en entornos diferentes que ayuden al entrenamiento.



“Egg Detection” usando
Yolov8

Obtención del Dataset



Estadísticas del Dataset:

- Total de imágenes: 1376
- Train : 1140 imágenes (82.85%)
- Valid : 119 imágenes (8.65%)
- Test : 117 imágenes (8.50%)



Clases encontradas:

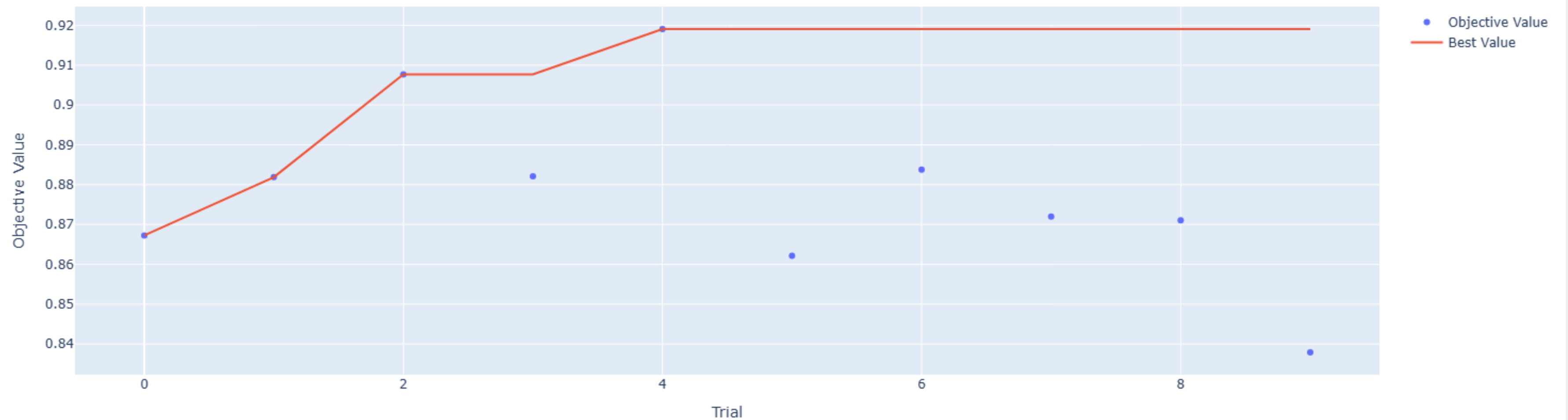
- Clase 0: 1181 imágenes
- Clase 1: 2453 imágenes

Datos del Dataset:

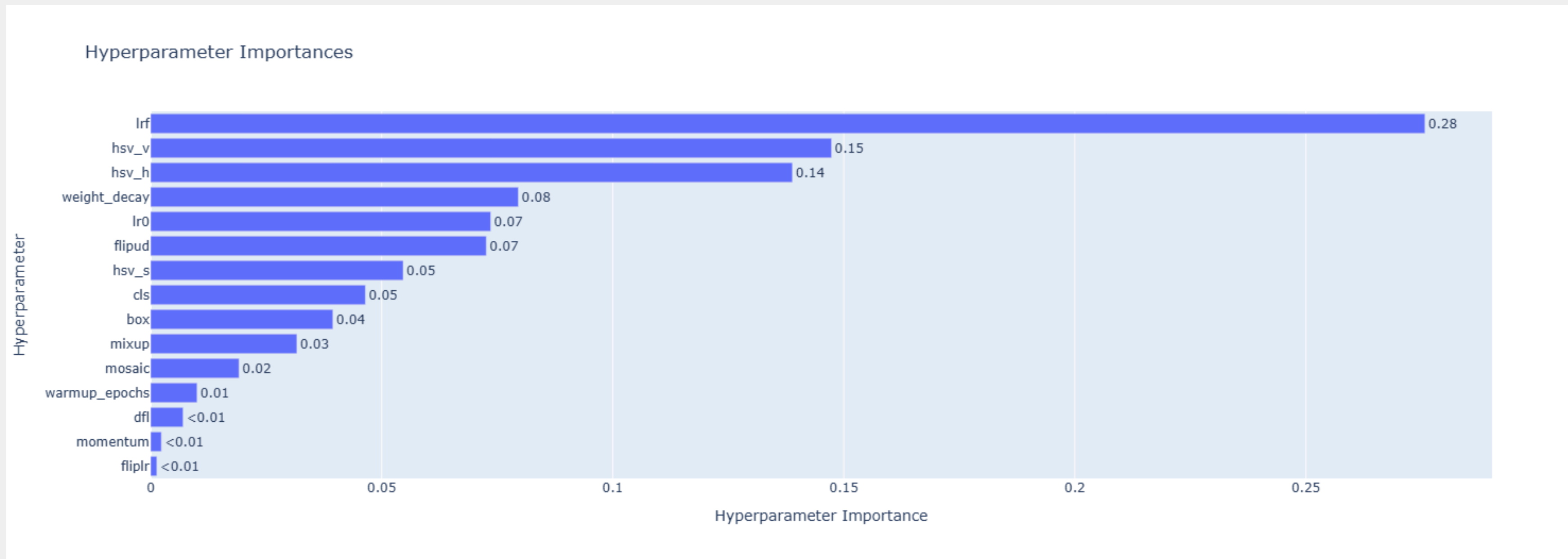
- 616 Imagenes subidas.
- Data augmentation utilizados: flip vertical y horizontal, rotaciones, ruido y blur, dejando un total de 1376 imágenes en total.
- Distribución: Train set: 82.65%, Valid set: 8.65%, Test set: 8.5%.

Ajuste de los Hyperparametros

Optimization History Plot

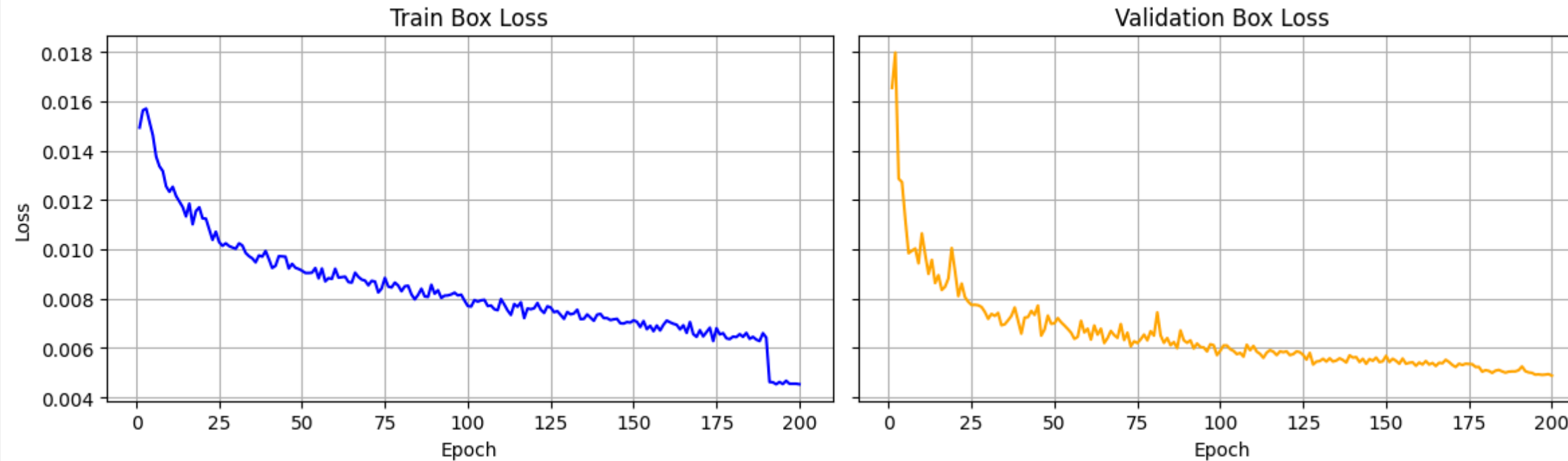


Ajuste de los Hyperparametros

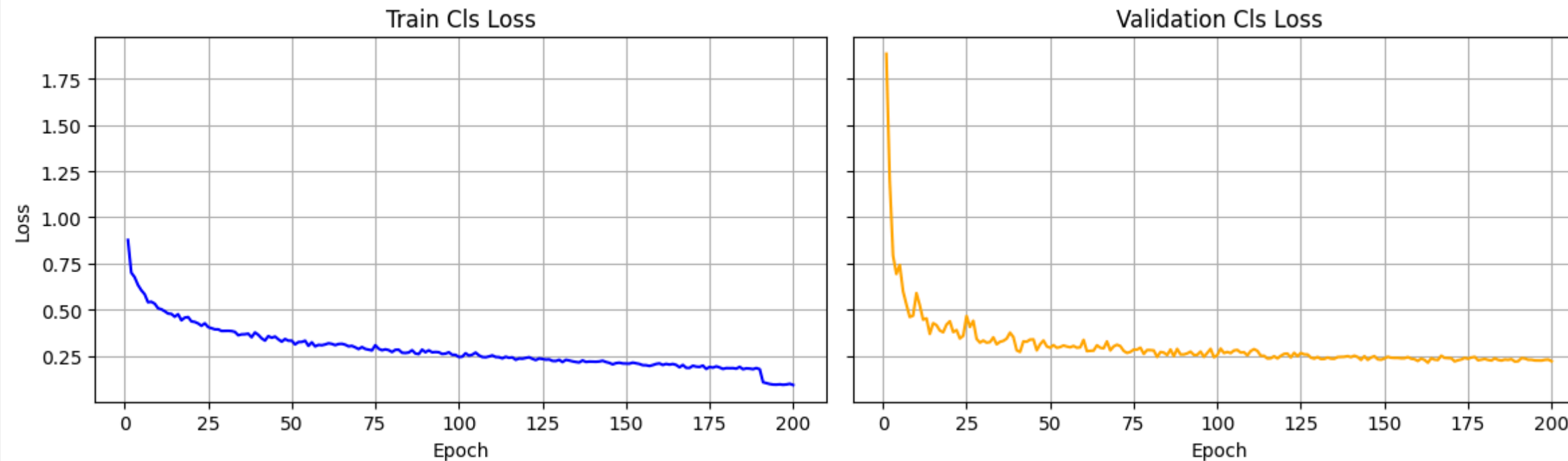


Curvas de Aprendizaje Obtenidas

Box Loss Comparison



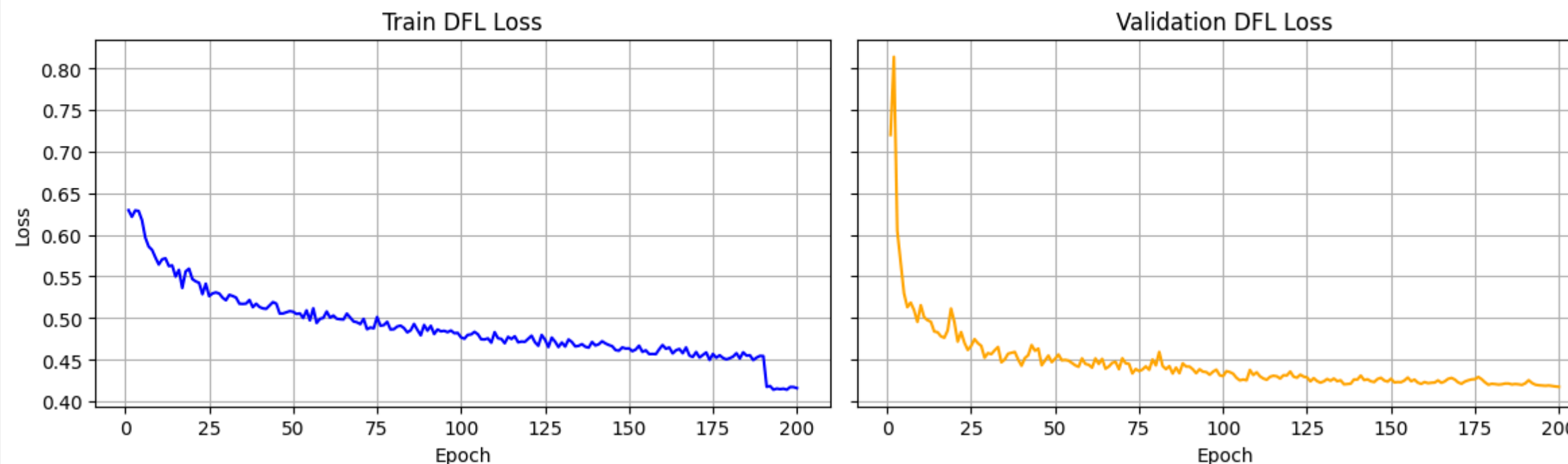
Cls Loss Comparison



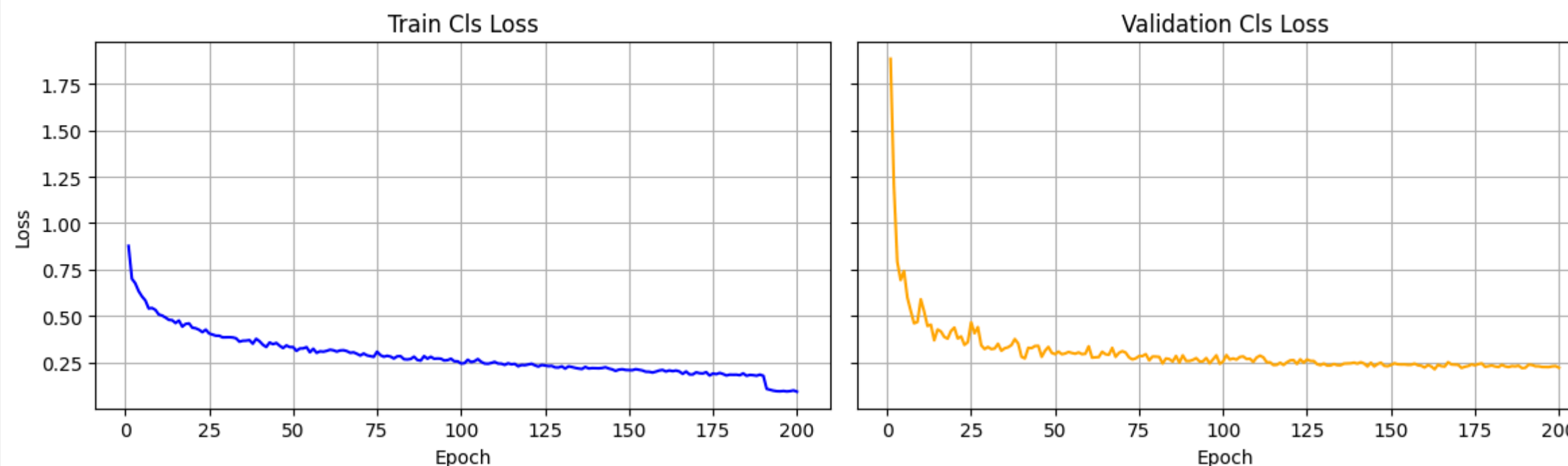
“Egg Detection” usando
Yolov8

Curvas de Aprendizaje Obtenidas

DFL Loss Comparison

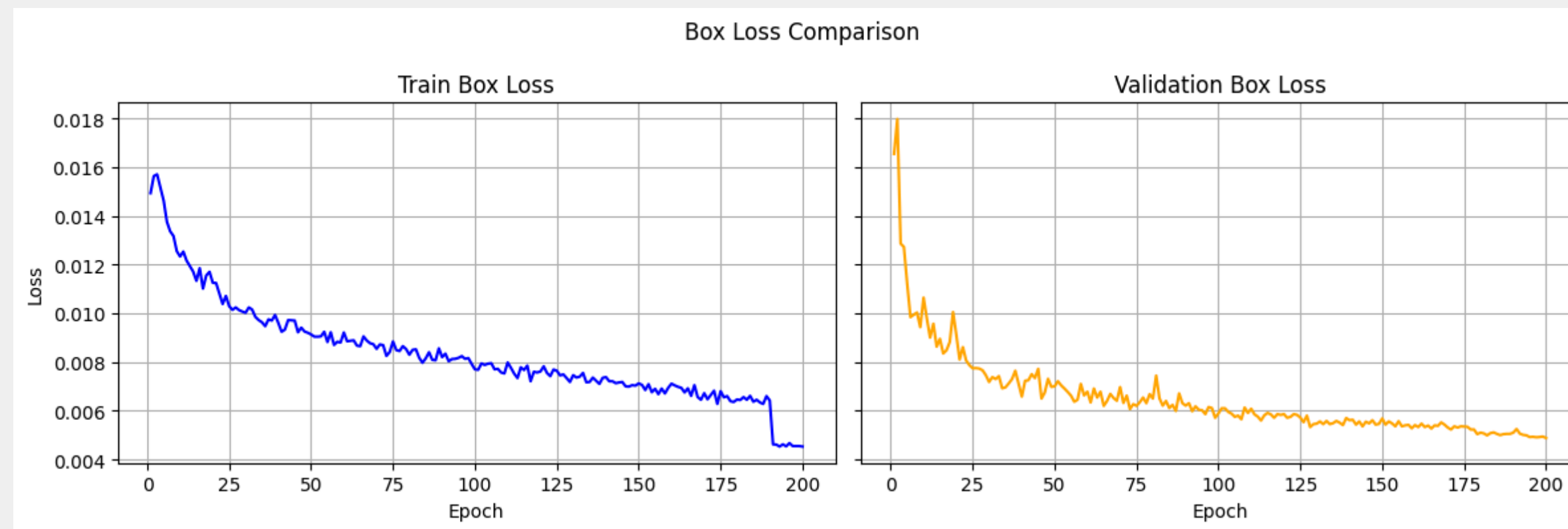


Cls Loss Comparison

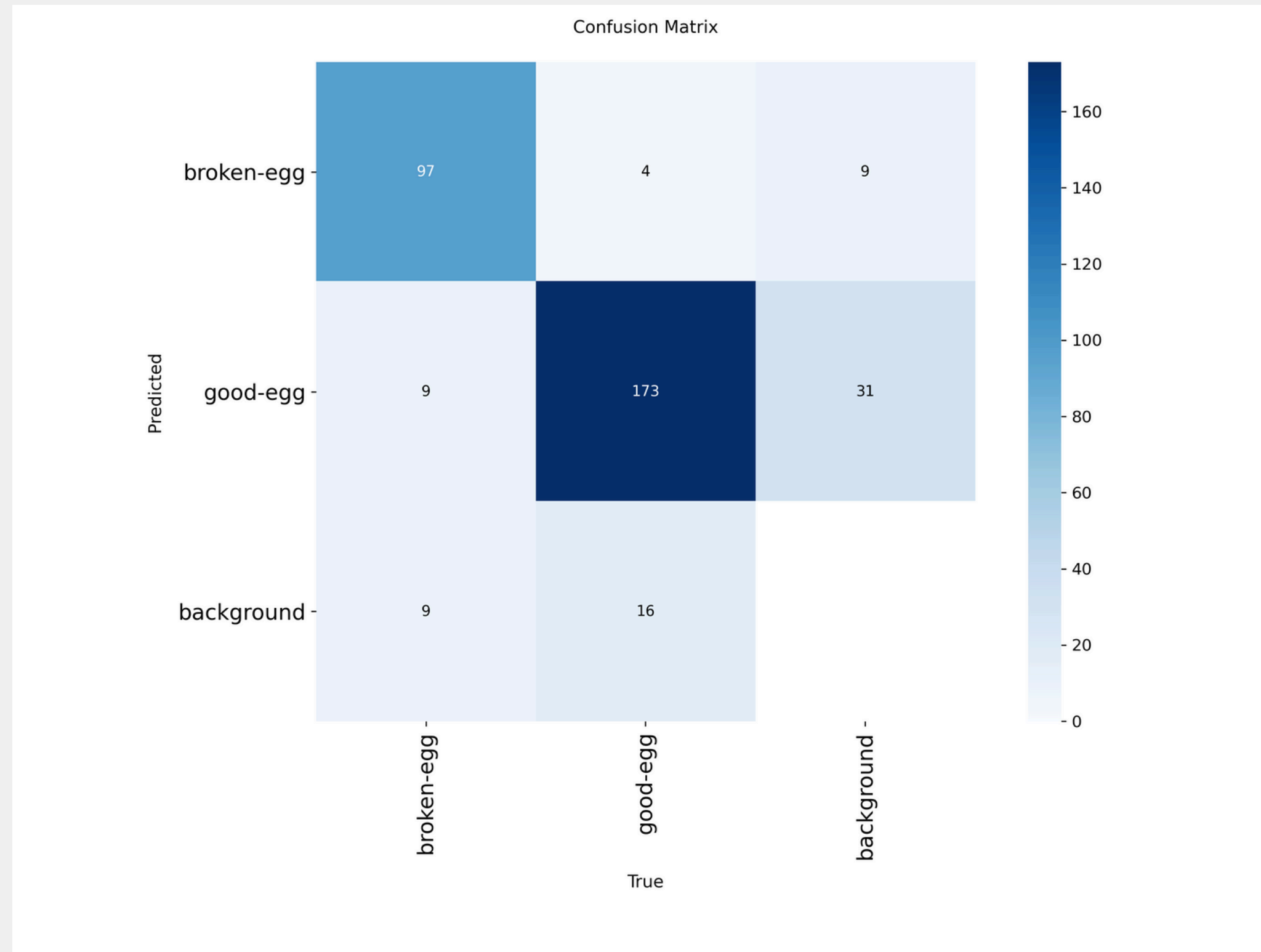


“Egg Detection” usando
Yolov8

Curvas de Aprendizaje Obtenidas



Matriz de Confusión Normalizada

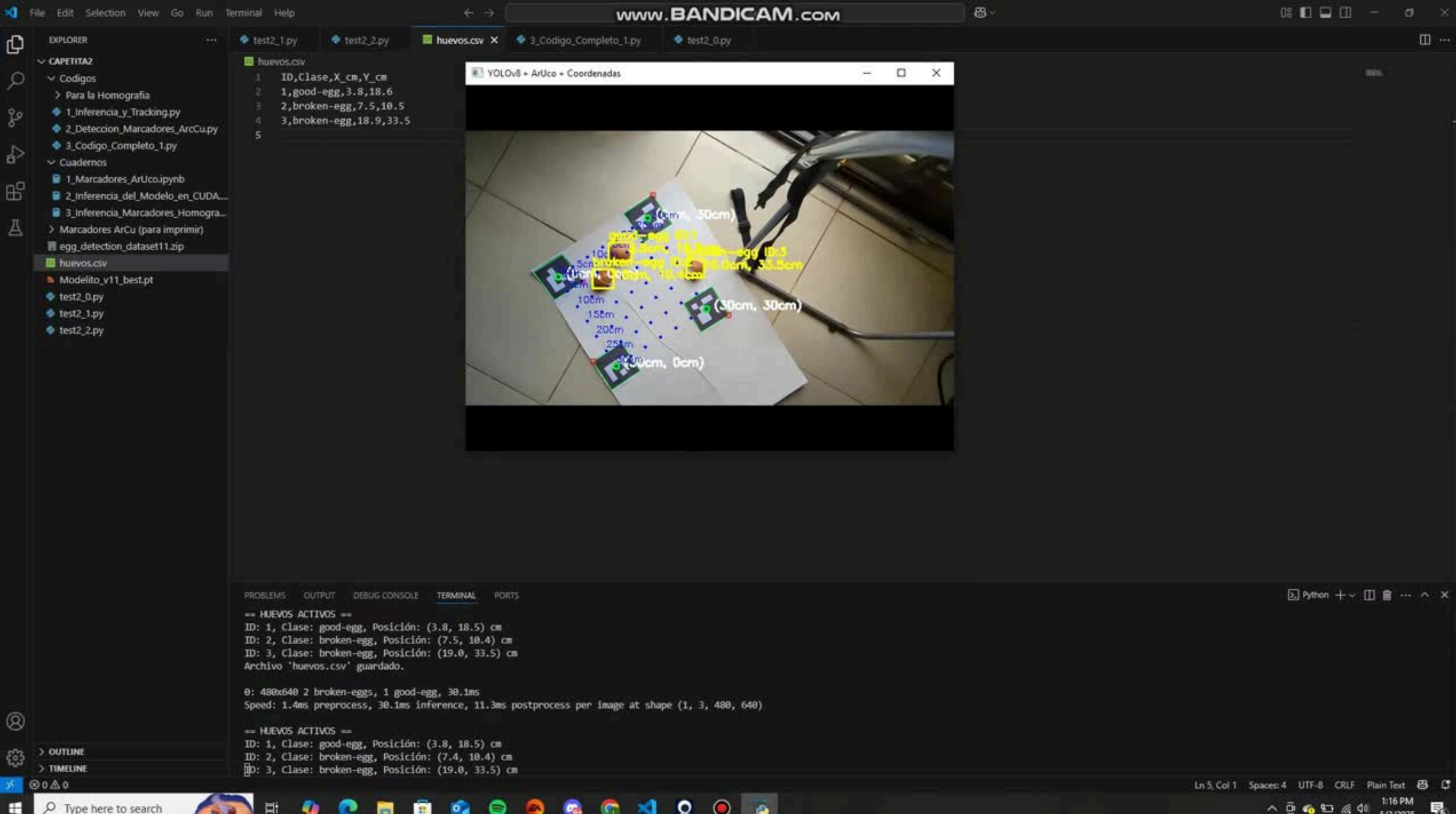


"Egg Detection" usando
Yolov8

Metricas de Error Obtenidas

Class	Images	Instances	Box(P)	Box(R)	Box(mAP50)	Box(mAP50-95)
all	119	308	0.875	0.874	0.92	0.878
broken-egg	72	115	0.915	0.846	0.927	0.88
good-egg	44	193	0.835	0.835	0.914	0.875

Inferencia en Tiempo Real y Tracking



The screenshot shows a VS Code IDE with a project named 'CAPETITA2'. The file explorer on the left lists several files, including 'huevos.csv' and 'Modelito_v11_best.pt'. The main editor window displays the 'huevos.csv' file, which contains the following data:

ID	Clase	X_cm	Y_cm
1	good-egg	3.8	18.6
2	broken-egg	7.5	10.5
3	broken-egg	18.9	33.5

The video feed in the center shows three eggs on a table. The first egg is labeled 'good-egg (ID: 1)' with coordinates (3.8cm, 18.6cm). The second egg is labeled 'broken-egg (ID: 2)' with coordinates (7.5cm, 10.5cm). The third egg is labeled 'broken-egg (ID: 3)' with coordinates (18.9cm, 33.5cm). The terminal window at the bottom shows the output of the program, including the detected objects and their positions:

```

== HUEVOS ACTIVOS ==
ID: 1, Clase: good-egg, Posición: (3.8, 18.5) cm
ID: 2, Clase: broken-egg, Posición: (7.5, 10.4) cm
ID: 3, Clase: broken-egg, Posición: (19.0, 33.5) cm
Archivo 'huevos.csv' guardado.

0: 480x640 2 broken-eggs, 1 good-egg, 30.1ms
Speed: 1.4ms preprocess, 30.1ms inference, 11.3ms postprocess per image at shape (1, 3, 480, 640)

== HUEVOS ACTIVOS ==
ID: 1, Clase: good-egg, Posición: (3.8, 18.5) cm
ID: 2, Clase: broken-egg, Posición: (7.4, 10.4) cm
ID: 3, Clase: broken-egg, Posición: (19.0, 33.5) cm
  
```

Repositorio

<https://github.com/carlobeni/Egg-Detection-with-YOLO/tree/main>



Muchas gracias!