

# Reporte de Laboratorio 2

## Conteo de Frutos Pequeños en Tiempo Real Utilizando una Plataforma Giratoria

Ricardo Emmanuel Uriegas Ibarra\*

\*Ingeniería en Tecnologías de la Información  
Universidad Politécnica de Victoria

### Resumen—

#### I. INTRODUCCIÓN

La disponibilidad de metodologías prácticas y fiables de detección de frutos en campo resulta fundamental para realizar previsiones precisas de la cosecha[<empty citation>].

#### II. DESARROLLO EXPERIMENTAL

En este trabajo se presenta un sistema de detección en tiempo real de frutos pequeños que giran en una plataforma giratoria.

Para lograr esto se necesita conocimiento en las siguientes áreas:

- Procesamiento de Imágenes
- Visión por Computadora
- Redes Neuronales

La mayor parte de estos conocimientos los hemos aprendido en unidades previas de la materia, por lo que no se abordarán en este reporte.

##### II-A. Dataset

Antes de todo debemos definir que es un dataset y porque es que lo ocupamos en este trabajo. Un dataset es un conjunto de datos que se utilizan para entrenar un modelo de machine learning. En este caso, el dataset que se utilizó fue completamente recolectado por nosotros mismos[<empty citation>], el cual consiste en imágenes de frutos pequeños en diferentes entornos; el dataset se compone de 1 sola clase, la cual es la de frutos. La cantidad de imágenes recolectada fue de 38 imágenes, pero realizando aumentación de datos se logro aumentar a 114 imágenes guardadas en roboflow[<empty citation>] para su uso en el entrenamiento del modelo.

Las técnicas de augmentation que se aplicaron a los datos fueron las siguientes:

- Flip: Horizontal, Vertical
- Rotación 90°: En sentido horario, antihorario, al revés
- Crop: 0 % mínimo, 20 % máximo
- Rotación: Entre -15° y +15°
- Shear: ±10° Horizontal, ±10° Vertical
- Hue: Entre -15° y +15°
- Saturación: Entre -25 % y +25 %
- Brillo: Entre -15 % y +15 %
- Exposición: Entre -10 % y +10 %

- Blur: Hasta 2.5px
- Ruido: Hasta 0.1 % de píxeles

##### II-B. Modelo

Debido a la baja cantidad de imágenes para entrenamiento que poseíamos, sumado a la eficiencia que debía de poseer el modelo para ser ejecutado en tiempo real, se opto por usar el modelo pre-entrenado para detección de objetos de la librería ultralytics[<empty citation>] yolov8n[<empty citation>].

##### II-C. Entrenamiento

Una vez ya se tenia la selección del modelo pre-entrenado a usar, se prosiguió con el entrenamiento del modelo dado el dataset mencionado anteriormente[<empty citation>].

Para el entrenamiento se realiza lo siguiente:

- Se divide el dataset en 3 partes, una para entrenamiento, otra para validación y otra para pruebas (80 % para entrenamiento, 10 % para validación y 10 % para pruebas).
- Se realiza el entrenamiento del modelo.
- Después se realiza la validación del modelo.
- Y por ultimo; haciendo uso de la sección del dataset para pruebas, se realiza la prueba a cada época modelo y al final se escoge el mejor modelo.

##### II-D. Uso del Modelo

Para usar el modelo en el código se agrego un archivo (UI.py) que haciendo uso de una interfaz gráfica en Qt6, carga el modelo entrenado y realiza la detección de los frutos en tiempo real mediante la cámara de la computadora.

Este codigo principalmente busca el modelo best1.onnx, debido a que en un principio se buscaba entrenar y guardar el modelo en este formato, pero sin embargo al final se opto por guardar el modelo en formato .pt debido a que se tenia problemas al cargar el modelo en formato onnxn en la interfaz gráfica en una de las computadoras.

### III. RESULTADOS

### IV. CONCLUSIÓN

#### REFERENCIAS

- [1] *Calculadora de Horas*. <https://www.calculator.net/hours-calculator.html>. Consultado el 19-03-2025.
- [2] *Cálculo de Rangos de Fechas en Línea*. <https://www.example.org/rangos-de-fechas>. Consultado el 19-03-2025.