

Trabajo Práctico Final.  
Fundamentos Matemáticos de Visión en Robótica (67.61)  
2025

## Enunciado.

### **Opción 1: Clasificador binario de defectos en piezas de acero. (nota máxima 7)**

DATASET: <https://drive.google.com/file/d/1U68yuhFy08tLrte4Klatu1X-LgbSpSCN/view?usp=sharing>

### **Opción 2: Clasificador multiclase de defectos en piezas de aluminio.**

DATASET: [https://drive.google.com/drive/folders/1Xq9rSOFiS7Ked\\_FEGl7a3htYH4Q74OI6?usp=sharing](https://drive.google.com/drive/folders/1Xq9rSOFiS7Ked_FEGl7a3htYH4Q74OI6?usp=sharing)

REF. PAPER: [https://drive.google.com/file/d/1\\_unTuOZx9jJ9\\_6YbBfWrfsgISPq1TxAu/view?usp=drive\\_link](https://drive.google.com/file/d/1_unTuOZx9jJ9_6YbBfWrfsgISPq1TxAu/view?usp=drive_link)

### **Opción 3: Presentar la solución a un problema de computer vision que implique el entrenamiento de un modelo de deep learning , la creación de un dataset y una comparación de métricas con otra solución preexistente.**

Items comunes a todos los enunciados


1. Análisis exploratorio del dataset:
  - a. Indicar la cantidad de muestras disponibles por cada clase. ¿Hay desbalance?
  - b. En caso de que exista un desbalance fuerte (diferencia entre muestras > 30%) qué técnicas de data augmentation podría utilizar para ampliar el dataset desbalanceado?
  - c. Visualizar un ejemplo por cada defecto observado
  
2. Creación de un baseline:
  - a. Plantee un algoritmo de procesamiento de imágenes clásico para aislar (detectar) imágenes con algún defecto. No es necesario que en este paso la solución sea perfecta ni que clasifique el tipo, pero dada una imagen sana y otra defectuosa debería poder detectar el defecto en una y no detectar nada en otra. Evalúe su solución para 100 imágenes sanas y 100 defectuosas mediante una matriz de confusión y un reporte de clasificación. Separe estas imágenes ya que deberán excluirse del dataset de entrenamiento del modelo en el ítem 3.

AYUDA: Un posible algoritmo (a grandes rasgos) podría ser: Binarización -> Extracción de contornos -> filtrar por área, longitud, etc... Puede elegir las imágenes con los defectos más notorios para facilitar el trabajo en caso de que opte por la opción con el dataset multiclase.
  
3. Entrenamiento y evaluación.
  - a. Cree un dataloader para cargar los datos.
  - b. Entrene un clasificador, binario o multiclase según corresponda, para clasificar las muestras. Puede construir la arquitectura utilizada en el artículo de referencia, crear un modelo propio o utilizar algún modelo pre existente incluido en alguna biblioteca como un *visión transformer* del paquete *transformers* de huggingface.
  - c. Grafique la función costo en función de las epoch para los datasets de entrenamiento y testing.
  - d. Una vez finalizado el entrenamiento genere una matriz de confusión y un reporte de clasificación para el modelo.
  - e. Compare el modelo con el algoritmo del ítem 2 evaluando sobre el conjunto de imágenes separado en ese inciso.

## Pautas para la presentación.

Se deberá enviar 48hs antes del día del examen un link de un repositorio de código que contenga una jupyter notebook con la solución y el informe del trabajo realizado incluyendo una justificación para todas las decisiones adoptadas. El informe se debe realizar en la misma jupyter notebook documentando cada paso.

La defensa del trabajo se realiza en fecha de coloquio mediante una presentación de **15 minutos** seguida de un intervalo de 15 minutos donde te interrogamos sobre tu trabajo y **también sobre los contenidos de toda la materia.**

 Tips para levantar datasets en colab.ipynb