

# Classification and Localization of Flash, Cuts, Normal Of Given Image

## Basic Algorithm :

### ◊ Algorithm Steps

#### 1. Input Data

- a. Load donut dataset images (JPEG/PNG)
- b. Each annotated with bounding boxes ( $[x, y, w, h]$ ) and class ID ..

#### 2. Data Augmentation to Create a Duplicate Dataset(Preprocessing)

- a. Apply transformations to increase dataset variability and reduce overfitting:
  - i. RandomHorizontalFlip (mirror donuts left-right).
  - ii. RandomRotation (simulate rotated donuts).
  - iii. ColorJitter (brightness, contrast changes).
  - iv. Optional: RandomAffine (shift/scale/rotate for diversity).
- b. These augmentations create synthetic variations, making the model more robust.
- c. Create 95 data sets with annotation using Roboflow(<https://app.roboflow.com/robbery-dataset>)

#### 3. Model Training

- a. Use Faster R-CNN (ResNet-50 backbone).
- b. Fine-tune for 3 classes: Background , Cut , Flash .
- c. Train on augmented dataset for ~10 epochs (or more if dataset is small).
- d. Save model

#### 4. Inference (Defect Detection)

- a. Load trained Faster R-CNN model in inference mode (`eval()`).
- b. Input a test image → model outputs:
  - i. Bounding boxes
  - ii. Labels (Cut/Flash/Background)
  - iii. Confidence scores

#### 5. Decision Logic

- a. If  $\text{score} \geq 0.5$  and label = Cut/Flash → classify as defect.
- b. If no defect predictions above threshold → classify as Normal.

## 6. Output Processing

- a. Console log defect type, confidence score, and bounding box.
- b. Draw bounding boxes on defective images.
- c. Label as “Normal” if no defect is found.

## Block Diagram

