WASTE DETECTION IN RAILWAY PLATFORMS

AIM:

To detect the waste on the railway platform floor and alert the cleaning team to clean the waste.

PROCEDURE:

1. **Dataset Collection**
   - Initially, we went to Taramani railway station to capture videos of the railway platforms in order to obtain the dataset for waste detection.
   - We collected various samples of waste images like bottles,wrappers, papers, and foil.

2. **Dataset Processing**

   - We converted the video into frames using Python.

```
print(count) #To give the count of the no.of frames collected from the video sample
```

```
7774
```

Figure 1: Gives the count value of the number of frames

Figure 2: Converts the video into image frames

3. Dataset Labeling

- After converting the video into frames, these frames are given as input images for custom training.
- The bounding boxes were labeled manually for all the frames.
- The classes of waste are:
    - Bottle
    - Paper
    - Wrapper
    - Foil
- The data was then split into train data, test data, and validation data, and then pre-training of the model took place.
- After the pre-training, the dataset was exported from the custom trained model to Google Colab for post training of the model.

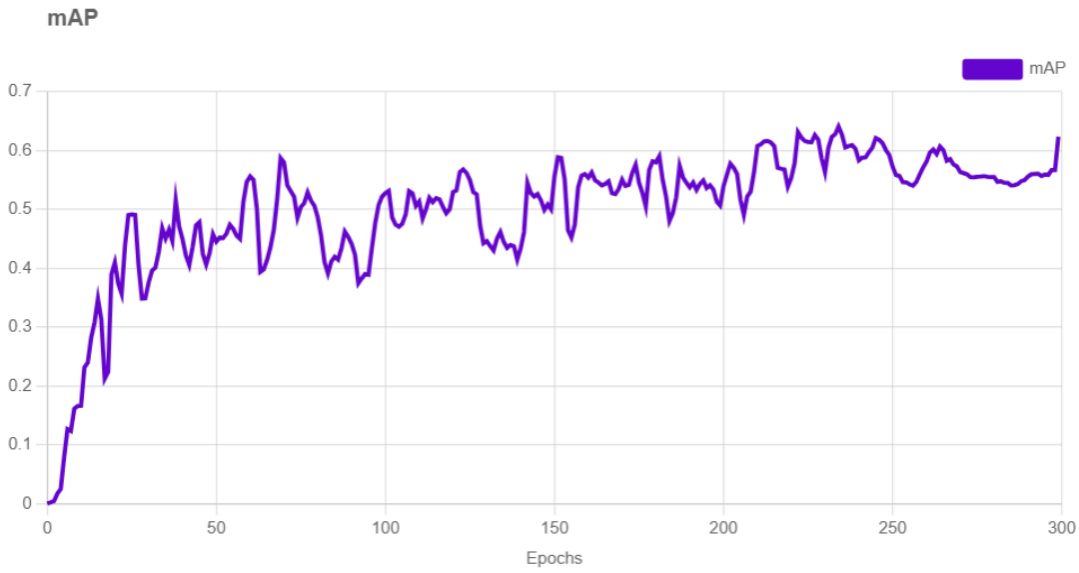

Figure 3: mAP graph

Figure 4: Loss graphs

4. Significance of the losses

(i) mAP (mean Average Precision)

Significance: mAP is a commonly used metric for evaluating the accuracy of object detection models. It measures both precision and recall, considering the precision-recall curve for different confidence thresholds.

Interpretation: A higher mAP indicates better performance, with a value of 1.0 representing perfect precision and recall.

(ii) Box Loss

Significance: Box loss is a component of the overall loss function used during the training of the model. It penalizes the model for incorrect predictions in terms of bounding box coordinates (x, y, width, and height).

Interpretation: Lower box loss values suggest better accuracy in predicting the bounding box locations of objects.

(iii) Class Loss

Significance: Class loss is another component of the loss function that penalizes the model for errors in predicting the class of detected objects.

Interpretation: Lower class loss values indicate better accuracy in assigning the correct class labels to the detected objects.

(iv) Object Loss

Significance: Object loss penalizes the model for failing to detect an object when it is present or for detecting an object when it is not present. It encourages the model to make confident predictions when objects are present and avoid false positives.

Interpretation: Lower object loss values suggest improved performance in terms of detecting objects and making accurate predictions about their presence.

5. Waste Detection

A one-shot pre-trained model is used for object detection purposes. In this algorithm, the model detects the objects with the help of dynamic bounding boxes that surround the object and segregate the object to the appropriate class or label.

- The model is trained after invoking the model function from the Ultralytics library.
- The labeled, custom, pre-trained model is invoked from roboflow into the Google Colab code.
- After the dataset is exported, training of the model takes place with this dataset, and then it undergoes the prediction and validation phases.

6. Testing

- Initially,when the platform does not have any waste objects, there are no bounding boxes to label the waste, which signifies that the platform is clean from waste and the cleaning team need not be alerted.

Figure 5: Clean Platform

- When there is waste in the testing image,

Figure 6: Output image after waste detection

RESULT AND INFERENCE:

```
Image(filename=f'/content/runs/detect/train2/results.png', width=600)
```

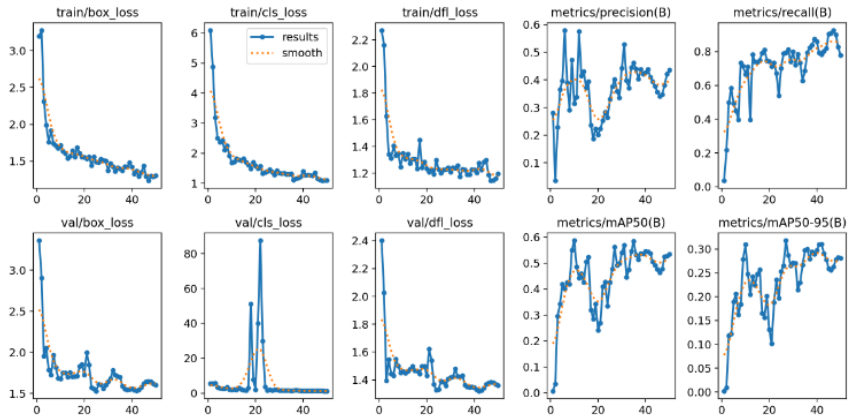

Figure 7: Post processing loss functions

In summary, these metrics and losses provide a comprehensive evaluation of the object detection algorithm. While mAP gives an overall measure of detection accuracy, box loss, class loss, and object loss help guide the training process by providing feedback on specific aspects of the

model's performance. During training, the goal is to minimize these losses to improve the accuracy and reliability of the object detection model.

```
[ ]  Image(filename=f'/content/runs/detect/train2/labels.jpg', width=600)
```

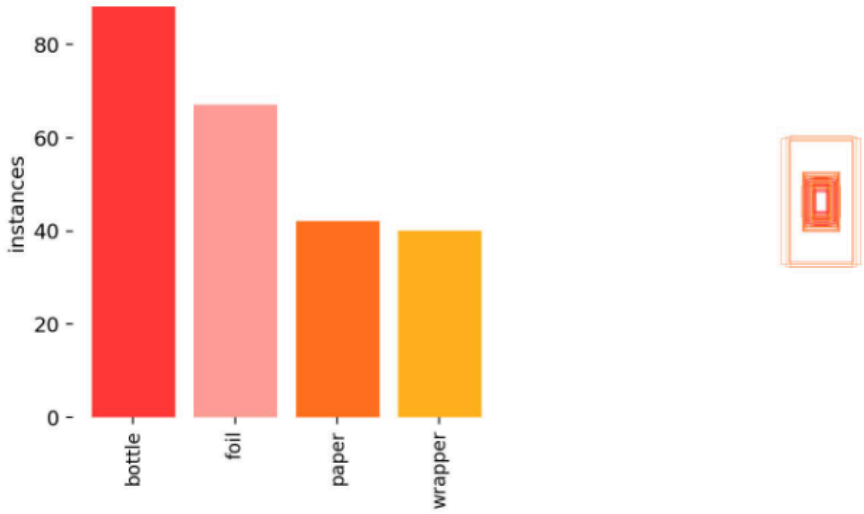

Figure 8: Post Processing label values

The above image signifies the number of instances in which the algorithm was able to predict the waste accurately.

Here is a link to the video of the demonstration of waste detection.