

## Model Development Phase Template

Date	2 Oct 2024
Team ID	739704
Project Title	Railway Sentry: Detecting Workers on Railway Tracks using YOLO V9
Maximum Marks	10 Marks

### Initial Model Training Code, Model Validation and Evaluation Report

For Railway Sentry's YOLO V9 model training, we configure the model with annotated track images, adjust hyperparameters, and initiate training. Validation and evaluation employ metrics like precision, recall, and mAP to ensure reliable worker detection accuracy.

#### Initial Model Training Code (5 marks):

```
from ultralytics import YOLO

# Load a model
model = YOLO("yolov9s.pt") # load a pretrained model (recommended for training)

# Train the model
results = model.train(data="/content/drive/MyDrive/Railway_sentry.v1i.yolov9/data.yaml", epochs=100, imgsz=642)
```

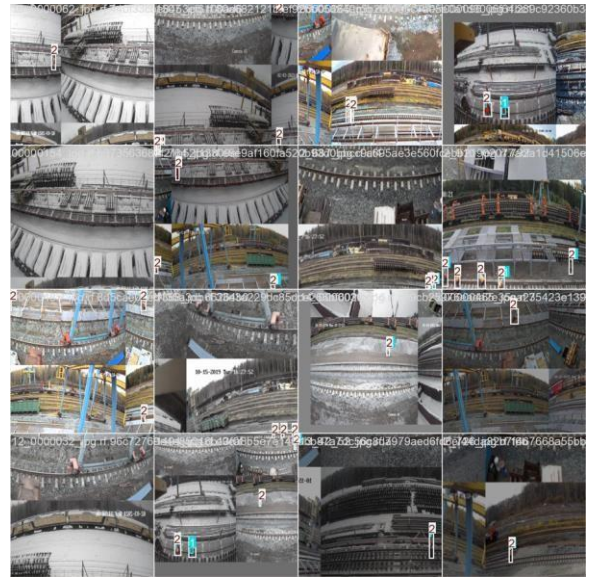
#### Model Validation and Evaluation Report (5 marks):

Model	Summary	Training and Validation Performance Metrics

## Model 1

```

YOLOv9s summary (fused): 488 layers, 7,168,249 parameters, 0 gradients, 26.7 GFLOPs
Class      Images  Instances  Box(P)      R      mAP50  mAP50-95: 100%
all         192       879       0.986       0.987   0.951   0.701
Helmet      147       350       0.878       0.865   0.931   0.593
Person      190       529       0.934       0.949   0.97    0.809
Speed: 0.3ms preprocess, 9.2ms inference, 0.0ms loss, 7.9ms postprocess per image
  
```



Model 2

```
results = model.val(data="/content/drive/myDrive/Railway sentry.v7i.yolov9/data.yaml", epochs=8)

WARNING ⚠️ imgs-[642] must be multiple of max stride 32, updating to [672]
Ultralytics 8.3.24 64 python-3.10.12 torch-2.5.0rcu21 CUDA=0 (Tesla T4, 15102MiB)
YOLOv9s summary (fused): 486 layers, 7,168,249 parameters, 0 gradients, 26.7 GiB
val: Scanning /content/drive/myDrive/Railway sentry.v7i.yolov9/valid/labels.cache... 192 images
Class Images Instances Box(P R mAP50 mAP50-95)
all 192 879 0.900 0.905 0.95 0.708
Person 147 350 0.878 0.863 0.932 0.604
Helmet 190 529 0.935 0.947 0.969 0.811
Speed: 0.5ms preprocess, 14.1ms inference, 0.1ms loss, 4.1ms postprocess per image
results saved to runs\dataset\features
```

