Visual Recognition HW2

111550034 黃皓君

Git link: Link

1. Introduction

The task involves digit recognition using an adapted Faster R-CNN framework. The implementation aims to enhance model robustness and prediction accuracy by introducing custom modifications. A modified Faster R-CNN model based on the **fasterrcnn_resnet50_fpn_v2** architecture is utilized. Inference outputs are post-processed to generate whole-number predictions from individual digit detections.

2. Method

Data Augmentation:

- Conversion to Tensor.
- Normalization: Each image is normalized to mean = [0.485, 0.456, 0.406] and std = [0.229, 0.224, 0.225], consistent with ImageNet statistics.
- Transformation Composition.

Model Architecture and Hyperparameters

- **Backbone:** ResNet-50 based Feature Pyramid Network (FPN), pre-trained on ImageNet (via PyTorch's ResNeXt101_32X8D_Weights.IMAGENET1K_V2).
- **Optimizer**: Stochastic Gradient Descent (SGD) with:
  - **Learning rate** = 0.005
  - **Momentum** = 0.9
  - **Weight decay** = 0.0005
- **Scheduler**: StepLR with step_size = 3 and gamma = 0.3
- **Batch size**: 4
- **Number of epochs**: Up to 10

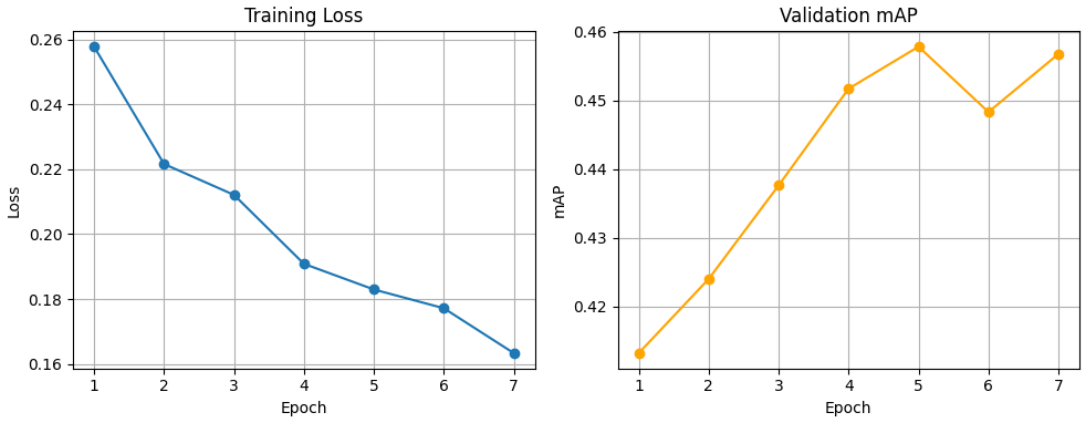
Training Strategy

The system is trained to optimize both localization and classification using multiple architectures, including ResNet-50, ResNet-50_v2, and MobileNet. During inference, the predictions from these different models are combined using Non-Maximum Suppression (NMS) to produce a robust set of detections for Task 1. The combined predictions are used to compute the

overall mAP and generate the final detection JSON output. However, when these aggregated results were used to construct whole-number predictions for Task 2, the performance was unsatisfactory. Therefore, for Task 2 only the prediction results from the ResNet-50_v2 model were utilized to form the whole-number outputs.

3. Results

(**ResNet-50_v2**)Training loss curves and mAP validation(on confidence = 0.7) shown below.



The validation mAP reaches 47.7% . On test is get 39% on mAP and 83% on accuracy.
After composing result from different models the test gets 41% accuracy on mAP.

4. Additional Experiments

Modified Loss Function Experiment:

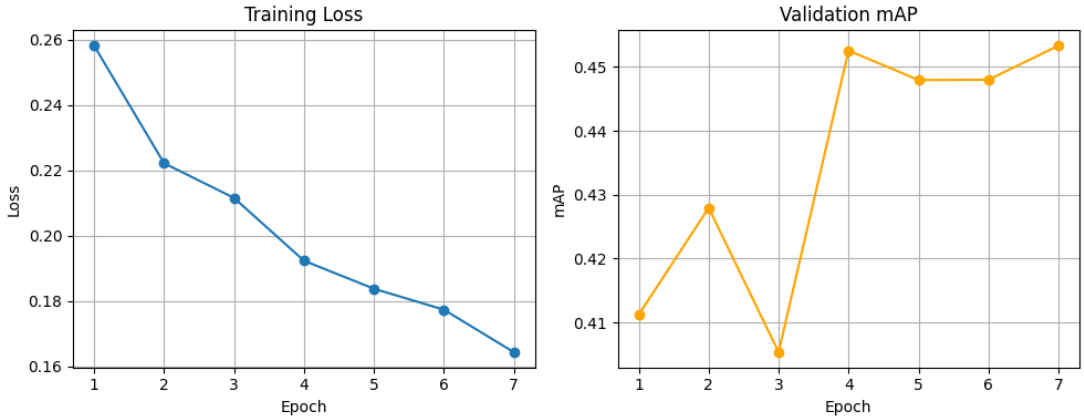
Hypothesis: Integrating a focal loss component into the classification branch mitigates the effect of class imbalance, especially for detecting small objects such as digits.

How This May Work:

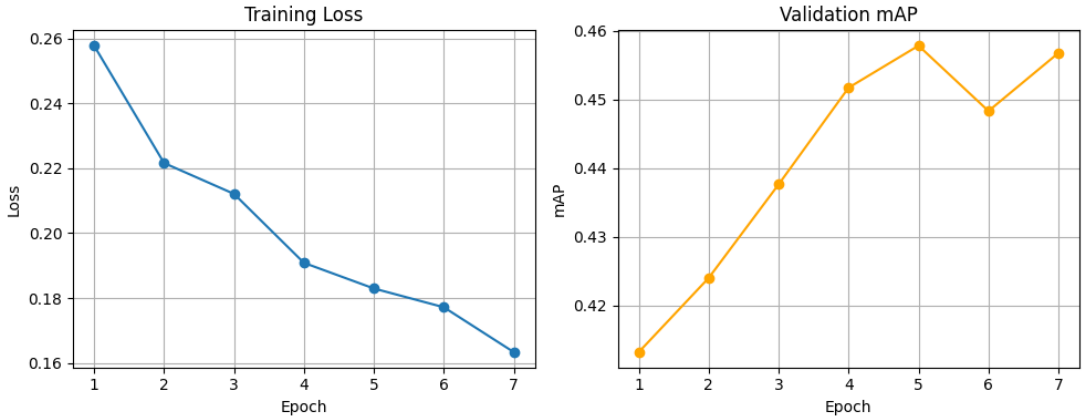
Focal loss reduces the impact of easy negative samples, directing focus toward more difficult examples. This adjustment helps in improving precision in the detection of under-represented digit classes.

Experiment: Despite similar training loss, validation mAP decreased when using focal loss. This may be due to suboptimal tuning of its hyperparameters (alpha, gamma) and an imbalance with the regression loss, leading to poorer generalization on the validation set.

Focal loss:



cross-entropy loss:



Combining Predictions

Hypothesis: Combining predictions from multiple outputs using Non-Maximum Suppression (NMS) can improve the overall detection quality by reducing the number of false positives and increasing the reliability of consistent detections.

How This May Work:

Aggregating predictions from several sources and applying NMS should merge overlapping bounding boxes into a single, more reliable detection. However, when these merged predictions are used to construct whole-number predictions (via sorting and concatenation of digits), inconsistencies may arise due to variations in confidence scores or misalignment among detections.

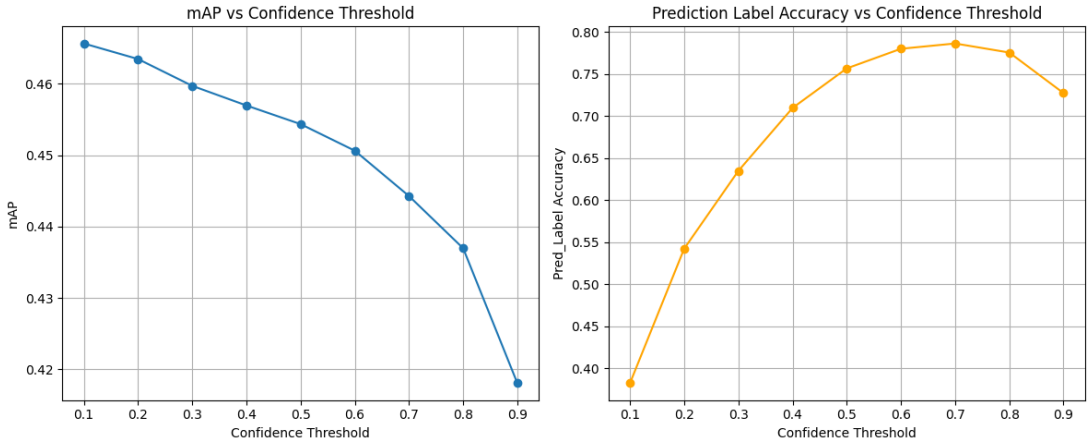
Experiment:

model	mAP (%)	accuracy (%)
ResNet50 FPN V2	39	**83**

ResNet50 FPN V2 (focal loss)	37	82
ResNet50 FPN	38	76
MobileNet V3	37	71
Combined (via NMS)	**41**	69

Threshold Evaluation:

Loops over a list of thresholds. Both mAP and predicted label accuracy are reported. So that I can choose the best threshold to generate prediction.



5. References

fasterrcnn_resnet50_fpn_v2 — Torchvision main documentation

torchvision — Torchvision 0.21 documentation

nms — Torchvision main documentation