**Visual Recognition HW3**

**111550034黃皓君**

**Git link:** [**Link**](https://github.com/qwer521/Visual-Recognition-Mask-RCNN)

1. **Introduction**

This task involves multi-class cell instance segmentation using the Mask R-CNN framework[1]. To investigate the effect of class-specific modeling, we conduct an ablation study where each semantic category is trained with a dedicated model. The goal is to evaluate whether isolated training improves performance over a shared unified detector. The inference results are then merged to generate the final submission.

**2. Method**

**Data Augmentation:**

* Conversion to Tensor.
* **Random Horizontal Flip** with 50% chance.
* **Brightness, Contrast, and Gamma Jittering** applied independently with random parameters:
  + Brightness factor ∈ [0.8, 1.2]
  + Contrast factor ∈ [0.8, 1.2]
  + Gamma factor ∈ [0.9, 1.1]

**Model Architecture and Hyperparameters**

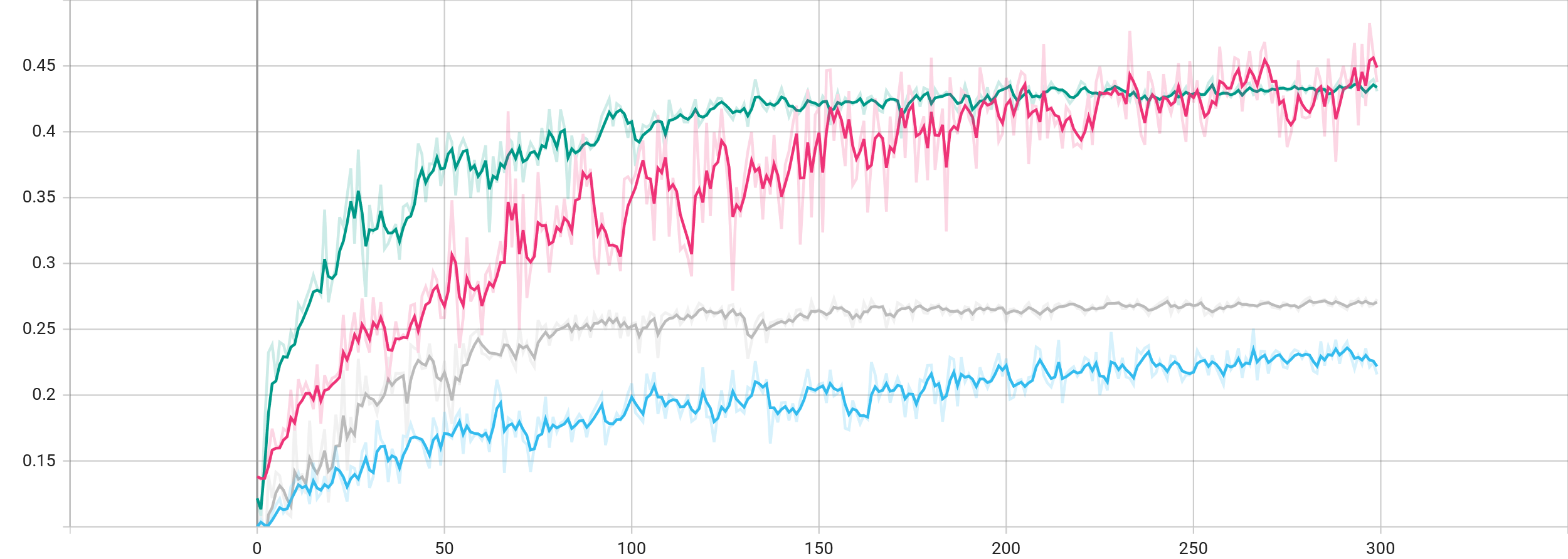
* **Backbone:** ResNet-50 + FPN, pre-trained on ImageNet (*maskrcnn\_resnet50\_fpn\_v2*).
* **Mask Branch:** 5×5 conv head followed by transposed convolution upsampling.
* **Head:** RoIAlign for feature pooling + class-specific *FastRCNNPredictor* and *MaskRCNNPredictor*.
* **Optimizer**: AdamW
  + lr\_backbone = 1e-5
  + lr\_heads = 1e-4
  + weight\_decay = 1e-4
* **Scheduler**: CosineAnnealingLR (min lr = 1e-6)
* **Batch size**: 2
* **Number of epochs**: 300

**Training Strategy**

* 4 separate models were trained, each handling **only one semantic class** (class1, class2, class3, class4).
* For each image, instance masks were extracted per class from .tif files, using instance ID > 0 as binary masks.
* When a sample contains **no valid instance** of the target class, a dummy zero-mask and background label is injected to maintain training stability.
* At inference time, four class-specific models were loaded and run independently over each test image.
* For each model, the class-specific label was assigned to all valid predictions (label == 1), and results were filtered based on a minimum score threshold of 0.01. After merging outputs from all models, we applied **NMS** with an IoU threshold of 0.5 to eliminate redundant or spatially overlapping masks across different classes, as commonly practiced in ensemble-based detection pipelines.

**3. Results**

Training AP50 of each classes shown below.

**

*Color: class-1 – blue, class-2 – pink, class-3 – green, class-4 – gray*

The Combined submission score is **32.51%**

**4. Additional Experiments**

**Class-wise Training and Inference:**

**Hypothesis:**

Training a separate Mask R-CNN model for each semantic class allows the network to focus on class-specific visual features without interference from inter-class competition. This may be especially beneficial when objects of different classes have very different shapes or scales.

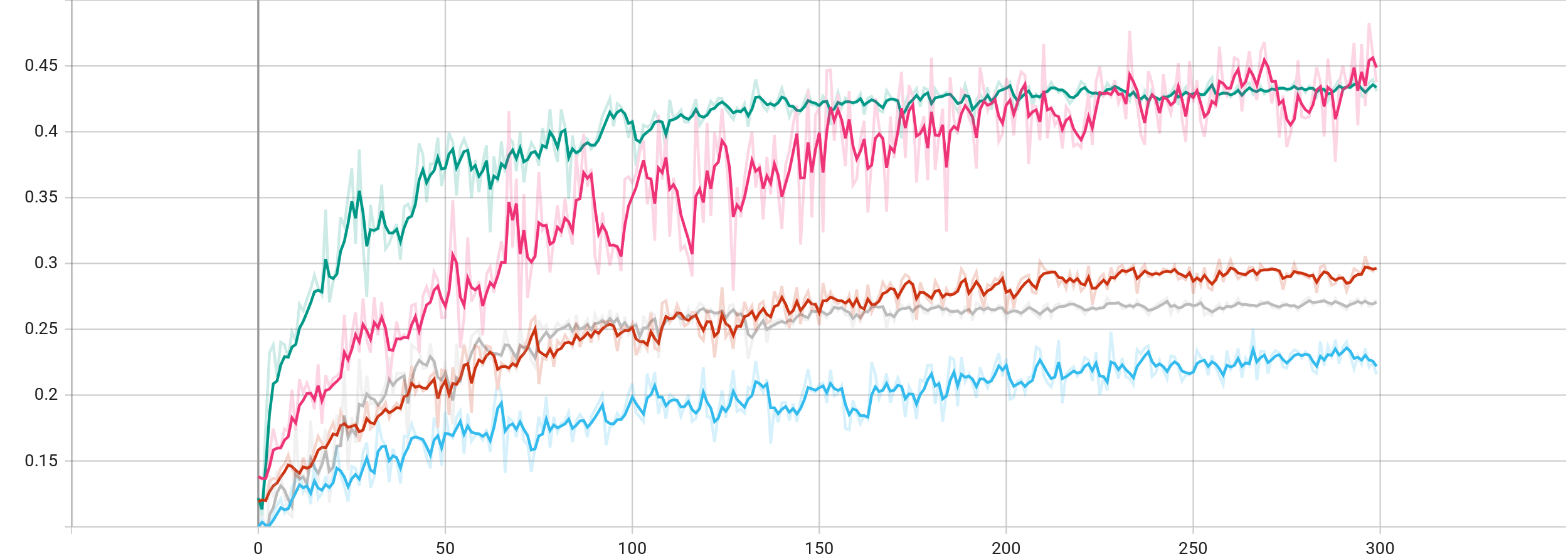
**How This May Work:**

By isolating training per class, the model simplifies its task into binary instance detection: “object or not” for a given class. This can potentially:

* Improve feature purity for small or rare classes
* Reduce class confusion in overlapping masks
* Allow customized hyperparameters or augmentation for each class

**Experiment:**

| **Model** | **Test AP (%)** | **Description** |
| --- | --- | --- |
| class1 | **22.18** | Trained exclusively on Class 1 |
| class2 | **44.88** | Trained exclusively on Class 2 |
| class3 | **43.37** | Trained exclusively on Class 3 |
| class4 | **27.07** | Trained exclusively on Class 4 |
| **Merged** | **32.51** | Combined predictions |
| **Baseline** | **31.67** | Standard multi-class Mask R-CNN |



*Color:* ***Baseline*** *– orange*

The class-wise training strategy yields a **2.65%** relative improvement over the baseline. This supports the claim that separating class training can enhance model accuracy, especially when coupled with proper instance handling and inference fusion.

**5. References**

[1] K. He, G. Gkioxari, P. Dollár, and R. Girshick, “Mask R-CNN,” in *Proc. ICCV*, 2017, pp. 2961–2969.  
[2] T.-Y. Lin, M. Maire, S. Belongie, et al., “Microsoft COCO: Common Objects in Context,” in *ECCV*, 2014.