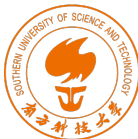


CS329 Machine Learning(H) Project Proposal: Modification of Focal Loss in Object Detection

Zubin Zheng: 12112328, Dahui Li: 12112411

Department of computer science and engineering, SUSTech

2023/12/9



① Introduction

② Literature Review

③ Methodology

④ Experiments

⑤ Conclusion

⑥ Reference

1 Introduction

2 Literature Review

3 Methodology

4 Experiments

5 Conclusion

6 Reference

Background

In autonomous driving, object detection and object tracking are crucial tasks that play a significant role in the decision-making process during the operation of intelligent vehicles.

We are interested in Camera-based 2D Object Detection.

- **Object detection:** an important computer vision task that deals with detecting instances of visual objects of a certain class (such as humans, animals, or cars) in digital images.
- **Goal:** develop computational models and techniques that provide the knowledge: What objects are where?
- **Metrics:** accuracy (including classification accuracy and localization accuracy) and speed[ZCS⁺23].

Background

- Object Detection Milestones

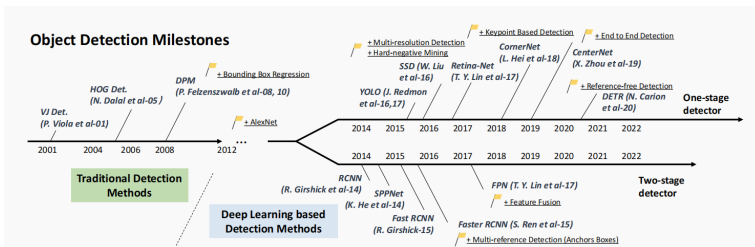


Fig. 1: A road map of object detection[ZCS⁺23].

- two-stage detectors: higher accuracy
- one-stage detectors: faster, simple
- question: could a simple one-stage detector achieve similar accuracy?

Motivation

- **Obstacle:** the extreme foreground-background class imbalance (T.-Y. Lin et al. found)[LGG⁺17].
 - training is inefficient as most locations are easy negatives that contribute no useful learning signal
 - the easy negatives can overwhelm training and lead to degenerate models.
- **Solutions**
 - hard negative mining
 - reshape loss function

Inspired by [LGG⁺17], we are interested in the design of loss function in RetinaNet. We want to explore whether we can improve the performance of object detection by modifying the loss function.

1 Introduction

2 Literature Review

3 Methodology

4 Experiments

5 Conclusion

6 Reference

Related work: **Loss Function**[ZCS⁺23]

- A general form of the loss function:

$$L(p, p^*, t, t^*) = L_{cls.}(p, p^*) + \beta I(t) L_{loc.}(t, t^*)$$
$$I(t) = \begin{cases} 1 & \text{IoU}\{a, a^*\} > \eta \\ 0 & \text{else} \end{cases}$$

- Classification loss: MSE/L2 loss, CE loss
Label Smooth:

$$H(q', p) = - \sum_{k=1}^K \log p(k) q'(k) = (1 - \epsilon) H(q, p) + \epsilon H(u, p)$$

- Localization loss:

$$Smooth_{L1}(x) = \begin{cases} 0.5x^2 & \text{if } |x| < 1 \\ |x| - 0.5 & \text{else} \end{cases}$$

Related work: Loss Function[LGG⁺17]

- Balanced Cross Entropy: $CE(p_t) = -\alpha_t \log(p_t)$
- Focal Loss: $FL(p_t) = -\alpha_t(1-p_t)^\gamma \log(p_t)$ where p_t is the probability of samples being correctly classified, α is a weighting factor, γ is a tunable focusing parameter.

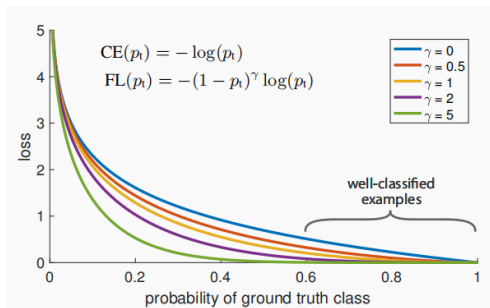


Fig. 2: Comparison of Cross Entropy and Focal Loss[LGG⁺17]

Related work: RetinaNet[LGG⁺17]

RetinaNet forms a single FCN comprised of a ResNet-FPN backbone, a classification subnet, and a box regression subnet.

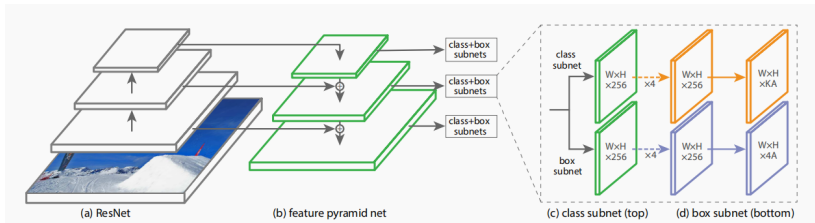


Fig. 3: The one-stage RetinaNet network architecture[LGG⁺17]

Focal Loss is used on the output of the classification subnet. RetinaNet is efficient and accurate: the best model, based on a ResNet-101-FPN backbone, achieves a COCO test-dev AP of 39.1 while running at 5 fps, surpassing the previously best published single-model results from both one and two-stage detectors.

1 Introduction

2 Literature Review

3 Methodology

4 Experiments

5 Conclusion

6 Reference

Novelty

- Possible drawback: Focal Loss down-weights all samples, for easy samples, the weight reduction is greater. However, for hard samples, it is not necessary to down-weight them.
- Novelty: We have modified the definition of Focal Loss to have the following properties:
 - down-weights easy samples
 - increases the loss weight of hard samples

Proposed method

- System setup: We still use the framework of RetinaNetNet, the only difference is that we have used the modified Focal Loss, denoted FL^* , and the settings of other parameters of the RetinaNet remain unchanged.
- Focal Loss* is defined as:

$$FL^*(p_t) = -\alpha_t \left(\frac{1-p_t}{p_t} \right)^\gamma \log(p_t)$$

where $\left(\frac{1-p_t}{p_t} \right)^\gamma$ is the modified modulating factor.

Proposed method

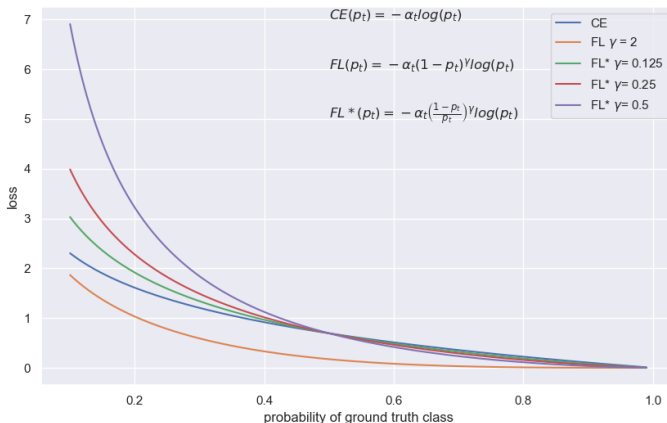


Fig. 4: Comparison of Cross Entropy, Focal Loss and Focal Loss*

1 Introduction

2 Literature Review

3 Methodology

4 Experiments

5 Conclusion

6 Reference

Experiments Setting

At present, we have found the RetinaNet source code on the GitHub repository and completed model training using Focal Loss on the coco dataset used in the paper.

(<https://github.com/0SliverBullet/CS329-Machine-Learning-H-Project>)

- Experiment Platform: the HPC-AI service platform from the Department of Computer Science and Engineering
- Resources:
 - Source code: <https://github.com/open-mmlab/mmdetection>
 - Dataset: coco2017
- Parameter Settings: epochs=12, optimizer='SGD', $\gamma=2.0$, $\alpha_t=0.25$

Initial Results

```
12/03 22:45:36 - mmengine - INFO - Checkpoints will be saved to /home/wangdx_lab/cse12112328/CS329Project/work_dirs/retinanet_r50_fpn_1x_coco.
12/03 22:46:00 - mmengine - INFO - Epoch(train) [1][ 50/58633] lr: 9.9098e-04 eta: 3 days, 20:13:50 time: 0.4719 dat
a time: 0.0087 memory: 3218 loss: 1.8995 loss_cls: 1.2174 loss_bbox: 0.6821
12/03 22:46:23 - mmengine - INFO - Epoch(train) [1][ 100/58633] lr: 1.9920e-03 eta: 3 days, 20:06:26 time: 0.4707 dat
a time: 0.0051 memory: 3217 loss: 1.9415 loss_cls: 1.2314 loss_bbox: 0.7101
12/03 22:46:46 - mmengine - INFO - Epoch(train) [1][ 150/58633] lr: 2.9930e-03 eta: 3 days, 18:58:35 time: 0.4541 dat
a time: 0.0049 memory: 3217 loss: 1.9006 loss_cls: 1.2133 loss_bbox: 0.6873
12/03 22:47:08 - mmengine - INFO - Epoch(train) [1][ 200/58633] lr: 3.9940e-03 eta: 3 days, 18:12:45 time: 0.4501 dat
a time: 0.0049 memory: 3218 loss: 1.9125 loss_cls: 1.2185 loss_bbox: 0.6940
12/03 22:47:31 - mmengine - INFO - Epoch(train) [1][ 250/58633] lr: 4.9950e-03 eta: 3 days, 17:52:16 time: 0.4531 dat
a time: 0.0045 memory: 3218 loss: 1.8680 loss_cls: 1.1724 loss_bbox: 0.6957
12/03 22:47:56 - mmengine - INFO - Epoch(train) [1][ 300/58633] lr: 5.9960e-03 eta: 3 days, 19:07:31 time: 0.4987 dat
a time: 0.0046 memory: 3218 loss: 1.9160 loss_cls: 1.2232 loss_bbox: 0.6928
12/03 22:48:19 - mmengine - INFO - Epoch(train) [1][ 350/58633] lr: 6.9970e-03 eta: 3 days, 18:43:58 time: 0.4526 dat
a time: 0.0050 memory: 3217 loss: 1.9109 loss_cls: 1.1982 loss_bbox: 0.7127
12/03 22:48:41 - mmengine - INFO - Epoch(train) [1][ 400/58633] lr: 7.9980e-03 eta: 3 days, 18:28:12 time: 0.4540 dat
a time: 0.0048 memory: 3217 loss: 1.8823 loss_cls: 1.2150 loss_bbox: 0.6672
12/03 22:49:03 - mmengine - INFO - Epoch(train) [1][ 450/58633] lr: 8.9990e-03 eta: 3 days, 17:35:57 time: 0.4233 dat
a time: 0.0049 memory: 3218 loss: 1.8853 loss_cls: 1.2151 loss_bbox: 0.6702
12/03 22:49:27 - mmengine - INFO - Epoch(train) [1][ 500/58633] lr: 1.0000e-02 eta: 3 days, 18:06:14 time: 0.4849 dat
a time: 0.0049 memory: 3217 loss: 1.8919 loss_cls: 1.2037 loss_bbox: 0.6882
12/03 22:49:51 - mmengine - INFO - Epoch(train) [1][ 550/58633] lr: 1.0000e-02 eta: 3 days, 18:22:16 time: 0.4768 dat
a time: 0.0049 memory: 3219 loss: 1.8940 loss_cls: 1.2155 loss_bbox: 0.6785
12/03 22:50:13 - mmengine - INFO - Epoch(train) [1][ 600/58633] lr: 1.0000e-02 eta: 3 days, 18:05:59 time: 0.4465 dat
a time: 0.0049 memory: 3218 loss: 1.8842 loss_cls: 1.2098 loss_bbox: 0.6744
12/03 22:50:34 - mmengine - INFO - Epoch(train) [1][ 650/58633] lr: 1.0000e-02 eta: 3 days, 17:24:38 time: 0.4159 dat
a time: 0.0049 memory: 3217 loss: 1.8804 loss_cls: 1.1936 loss_bbox: 0.6868
12/03 22:50:58 - mmengine - INFO - Epoch(train) [1][ 700/58633] lr: 1.0000e-02 eta: 3 days, 17:53:35 time: 0.4930 dat
a time: 0.0049 memory: 3220 loss: 1.8802 loss_cls: 1.1997 loss_bbox: 0.6805
12/03 22:51:20 - mmengine - INFO - Epoch(train) [1][ 750/58633] lr: 1.0000e-02 eta: 3 days, 17:37:58 time: 0.4409 dat
a time: 0.0051 memory: 3217 loss: 1.8767 loss_cls: 1.1729 loss_bbox: 0.7038
```

Fig. 5: Training Process

Initial Results

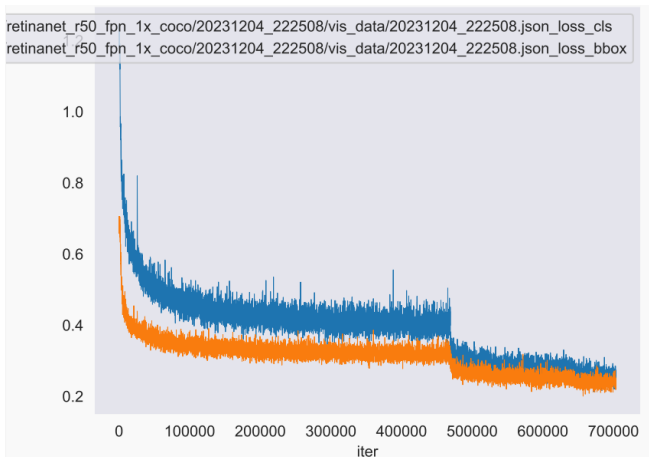
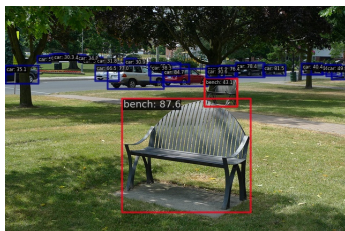


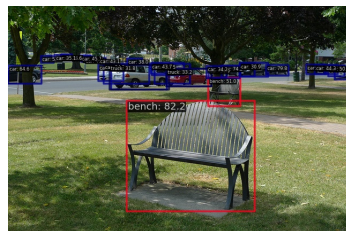
Fig. 6: The loss varies with the number of iterations

Initial Results

| detector | mAP | mAP_{50} | mAP_{75} | mAP_S | mAP_M | mAP_L |
|---------------------------|--------|------------|------------|---------|---------|---------|
| retinanet_r50_fpn_1x_coco | 0.3280 | 0.5010 | 0.3500 | 0.1740 | 0.3530 | 0.4380 |



(a) results of pretrained model



(b) results of our trained model

Fig. 7: Results Comparison of pretrained model and our trained model

Initial Results



Fig. 8: Object Detection in SUSTech Campus

1 Introduction

2 Literature Review

3 Methodology

4 Experiments

5 Conclusion

6 Reference

Goals and Objectives

- What we have done:
 - Literature review
 - Modified the Focal Loss
 - Run the source code in coco and get a trained model as baseline
- Goal: Performance comparison between original and modified loss functions, expect an improvement in performance(AP).

Task assignments and Project schedule

| Dates | Tasks | Remarks |
|--------|---|------------------------------|
| Week4 | Read the final project requirements | Final project is issued |
| Week5 | Assess difficulty level, decide which job to do and which task to be done: | |
| Week6 | Camera-based 2D Object Detection | |
| Week7 | Roughly read the papers in the task we have chosen | |
| Week8 | Choose the one that we are interested in: RetinaNet | |
| Week9 | Search for the source code on the GitHub repository | |
| Week10 | | |
| Week11 | Thoroughly read the paper: Focal Loss for Dense Object Detection Read the source code Download the original dataset coco used in the paper Compare the running results with the experimental results in the paper | |
| Week12 | Modify the definition of loss function and run on coco Prepare for the proposal presentation and write proposal report | Final project proposal |
| Week13 | Find other different datasets and test on them | |
| Week14 | Performance comparison between original and modified loss functions tune parameters, analysis, modification on loss function | |
| Week15 | Test our code in SUSTech campus Summarize this project and write the final report | Final project submission DDL |

1 Introduction

2 Literature Review

3 Methodology

4 Experiments

5 Conclusion

6 Reference

[LGG⁺17] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár.

Focal loss for dense object detection.

In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.

[ZCS⁺23] Zhengxia Zou, Keyan Chen, Zhenwei Shi, Yuhong Guo, and Jieping Ye.

Object detection in 20 years: A survey.

Proceedings of the IEEE, 2023.

Thanks!