|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| of the head 256 96 96 96 96 | SE V | Learning Rate Decay | Warm-up | 0.6821 0.6677 0.6952 0.7034 0.7349 | 0.5560 0.5524 0.5413 0.5404 0.5420 | HMean 0.6127 0.6046 0.6087 0.6112 0.6239 | Size (M) 7 4.1 2.6 2.6 2.6 | (CPU, ms) 406 213 173 173 173 |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |

Table 3: Ablation study of inner\_channel of the head, SE, cosine learning rate decay, learning rate warm-up for text detection.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Task Text Detection Direction | Initial Learning Rate 0.001 0.001 | Batch Size 16 512 | Ablation Total Data Data | |
| 400 100 | 60 100 |
| Classification Text Recognition | 0.001 | 1024 | 500 | 100 |
|  |  |  |  |  |
|  |  |  |  |  |

Table 4: Implementation details of the model training..

|  |  |  |  |
| --- | --- | --- | --- |
| Backbone MobileNetV3\_ large\_x1 | HMean 0.6463 | Model Size (M) 16 | Inference Time (CPU, ms) 447 |
| MobileNetV3\_ large\_x0.5 MobileNetV3\_ large\_x0.35 | 0.6127 0.5935 | 7 5.4 | 406 367 |
| MobileNetV3\_ small\_x1 | 0.5919 | 7.5 | 380 |
|  |  |  |  |
|  |  |  |  |

Table 5: Compare the performance of the different back bones for text detection.

ference time of the different scales of MobileNetV3 change greatly. In PP-OCR, we choose MobileNetV3\_large\_x0.5 to balance accuracy and efficiency.

balance accuracy and efficiency. Tabel 3]shows the ablation study of inner channel of the head, SE, cosine learning rate decay, learning rate warm-up for text detection. Firstly, by reducing the internal channels of the detector head from 256 to 96, the model size was re- duced by 41%, and the inference time was accelerated by nearly 50% with HMean only dropped slightly. Therefore, reducing the inner channel is an effective way to lighten the detector. Then, when remove the SE block of the de. tector backbone, the model size is reduced 36.6% and the inference time has accelerated 18.8% further. Meanwhile, HMean will not be affected. Therefore, for text detection, the accuracy improvement of SE blocks is limited, but the time cost is very high. Finally, using both cosine learning rate decay instead of the fix learning rate and learning rate

|  |  |  |  |
| --- | --- | --- | --- |
| Pruner V | HMean 0.6239 0.6169 | Size (M) 2.6 1.4 | (SD 855, ms) 164 133 |
|  |  |  |  |
|  |  |  |  |

Table 6: Ablation study of FPGM pruner for text detection.

|  |  |  |  |
| --- | --- | --- | --- |
| Backbone MobileNetV3\_ small\_x0.5 MobileNetV3 | Accuracy 0.9494 0.9403 | Model Size (M) 1.34 0.85 | Inference Time (CPU, ms) 3.22 3.21 |
| small\_x0.35 ShuffleNetV2\_ x0.5 | 0.9017 | 1.72 | 3.41 |
|  |  |  |  |
|  |  |  |  |

Table 7: Compares the performance of the different back- bones for direction classification.

warm-up, HMean will be improved obviously. At the same time, the model size and the inference time will not be af- fected. Cosine learning rate decay and learning rate warm- up are effective strategies for text detection.

up are effective strategies for text detection. Table6shows the ablation study of FPGM pruner for text detection. Using FPGM pruner, the model size is reduced 46.2% and the inference time has accelerated 18.9% on SD 855 device with HMean slightly dropped. Therefore, FPGM pruner can prune the text detection model effectively.

# 3.3Direction Classification

Table 7compares the performance of different backbones for direction classification. The accuracy of MobileNetV3 with difference scales (0.35, 0.5) are close. The model size and the inference time of MobileNetV3\_small\_x0.35 are much better. Besides, ShuffleNetV2 is used to train a di- rection classifier in some previous work. From the table, whether it's accuracy or the model size or the inference time, ShuffleNetV2 is not a good choice.

ShuffleNetV2 is not a good choice. Tabel9 shows the ablation study of data augmentation for direction classification. The baseline accuracy of text di- rector classify without data augmentation is only 88.79%. When we adopt BDA (base data augmentation), the accu- racy can boost 2.55%. We also verified that RandomErasing