

RefineDetLite: A Lightweight One-stage Object Detection Framework for CPU-only Devices

Chen Chen¹, Mengyuan Liu¹, Xiandong Meng², Wanpeng Xiao¹, Qi Ju¹
¹Tencent TEG AI ²The Hong Kong University of Science and Technology

Fbeckhamchen, mengyuanliu, wanpengxiao, damonju@tencent.com, xmengab@connect.ust.hk

Abstract

Previous state-of-the-art real-time object detectors have been reported on GPUs which are extremely expensive for processing massive data and in resource-restricted scenarios. Therefore, high efficiency object detectors on CPU-only devices are urgently-needed in industry. The floating-point operations (FLOPs¹) of networks are not strictly proportional to the running speed on CPU devices, which inspires the design of an exactly “fast” and “accurate” object detector. After investigating the concern gaps between classification networks and detection backbones, and following the design principles of efficient networks, we propose a lightweight residual-like backbone with large receptive fields and wide dimensions for low-level features, which are crucial for detection tasks. Correspondingly, we also design a light-head detection part to match the backbone capability. Furthermore, by analyzing the drawbacks of current one-stage detector training strategies, we also propose three orthogonal training strategies—Iou-guided loss, classes-aware weighting method and balanced multi-task training approach. Without bells and whistles, our proposed RefineDetLite achieves 26.8 mAP on the MSCOCO benchmark at a speed of 130 ms/pic on a single-thread CPU. The detection accuracy can be further increased to 29.6 mAP by integrating all the proposed training strategies, without apparent speed drop.

1. Introduction

Object detection is a fundamental technology in the computer vision society and is also a crucial component for many high-level artificial intelligence tasks, e.g., object tracking [59], vision-language transferring [9, 13], surveillance, autonomous driving [58] and robotics. Benefited from the rapid development of deep learning, the accuracy of object detection has been greatly improved. However, with the explosive growth of social media informa-

tion, the high computational complexity seriously hinders the wide applications of object detection algorithms. Therefore, much attention has been paid to the study of how to make trade-off between detection accuracy and implementation complexity. Thanks to the powerful parallel processing ability of GPUs, many researchers claimed they have achieved real-time detection. However, GPUs are still extremely high cost in terms of dealing with massive data. Consequently, research into fast object detection pipelines on computationally constrained devices (e.g., CPU-only computers and mobile devices) is extremely urgent.

Inspired by the pioneering deep-learning-based R-CNN serials ([20, 19, 47]), most state-of-the-art detectors are inclined to exploit classical classification networks [22, 53] as the backbone part. Obviously, the computational complexity of backbone networks is the important bottleneck that affects the running efficiency of the whole detector, and hence many lightweight algorithms employ famous efficient convolution networks [25, 49, 24, 62, 41, 27, 18]) instead. However, as pointed out in [35], there exists gaps between the design principles of classification and detection networks. For instance, the larger receptive fields and wider feature vectors of early stages are crucial for improving localization ability, while classification networks care only about the feature representation ability of the last layer. Therefore, directly employing classification networks as the backbones maybe not the optimal strategy. Additionally, another important issue that must be recognized is that the number of FLOPs is not strictly proportional to the running time since many other factors (e.g., memory access cost and degree of parallelism) impact the practical network latency [41]. Therefore, how to design an actually “fast” detection backbone network running on a CPU is a critical demand in industrial practice.

Typically, CNN-based object detectors are categorized into either two-stage detectors or one-stage detectors based on different processings of the detection part. Two-stage [47] detectors usually contain a region proposal network (RPN), a RoI warping module and a localization and classification subnet. More elegantly, one-stage de-

¹Here, FLOPs means the number of multiply-adds following [41].

