Transfer Rate Estimation in Edge-Cloud Neural Network Solution for Object Detection

胡 力博[†]　王 濤[†]　周 宇騁[†]　渡辺 裕[†]　　榎本 昇平[‡]　史　旭[‡]　坂本　啓[‡]　江田 毅晴[‡]
Libo Hu[†]　Tao Wang[†]　Yucheng Zhou[†]　Hiroshi Watanabe[†]　　Shohei Enomoto[‡]　Xu Shi[‡]　Akira Sakamoto[‡]　Takeharu Eda[‡]

[†] 早稲田大学　　[‡] NTT ソフトウェアイノベーションセンタ
Waseda University　　NTT Software Innovation Center

1. Introduction

Edge and cloud cooperative approach for object detection has been proposed [1][2]. Edge devices operate not only to acquire images but also to recognize specific objects. Light weight neural network such as MobileNet is a typical tool for mobile edge devices. However, edge only approach cannot take full advantage of the cloud's cognitive capabilities. Edge-cloud cooperative approach has been proposed to solve this problem.

Data of feature map should be transferred from edge to cloud. This is not a problem when the number of edges is small, but when the number is large, the transfer rate becomes a bottleneck.

Within the cooperative approaches [1], data transfer is realized by three ways (a) lossless, (b) quantized lossless, and (c) lossy compression. The distortion of feature map and transfer rate is in trade off relation. The edges may be IoT devices, and the number of edges may be more than one [2]. When edge can recognize specific objects by itself, transfer rate can be reduced. The self-cognitive capability is realized by equipping exit-points in the edge [3][4]. When edge has less confidence for the recognition result, the feature maps from the branch exit will be processed by int8 quantization to reduce transfer data and sent to cloud. The number of edges, size of feature map, lossless or lossy compression are the basic factors to determine transfer rate.

2. Framework

We propose a system-level solution for edge-cloud cooperative neural network for object detection, especially people recognition for surveillance system. As a typical neural network for the edge and cloud, YOLOv2 and modified YOLOv3 can be considered. BranchyNet can be incorporated into YOLOv2 at the edge to have exit-point capability. We applied BranchyNet in Darknet53 (backbone network of YOLOv3) as a simulation of the ideal Edge-cloud cooperative network. Fig. 1 shows the 3 exit-points we chose of Darknet53 in preliminary experiments. We chose the conv2 layer, conv5 layer and conv8 layer as the branch exits.

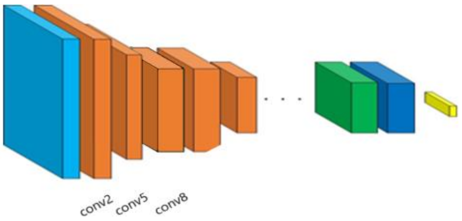

Fig. 1 Backbone of proposed Edge-cloud cooperative network

3. Cloud Operation and Investigation

In this paper, we only focus on the operation of the cloud side. In general, surveillance system to detect people at a shopping mall needs more than about 100 cameras. Let the number of cameras "m". The average number and variance of people capture in one camera are given by parameters "n" and "s". An input image is divided into 7x7 cells in YOLOv3. Ratio "r" denotes the area where people should be detected in cloud within cells. Note that the area which have been already recognized at the edge are omitted. Quantization accuracy loss which determines latency is confirmed in Table.1. The experiments include 2 datasets collected from ImageNet each with int8 quantized data from 3 branch exits.

Branch exits	Top1 of Fp32	Top1 of Int8	Loss
ImageNet1. Exit1	82.917%	75.688%	7.229%
ImageNet1. Exit2		77.771%	5.146%
ImageNet1. Exit3		78.729%	4.188%
ImageNet2. Exit1	70.305%	66.008%	4.297%
ImageNet2. Exit2		68.540%	1.765%
ImageNet2. Exit3		68.674%	1.631%

Table. 1 Estimate quantization accuracy loss

4. Conclusion

In this paper, we investigated an edge-cloud cooperative neural network with BranchyNet incorporated. By the preliminary experiment, int 8 quantization accuracy loss is confirmed and as a result it wouldn't' lead to a sharp drop in accuracy.

References

[1] H. Choi, and I.V. Bajic: "Deep Feature Compression for Collaborative Object Detection," IEEE International Conference on Image Processing (ICIP2018) WP. P6.8, Oct. 2018

[2] S. P. Chinchali, E. Cidon, E. Pergament, T. Chu, and S. Katti: "Neural Networks Meet Physical Networks: Distributed Inference Between Edge Devices and the Cloud," ACM Workshop on Hot Topics in Networks (HotNets2018), pp.50-56, Nov. 2018

[3] S. Teerapittaynon, B. McDanel, and H.T. Kung: "BrancyNet: Fast Inference via Early Exiting from Deep Neural Network," 2016 International Conference on Pattern Recognition (ICPR2016), pp.2464-2469, Dec. 2016

[4] E. Li, Z. Zhou, and X. Chen: "Edge Intelligence: On-Demand Deep Learning Model Co-Inference with Device-Edge Synergy," ACM Workshop on Mobile Edge Communications (MECOMM'18), pp.31-26, Aug. 2018