ICPR 2020
25th INTERNATIONAL CONFERENCE ON PATTERN RECOGNITION
Milan, Italy 10 | 15 January 2021

Compression of YOLOv3 via Block-wise and Channel-wise Pruning for Real-time and Complicated Autonomous Driving Environment Sensing Applications

Jiaqi Li*, Yanan Zhao*, Li Gao*, Feng Cui†

*Beijing Institute of Technology †Beijing Smarter Eye Technology Co. Ltd

Abstract: Nowadays, in the area of autonomous driving, the computational power of the object detectors is limited by the embedded devices and the public datasets for autonomous driving are over-idealistic. In this paper, we propose a pipeline combining both block-wise pruning and channel-wise pruning to compress the object detection model iteratively. We enforce the introduced factor of the residual blocks and the scale parameters in Batch Normalization (BN) layers to sparsity to select the less important residual blocks and channels. Moreover, a modified loss function has been proposed to remedy the class-imbalance problem. After removing the unimportant structures iteratively, we get the pruned YOLOv3 trained on our datasets which have more abundant and elaborate classes. Evaluated by our validation sets on the server, the pruned YOLOv3 saves 79.7% floating point operations (FLOPs), 93.8% parameter size, 93.8% model volume and 45.4% inference times with only 4.16% mean of average precision (mAP) loss. Evaluated on the embedded device, the pruned model operates about 13 frames per second with 4.53% mAP loss. These results show that the real-time property and accuracy of the pruned YOLOv3 can meet the needs of the embedded devices in complicated autonomous driving environments.

1. Method

For pruning the YOLOv3 model, both block-wise pruning and channel-wise pruning are performed iteratively. The steps taken are shown in Fig.1.

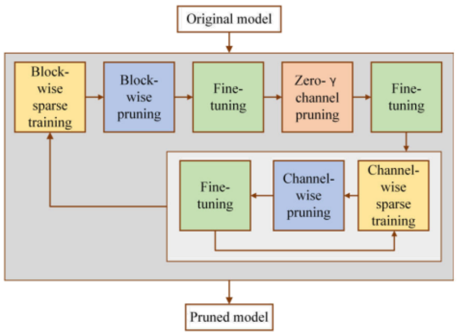

Fig. 1. The pipeline of model pruning. The block-wise pruning is performed iteratively and the channels are pruned iteratively after each block-wise pruning.

1.1 Block-wise Pruning

As Fig.2 shows, a scale factor λ is added to multiply with the output of the residual block. The absolute value $|\lambda^i|$ represents the importance of the block. We impose L1 regularization term on λ and use fast iterative shrinkage-thresholding (FISTA) algorithm to obtain sparse λ.

$$\mathbf{R}^{i+1} = \mathbf{R}^i + \lambda^i \mathcal{F}^i(\mathbf{R}^i, \mathbf{W}^i)$$

$$\mathcal{L}_{bloreg} = \zeta \sum_{\lambda^i \in \Lambda} \|\lambda^i\|_1$$

$$\text{Loss} = \mathcal{L}_{yolo} + \mathcal{L}_{bloreg}$$

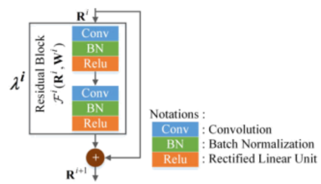

Fig. 2. Processing of YOLOv3 residual blocks. Every residual block of the YOLOv3 model has two conv-bn-relu groups.

The importance of the residual blocks can be sorted according to $|\lambda^i|$. The residual blocks with smaller $|\lambda^i|$ can be removed entirely.
After pruning, the model should be fine-tuned.

Fig.3 shows the distributions of λ with different ζ after sparsity training.

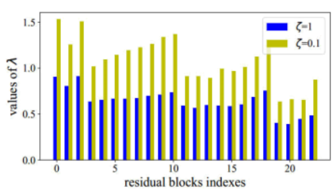

Fig. 3. The distributions of λ with different ζ after sparsity training.

1.2 Channel-wise Pruning

The absolute value $|\gamma^{i,j}|$ in each BN layer can reflect the importance of the channel. γ is also constrained by the L1-norm penalty, and we use SGD to optimize and get sparse gamma.

$$b_{out}^{i,j} = \gamma^{i,j} \frac{b_{in}^{i,j} - \mu_B^{i,j}}{\sqrt{\sigma_B^{i,j} + \varepsilon}} + \beta^{i,j}$$

$$\mathcal{L}_{chareg} = \xi \sum_{\gamma^{i,j} \in \Gamma} \|\gamma^{i,j}\|_1$$

$$\text{Loss} = \mathcal{L}_{yolo} + \mathcal{L}_{chareg}$$

Fig.4. shows the distributions of the scale factors γ in BN layers after channel-wise sparsity training with different ξ. Then we can remove the less important channels according to the sorted sparse γ. After pruning, the model should be fine-tuned.

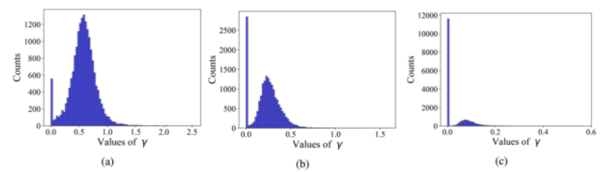

Fig. 4. Distributions of the scale factors γ in BN layers after channel-wise sparsity training with different ξ. (a) $\xi = 0.0001$, (b) $\xi = 0.001$, (c) $\xi = 0.01$.

2. Experiments and Results

2.1 Dataset

The classes in the datasets are defined as 'Body of Express Vehicle', 'Front of Express Vehicle', 'Rear of Express Vehicle', 'Body of Car', 'Front of Car', 'Rear of Car', 'Body of SUV', 'Front of SUV', 'Rear of SUV', 'Body of Minibus', 'Front of Minibus', 'Rear of Minibus', 'Body of Bus', 'Front of Bus', 'Rear of Bus', 'Body of Truck', 'Front of Truck', 'Rear of Truck', 'Pedestrian', 'Bicyclist', 'Motorcyclist' and 'Tricyclist'.

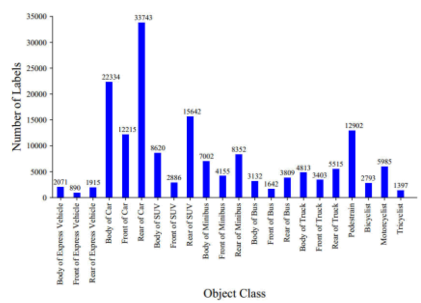

Fig. 5. Summary of our datasets. The histogram of each class shows that the problem of class-imbalance still presents.

There are 15,601 annotated static images including four kinds of scenes: freeway, urban road, suburb and residential area. Furthermore, the datasets also cover the scenes under poor illumination conditions such as the backlight scene.

2.2 Experiments and Results

The performance on the validation set of all models during iterative pruning. We choose YOLOv3-2nd-block-pruning-2nd-channel-pruning model as the final results of the experiments. The final model save 79.7% FLOPs, reduce 93.8% parameter size, compress 93.8% model volumes as well as save 45.4% inference times, with only 4.16% mAP declines.

TABLE I. EVALUATION OF BASELINE MODEL AND PRUNED MODELS

Models	FLOPs (G)	Parameter Size (M)	Volume (M)	Average Inference Time (ms)	mAP (%)	Precision (%)	Recall (%)
YOLOv3- baseline	81.169	61.637	246.8	11.89	79.4	68.1	84.3
YOLOv3-1st-block-pruning	63.593	33.035	132.3	9.05	78.3	44.0	86.0
YOLOv3-1st-block-pruning-1st-channel-pruning	51.380	19.417	77.8	7.78	78.0	44.4	86.0
YOLOv3-1st-block-pruning-2nd-channel-pruning	40.707	13.719	55.0	7.17	78.5	44.6	86.1
YOLOv3-2nd-block-pruning	37.511	11.832	47.4	6.55	77.4	41.4	86.1
YOLOv3-2nd-block-pruning-1st-channel-pruning	33.267	9.197	36.9	6.53	77.5	43.6	85.5
YOLOv3-2nd-block-pruning-2nd-channel-pruning	16.506	3.848	15.4	6.49	76.1	37.7	85.2
YOLOv3-2nd-block-pruning-3rd-channel-pruning	9.999	2.343	9.4	6.45	70.6	29.6	82.0
YOLO-tiny	6.807	8.719	34.9	2.01	54.5	34.6	63.8

The embedded device is a Xilinx® ZCU104 board with one B2304 core with 16 threads running at 330 MHz and DNNDK v3.0.

Deployed on the embedded device, he model can operate about 13 frames per second (FPS), which meets the needs of actual autonomous driving. Compared with the original model on the server, the mAP of the pruned model drops 4.53% of the mAP of the original model on the server.

original model on server **pruned model on embedded device**

Fig. 5 The detection results of the original model and YOLOv3-2nd-block-pruning-2nd-channel-pruning model on the embedded device

Acknowledgements: Jiaqi Li, Yanan Zhao and Li Gao are supported by a grant from the National Key R&D Program of China (no. 2017YFC0804808, no. 2017YFC0804803, no. 2018YFB0105205-02). Feng Cui is supported by a grant from the National Science and Technology Major Project from Minister of Science and Technology, China (no. 2018AAA0103100).

* Please contact *3120180345@bit.edu.cn* for further information.