17th International Conference on Machine Vision Applications (MVA)
Fully Online, July 25–27, 2021.

Critically Compressed Quantized Convolution Neural Network based High Frame Rate and Ultra-Low Delay Fruit External Defects Detection

Jihan Zhang[1], Dongmei Huang[2], Tingting Hu[1,2], Ryuji Fuchikami[2], Takeshi Ikenaga[1]

[1]Graduate School of Information, Production and Systems, Waseda University
Kitakyushu 808-0135, Japan
[2]Panasonic Corporation
Fukuoka 812-8531, Japan
florentino_zhang@fuji.waseda.jp

Abstract

*High frame rate and ultra-low delay fruit external defects detection plays a key role in high-efficiency and high-quality oriented fruit products manufacture. However, current traditional computer vision based commercial solutions still lack capability of detecting most types of deceptive external defects. Although recent researches have discovered deep learning's great potential towards defects detection, solutions with large general CNNs are too slow to adapt to high-speed factory pipelines. This paper proposes a critically compressed separable convolution network, and bit depth degression quantization to further transform the network for FPGA acceleration, which makes the implementation of CNN on High Frame Rate and Ultra-Low Delay Vision System possible. With minimal searched specialized structure, the critically compressed separable convolution network is able to handle external quality classification task with a minuscule number of parameters. By assigning degressive bit depth to different layers according to degressive bit depth importance, the customized quantization is able to compress our network more efficiently than traditional method. The proposed network consists 0.1% weight size of MobileNet (alpha = 0.25), while only a 1.54% drop of overall accuracy on validation set is observed. The hardware estimation shows the network classification unit is able to work at 0.672 ms delay with the resolution of 100*100 and up to 6 classification units parallelly.*

1. Introduction

Fruit external defects detection plays a key role in the process of product quality control. Accurate and efficient dislodgment of defective fruits leads to lower loss and less quality problems in final product. However, due to the variety of defects and similarity between stem/calyx and defection concavities, the current commercial solutions based on traditional computer vision still lack algorithms capable of detecting most types of external defects [1]. By contrast, classification methods based on learned features tend to be more effective for fruit defects detection. In the early years, researches [2][3] have already shown neural network's potential on defects detection. With the rapid progress of the understanding towards deep learning, much attention has been drawn to applicating neural network based methods to fruit defects detection. Z. da Costa et al. [4] applied ResNet to tomato external defects recognition, which reached 94.6% accuracy. Zhou et al. [5] achieved 93.8% average accuracy of green plum defects with a 16-layer CNN.

Although recent works achieved much better accuracy than traditional CV methods, none of them target to ultra-high processing speed. Zhou et al. [5] test each picture within 84.69 ms and Yadav et al. [6] identify each in 185 ms. In order to meet the rocketing demand of high production efficiency in advanced modern manufactory, external defects detection system with both high robustness and high processing speed remains an urgent need.

High Frame Rate and Ultra-Low Delay Vision System is the key to efficiency-oriented Factory Automation (FA) facilities. Using high-speed camera as input, specialized FPGA device is mapped with specialized image processing algorithms. Unlike a Von Neumann processor which is unable to process images until the image is stored in memory, such system is able to process images with and arithmetic unit directly connected to the image sensor by using FPGA. Serving as cerebellum-like role, it is able to output high-speed feedback to actuators. Recent works have made significant progress on such systems. Hu et al. [7] proposed local and global parallel based matching system, able to work at 1306 FPS and 0.808 ms delay. However, it still remains a formidable challenge to speed up CNN to millisecond-level. Firstly, there's a contradictory between huge computation complexity of general CNN and limited resources on FPGA. Therefore, a specially tailored tiny CNN is needed. Several works have focused on this direction. Hinton et al. [8] proposed knowledge distillation to train a tiny network with additional information from large trained models. However, it does not further explore the minimal size scale, which is especially important for hardware implementation. Secondly, 32-bit calculation in CNN on GPU is not hardware-friendly. In order to solve this problem, various quantization plans are proposed. Yoni et al. [9] proposed 4-bit integer quantization for deployment of pretrained models on limited hardware resources. Also, since the High Frame Rate Ultra-Low Delay architecture arithmetic resources, more specialized optimization of arithmetic's able to be performed on each layer individually.

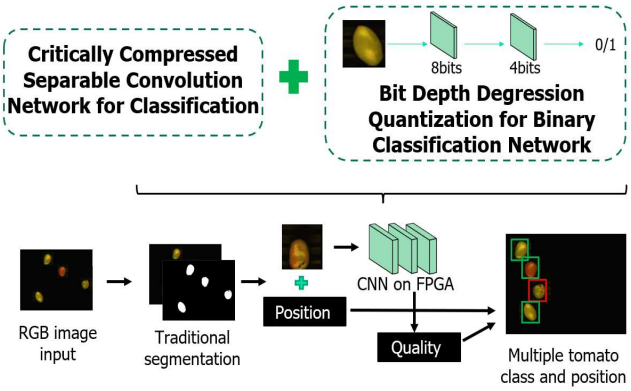

Figure 1. Framework of defects detection system.

Aiming at high frame rate and ultra-low delay processing for robust defects detection, this paper proposes (A) Critically compressed separable convolution neural network for classification. By building a plain separable convolution skeleton and implement critically compressed minimal structure searching, a CNN tailored for the certain task is obtained. (B) Bit depth degression quantization for binary classification network. The network is quantized in a specialized pattern, with lower bit depth in latter layers. By combining these two proposals, it's able to classify the defective and healthy fruits with high accuracy within millisecond-level speed.

2. Proposals

This section shows the algorithm framework of High Frame Rate Ultra-Low Delay fruit external defects detection system and the two proposals for the construction of a critically compressed quantized CNN tailored for this task. Fig. 1 shows the algorithm framework of the detection system. The input images are in dark background with several fruit instances in it. Firstly, a topological structure analysis based traditional segmentation method is considered to locate fruits' position, which is based on [10]. The precondition that allows us to implement this concise segmentation is, different from nature images, images shot from automation pipelines may share a fixed and simple background. Therefore, the position of each fruit is detected and the picture of each fruit is cut out. Secondly, a specially tailored CNN classifies the quality (defective or healthy) of each fruit instance. As the segmentation part largely decreases the input image pixel, the throughput of CNN is significantly decrease for each frame, which ensure the system's processing time closer to millisecond level.

Both proposals aim at the construction of a lightweight hardware-friendly CNN tailored for fruit external defects detection.

2.1 Critically compressed separable convolution network for classification

Tiny network size is a necessity for CNN-equipped High Frame Rate Ultra-low Delay System. This section introduces two steps to construct a critically compressed separable convolution network for classification. Firstly,

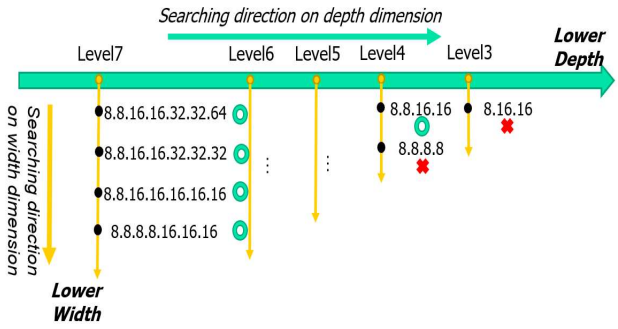

Figure 2. Concept of critically compressed minimal structure searching.

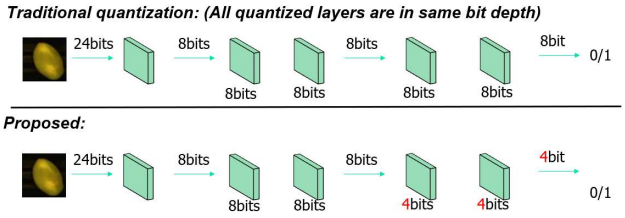

Figure 3. Concept of bit depth degression quantization for binary classification.

a plain network skeleton with separable convolution blocks is built. Secondly, critically compressed minimal structure searching is used to find the most suitable structure for the task.

In the first step, the shortcuts are eliminated. This is because, in a residual block, in order to add the input feature map to the feature map after the second convolution layer, the input feature map should be saved until the second convolution ends. The feature map contains most intermediate parameters of the inference process. Therefore, the existence of shortcut causes extra memory consumption, which is a precious resource for FPGA. By using the plain structure, it is able to release the input feature map as soon as the first convolution is done, while accuracy decrease is not significant in binary classification task.

Also, every traditional convolution is replaced with separable convolution kernels. There are mainly two reasons: (1) Separable convolution consumes less parameters and calculation than traditional convolution but remains similar capacity [11]. (2) Separable convolution largely reduces the data fan-in of the computation kernel. In hard-wired type CNN implementation, the hardware resource consumption of convolution kernel rises rapidly with the increase of data fan-in [12]. By replacing a K*K*n convolution kernel with n K*K depthwise kernel, the fan-in of one kernel decreases by n times.

For the second step, minimal structure searching is proposed to find the critically compressed network structure capable for our task. As shown in Fig. 2, values are gradually decrease on both depth dimension and width dimension following certain pattern.

The searching schedule is explained in the following. Before starting a searching, the acceptable accuracy decrease rate k and base model accuracy X are first set. Then for each model with a parameter number decrease Δ, the acceptable lowest accuracy Y is calculated by Y =

X-k Δ . When the actual accuracy is below Y, this training model is given up and searching on other models start. When all designed models with different width on a certain depth level are searched, then searching turns to a lower depth level. If unacceptable accuracy below Y is observed both on the lower depth and width level of one model, then the searching terminates and this network is selected as the most suitable model. For our task, a structure with 4 separable convolutional blocks (consist of 8, 8, 16, 16 channels) and one dense layer is selected following this searching schedule.

2.2 Bit depth degression quantization for binary classification network

Although the network size is significantly decreased through structure searching, it is still not suitable for direct implementation on FPGA because of the calculation is too resource-consuming. Several works focus on efficient quantization plan for neural networks, e.g. [13]. However, traditional quantization is unable to perform ideal compression efficiency on our binary classification network. Also, since High Frame Rate Ultra-Low Delay System doesn't share arithmetic resources, individual optimization of arithmetic is able to be implemented on each layer. Therefore, bit depth degression quantization is proposed for optimizing quantization bit depth for different layers. The concept of this method is shown in Fig. 3. Although mix precision quantization is discussed in several directions [14], the especial adaptability of bit depth degression style mix precision quantization to binary classification task isn't fully discussed.

In binary classification task, as output is only 1 bit (for our task, healthy or defective), bit depth of latter layers is not as important as the front ones. The necessary conditions of this design principle are (A) Bit depth of front layers is more important than it of latter layers. (B) Binary classification enhances the effect of condition A.

$$E(e_L^2) \approx E\left\|W_L n_{X_L}\right\|_F^2 + E\left\|n_{W_L} X_L\right\|_F^2 \quad (1)$$

$$E(e_L^2) \le E\left\|n_{W_L}\right\|_F^2 E\|X_L\|_F^2 + E\|W_L\|_F^2 E(e_{L-1}^2) \quad (2)$$

The mean square error (MSE) of the output of a linear layer and its quantized model is calculated by equation (1) [8]. Then, by using Cauchy Schwarz inequality and by assuming the weights/activations/noise are statistically independent, equation (2) is inferred, which is a recursive expression of the MSE caused by quantization. W_L stands for weight matrix of linear layer L. n_{W_L} stands for the average quantization noise in weight matrix of layer L. X_L stands for the activation matrix of layer L. n_{X_L} stands for the average quantization noise in activation matrix of layer L.

The expression in equation (2) indicates that quantization error accumulates through convolution layers, and lower bit depth causes larger quantization error, therefore lead to larger error in output. Therefore, using more precise in front layers are much more important than in latter layers, when total model size is limited. In other words, condition A is proved. Besides, condition B is an obvious fact, because binary class classification has better error tolerance than multiple class classification. Therefore, we have proved the design principle of bit depth degression quantization for binary classification.

In our implementation, in order to counteract the accuracy loss caused by bit depth degression, we decrease the bit depth only at where the channel doubles (in our network, it means the pointwise convolution which doubles the channel from 8 to 16). More channels mean more features are created to describe one class, which indicates more feature extraction ability. As it is used together with the bit depth decrease, it would counteract the impact that less precision on each feature brings.

With the bit depth degression, weight storage becomes half in latter layers. Also, the inference speed remarkably improved for part of 8-bit compute engines are replaced by 4-bit compute engines. More than that, our quantization schedule is several times more efficient than traditional quantization, which we would show in the evaluation section.

3. Evaluation Results

3.1. Overall classification performance

In this section, the classification performance of the proposed network is evaluated. We use the open-source dataset of [4] as our test dataset. This dataset contains more than 43 thousand 100*100*3 images of three Brazilian varieties of tomatoes. We select one of the most widely used lightweight network, MobileNet (α = 0.25) as baseline. Table 1 shows the comparation of accuracy result and Table 2 shows the comparation of model size. Our network consumes 0.1% size of MobileNet (α = 0.25), while only 1.54% overall accuracy drop and 11.37% defective accuracy drop on validation set is observed.

Table 1. Accuracy of baseline and our work.

Model	Val accuracy for healthy	Val accuracy for defective	Train accuracy	Val accuracy
MobileNet (α = 0.25)	99.50%	85.08%	99.96%	97.88%
Our work	99.31%	73.71%	96.90%	96.34%

Table 2. Parameter number and model size comparation.

Model	Total params	Bit depth	Size percentage
ResNet50	23M	32	10500%
MobileNet (α = 0.25)	219,058	32	100%
Our work	1,310	8/4	0.1%

3.2. Quantization performance

The overall accuracy evaluation is not sufficient to prove the effectiveness of the proposed quantization. In this section, the compression efficiency of our quantization method, traditional quantization method (full-4-bit) on our base model (full-8-bit) are compared. For each quantized model, quantization aware training is implemented. The compression efficiency is calculated by equation (3).

$$CE = \frac{1-\frac{Compressed\ model\ size}{Base\ model\ size}}{1-\frac{Compressed\ model\ accuracy}{Base\ model\ accuracy}} \quad (3)$$

Selecting full-8-bit quantization model as base model, the compression efficiency comparison is shown in Table 3. With full-4bits (traditional) quantization, 50% storage is saved by sacrificing 1.79% overall accuracy on validation set. With bit depth degression quantization, 33% storage is saved by sacrificing only 0.17% overall accuracy. In conclusion, bit Depth Degression Quantization performs 6.95 times compression efficiency of full-4-bit quantization on our network.

Table 3. Compression efficiency comparation.

	Train accuracy	Val accuracy	Size%	CE
Full-8-bit	97.03%	96.51%	100%	-
Full-4-bit	95.06%	94.42%	50%	26.95
Our work	96.90%	94.72%	67%	187.34

3.3. Hardware estimation

Target to Xilinx FPGA Kintex-7 XC7K325T, we consider estimating hardware resource with HLS as the first step towards CNN based defect detection with high frame rate and ultra-low delay architecture. The current hardware estimation is shown is Table 4. The processing delay is 0.672 ms per frame with instance image resolution 100*100, which achieved millisecond requirement.

According to the resource estimation, up to 6 classification network units are able to be mapped parallelly.

Table 4. Hardware estimation.

Resource utilization	#LUT	29014 (14%)
	#DSP48E	0 (0%)
	#BRAM	78 (8%)
Performance index	Clock period (ns)	8.59
	Throughput (cycles)	78242
	Processing delay (ms)	0.672

4. Conclusion

This paper proposes a deep learning based high frame rate ultra-low delay fruit defects detection solution. A critically compressed CNN with bit depth degression quantization is proposed as classification core and mapped on FPGA. It is able to work at 0.672 ms delay per frame, while keeps convincing accuracy.

Acknowledgment

This work was supported by KAKENHI (21K11816).

References

[1] S.Cubero et al.: "Automated Systems Based on Machine Vision for Inspecting Citrus Fruits from the Field to Postharvest—a Review," *Food Bioprocess Technol*, vol.9, pp.1623–1639,2016

[2] K.Nakano.: "Application of neural networks to the color grading of apples," *Comput. Electron. Agric.*, vol.18, pp.105-116, 1997

[3] R.Diaz et al.: "Comparison of three algorithms in the classification of table olives by means of computer vision," *J. Food Eng.*, vol.61, pp.101-107, 2004

[4] A.Costa et al.: "Computer vision based detection of external defects on tomatoes using deep learning," *Biosystems Engineering*, vol.190, pp.131-144, 2020

[5] H.Zhou et al.: "Defect Classification of Green Plums Based on Deep Learning," *Sensors*, vol.20, no.23, 2020

[6] S.Yadav et al.: "Identification of disease using deep learning and evaluation of bacteriosis in peach leaf," *Ecological Informatics*, vol.61, 2021

[7] T.Hu et al.: "FPGA implementation of high frame rate and ultra-low delay vision system with local and global parallel based matching," *2017 Fifteenth IAPR International Conference on Machine Vision Applications*, pp.286-289, 2017

[8] Y.Choukroun et al.: "Low-bit Quantization of Neural Networks for Efficient Inference," *2019 IEEE/CVF International Conference on Computer Vision Workshop*, pp.3009-3018, 2019

[9] G.Hinton et al.: "Distilling the knowledge in a neural network," *arXiv preprint,* arXiv:1503.02531, 2015

[10] S.Suzuki et al.: "Topological structural analysis of digitized binary images by border following. Computer Vision," *Graphics and Image Processing 1985*, vol.30, pp.32–46, 1985

[11] A.Howard et al.: "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint*, arXiv:1704.04861, 2017

[12] Y.Umuroglu et al.: "LogicNets: Co-Designed Neural Networks and Circuits for Extreme-Throughput Applications," *2020 30th International Conference on Field-Programmable Logic and Applications*, pp. 291-297, 2020

[13] B.Jacob et al.: "Quantization and training of neural networks for efficient integer-arithmetic-only inference," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.2704-2713, 2018

[14] H.Habi et al.: "HMQ: Hardware Friendly Mixed Precision Quantization Block for CNNs," *arXiv preprint,* arXiv:2007.09952, 2020