https://doi.org/10.2352/EI.2024.36.7.ISS-285

Adaptive bit depth control for neural network quantization

Youngil Seo, Dongpan Lim , Jungguk Lee, Seongwook Song; Samsung Electronics Ltd; Hwaseong/Korea

Abstract

Recently, many deep learning applications have been used on the mobile platform. To deploy them in the mobile platform, the networks should be quantized. The quantization of computer vision networks has been studied well but there have been few studies for the quantization of image restoration networks. In previous study, we studied the effect of the quantization of activations and weight for deep learning network on image quality following previous study for weight quantization for deep learning network.

In this paper, we made adaptive bit-depth control of input patch while maintaining the image quality similar to the floating point network to achieve more quantization bit reduction than previous work. Bit depth is controlled adaptive to the maximum pixel value of the input data block. It can preserve the linearity of the value in the block data so that the deep neural network doesn't need to be trained by the data distribution change.

With proposed method we could achieve 5 percent reduction in hardware area and power consumption for our custom deep network hardware while maintaining the image quality in subejctive and objective measurment. It is very important achievement for mobile platform hardware.

Introduction

Deep neural networks (DNNs) have become the state-of-the-art in the computer vision and sequence modeling problems like image classification, object detection, speech recognition. However, they usually suffer from high cost computation and memory costs with a huge amount of parameters. For example, Krizhevsky et al's research [1] and Simonyan et al's approach [2] show huge amount of parameters and deep layers. So it's very difficult to deploy deep networks on the mobile platforms that have limited power and computation resources.

This led to plentiful reserch that focus on model size and inference time of DNNs without degradation of performance. Approches in this researches consist of a few categories.

First, there are researches that design efficient architecture to exploit computation and memory like MobileNet, SqueezeNet, and DenseNet. There is also an approach like DPA Net [38] to make efficient network by taking image restoration algorithm analysis using distortion prior. Also DPA Net [38] tried to exploit the property of the priors.

Second, pruning, one of network compression method is the removal of irrelevant units (weights, neurons or convolutional filters)[5]. Network compression methods implicitly or explicitly aim at the systematic reduction of redundancy in neural network models while at the same time retaining a high level of task accuracy [4].

Lastly, quantization is the reduction of the bit-depth of weights or activations, which is particularly desirable from a hardware perspective[6].

Network quantization for vision applications like classification, image segmentation and object detection has drawn great attention of researchers [1] [2] [7] [8] [9]. Approaches for low-bit quantization of neural networks have been made for these applications. There are binary weight networks [10] [11] and ternary networks [12] [13] [14]. But owing to requirement of high bit-depth and high resolution there are few prior art on quantization of image restoration problems like demosaicing, super resolution and deblurring, etc. Seo et al [39] showed the effect of the weight quantization as the bit-depth changes.

In this paper, we made adaptive bit-depth control of input patch while maintaining the image quality similar to the floating point network to achieve more quantization bit reduction than previous work. Bit depth is controlled adaptive to the maximum pixel value of the input data block. It can preserve the linearity of the value in the block data so that the deep neural network doesn't need to be trained by the data distribution change.

With proposed method we could achieve 5 percent reduction in hardware area and power consumption for our custom deep network hardware while maintaining the image quality in subejctive and objective measurment. It is very important achievement for mobile platform hardware.

The bit depth control adaptive to the maximum input block data changes only precision. For low code data, full precision is used because human is more sensitive to value change at low code. For high code data, less precision is used, but subjective quality does not drop much. So in the point of subjective quality, ABC can maintain the quality in the most of cases.

Also maximum data checking algorithm is very simple. When MSB bits are checked then checking process is done. Overall added algorithm HW or code can be very small.

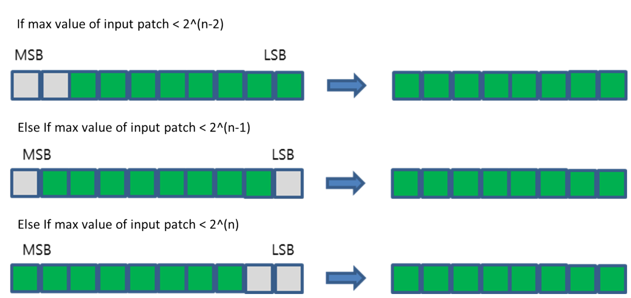

Figure 1. *Adaptive bit depth control : Bit depth is controlled adaptive to the maximum pixel value of the input data block.*

Releated works

In this work we focus mostly on quantization for demosaicing that is one of the image restoration and image signal processor, so we will brifly review related works.

Demosaicing of bayer color filter array has been extremely studied. [15], [16].There are various conventional approches, such as color difference based interpolation [17], [18], frequency domain filtering [19], [20], [21], and reconstruction methods [22], [23]. But for new other patterns, other effort like hand-crafted algorithms should be applied to solve it. So there is also universal approach [24].

Deep learning approaches to demosaicing has been applied [25], [26], [27], [28]. Previously, many researches focused on the bayer CFA demosaicing, but there are researches on Quad bayer pattern and Nona pattern demosaicing also [29], [30]. Deep learning methods have better image quality in complex CFA pattern demosaicing although they require high computation cost.

Especially we focus on RGBW CFA and its demosaicing. There are conventional algorithms like [33], [34] and deep learning approaches like [35], [39], [40]. Here our approach is related to deep learning RGBW demosaicing.

To deploy deep network on mobile platform, quantization is needed usually. There are two types of quantization methods. It is often desirable to reduce the model size by quantizing weights and activations post-training, without the need to re-train/fine-tune the model. These methods, commonly referred to as post-training quantization, are simple to use and allow for quantization with limited data [31].Quantization-aware training simulates quantization during training so that the quantization parameters can be learned together with the model using training data [32].

Problem statement

Deplyment on a mobile platform such as mobile phone requires quantization of network. There have been many studies on quantization for the DNN of vision processing like classification, segmentation, face detection and so on. But there are few studies on quantization for DNN of image restoration like demosaicing, denoising, deblur and super resolution. Conventional AI platform or AI hardware support just fixed bits like 8 bit or 16 bit integer operations and activations, but for customized AI hardware, bit reduction is directly connected to the reduction of hardware area and power.

In this paper, we made adaptive bit-depth control of input patch while maintaining the image quality similar to the floating point network to achieve more quantization bit reduction than previous work. Bit depth is controlled adaptive to the maximum pixel value of the input data block. It can preserve the linearity of the value in the block data so that the deep neural network doesn't need to be trained by the data distribution change.

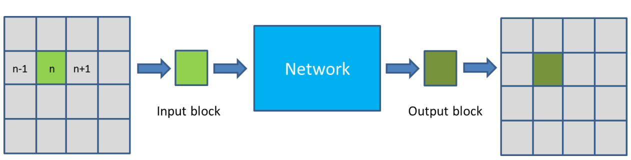

Figure 2. *For processing at mobile platform, block processing is assumed.*

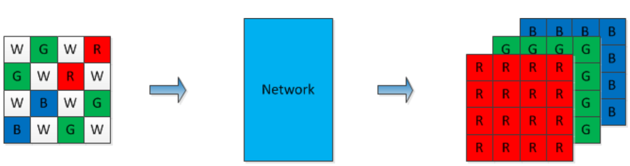

Figure 3. *RGBW demosaicing with deep learning network*

Proposed method

In this paper, we made adaptive bit-depth control of input patch while maintaining the image quality similar to the floating point network to achieve more quantization bit reduction than previous work. Bit depth is controlled adaptive to the maximum pixel value of the input data block. It can preserve the linearity of the value in the block data so that the deep neural network doesn't need to be trained by the data distribution change.

Basic idea is shown in Fig. 1, And for mobile platform processing the block data processing is assumed because the memory is limited in that kind of platform shown in Fig. 2. After processing in neural network system, then the data should be recovered to the original bit so that the inverse process is applied to the result data and the dihtering is used.

Here the bit depth control adaptive to the maximum input block data changes only precision. For low code data, full precision is used because human is more sensitive to value change at low code. For high code data, less precision is used, but subjective quality does not drop much. So in the point of subjective quality, ABC can maintain the quality in the most of cases.

Also maximum data checking algorithm is very simple. When MSB bits are checked then checking process is done. Overall added algorithm HW or code can be very small.

Like the preivous work we had experiments to find which bit is most adequate for the quantization of image restoration network.There are two quantizations in the network showed in Fig. 4, one is weight quantization and the other one is feature map (activation) quantization. Here we set the weight bit as 8 bit and watched how bit-depth of quantization for activations affects image quality. Here we used only post training quantization to see the direct effect of quantization on image quality.

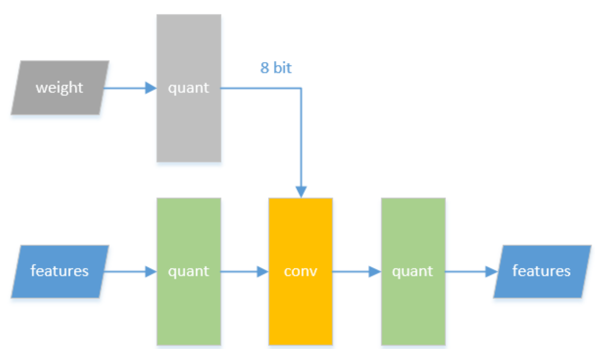

Figure 4. *Quantization in deep network*

We used Tensorflow as a base quantization tool and our quantized model architecture is based on their quantization network architecture. But to design a custom deep network hardware, we proposed our noble approach to reduce hardware area and power without degradation of image quality. First one is in our hardware we applied the adequate bit-depth in the feature map and second one is we used layer folding approach to fold quantization layers and prelu layer. Our folding approach can be used with other relu-like activations also.

$$\mathbf{r} = \mathbf{S}(\mathbf{q} - \mathbf{Z}) \qquad (1)$$

where $\mathbf{r}$ - real number, $\mathbf{q}$ - quantized number, $\mathbf{S}$ - scaling factor, $\mathbf{Z}$ - zero point. The basic quantization scheme is the affine mapping of integer **q** to real number **r**. In our approach to make hardware simple and reduce hardware size, we used symmetric quantization so that **Z** is zero.

$$\mathbf{S_3 q_3^{(i,k)}} = \sum_{\mathbf{j=1}}^{\mathbf{N}} \mathbf{S_1 q_1^{(i,j)} S_2 q_2^{(j,k)}} \tag{2}$$

$$\mathbf{q_3^{(i,k)}} = \mathbf{M} \sum_{\mathbf{j=1}}^{\mathbf{N}} \mathbf{q_1^{(i,j)} q_2^{(j,k)}} \tag{3}$$

Quantization of convolution can be written as the above equation. And **M** is requantization scaling factor.

$$\mathbf{M} = \frac{\mathbf{S_1 S_2}}{\mathbf{S_3}} = \frac{\mathbf{S_w S_i}}{\mathbf{S_o}} \tag{4}$$

where $\mathbf{S_w}$ is scaling of weight, $\mathbf{S_i}$ is scaling of convolution input and $\mathbf{S_o}$ is scaling of convolution output. This is the scaling term to calculate quantized integer output. There are two quantization layers before and after prelu layer. To fold quantization layer and prelu layer, we changed $\mathbf{S_o}$ as output scaling of prelu instead of convolution output scaling. For positive output we used this as it is while for negative output, it is multiplied by α like below.

$$\mathbf{M} = \alpha \frac{\mathbf{S_w S_i}}{\mathbf{S_o}} \tag{5}$$

In Fig. 6 left network diagram shows original quantized network and right diagram is folded quantization and prelu layers.

In this work like 3D graphics architecture testing environment(GATE) [3] that models graphics hardwrae architecture, we also implemented the network inference environment that models custom network inference hardware.

And we used our own network called DePhaseNet that we proposed in the previous research [40]. Its features are multi-level network with multi-phase inputs to adopt various phase schemes and correlations.

Experimental results

We made experiments by preparing pairs of RGBW CFA pattern images and ground truth RGB images. The network was trained on MIT dataset and HDR+ Burst Photography Dataset [36] seperately. We measured our algorithm on Kodak dataset [37] and real RGBW-K (kodak) image. We examined the effect of activation quantization bit depth for original 10 bit, LSB 2 bit truncated 8 bit and adaptive bit depth control applied 8 bit seperately.

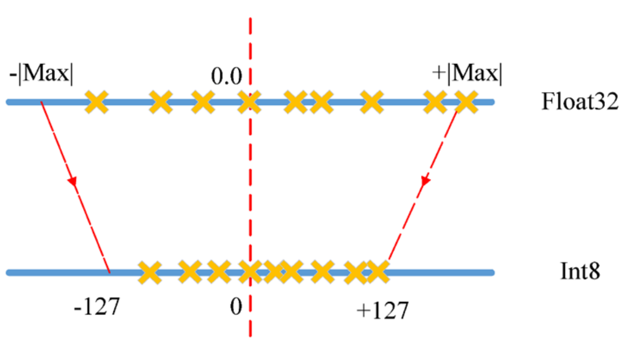

Figure 5. *Symmetric quantization*

In Table. 1 and Fig. 8, objective image quality evaluation results on various bits for 10 bit RGBW input are provided. For activation bit-depth 10 bit and 9 bit, ABC 8 bit shows best score and with activation 9 bit, the quality is almost similar to those for activation 12 bit.

Results for quantization in HDR+ dataset, PSNR [dB] for original 10 bit, LSB truncated 8 bit and ABC 8 bit

bit	10 bit	truncated 8 bit	ABC 8 bit
A12	42.454	42.204	42.309
A11	42.25	42.049	42.292
A10	33.038	41.623	42.125
A9	22.329	40.109	41.741

Subjective evaluation of experimental results show that we could see more quantization noises are shown in lower bit depth.

In kodak dataset, 9 bit is optimal and there is difference between lower bit and 9 bit, but there's no noticeable difference between 9 bit and float. Image results are shown in Fig. 9 and Fig. 10 for previous work.

And we made tests on 10 bit real RGBW-K raw images, and we could see clear advances of ABC in Fig. 11. For quantized network output results on RGBW image for activation 9 bit ABC

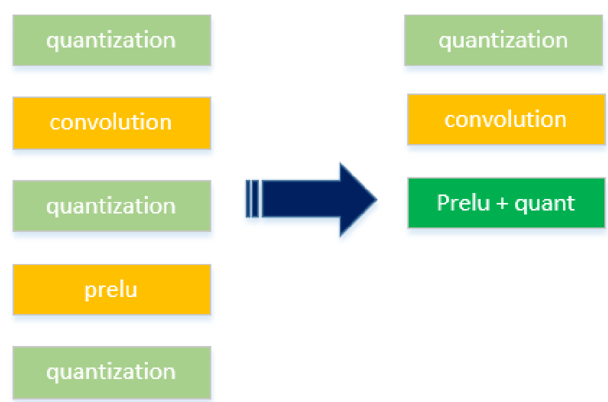

Figure 6. *left one is original quantized network and right one is prelu and quantization layers are folded*

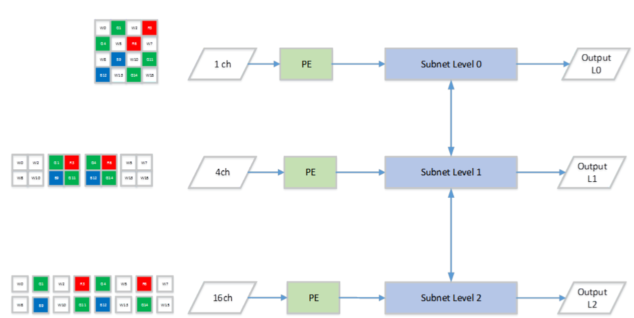

Figure 7. *DePhaseNet*

8 bit has much better. quality compared to the original 10 bit.

With proposed method we could achieve 5 percent reduction in hardware area and power consumption for our custom deep network hardware.

Conclusion

In this paper, we made adaptive bit-depth control of input patch while maintaining the image quality similar to the floating point network to achieve more quantization bit reduction than previous work. Bit depth is controlled adaptive to the maximum pixel value of the input data block. It can preserve the linearity of the value in the block data so that the deep neural network doesn't need to be trained by the data distribution change.

With proposed method we could achieve 5 percent reduction in hardware area and power consumption for our custom deep network hardware while maintaining the image quality in subejctive and objective measurment. It is very important achievement for mobile platform hardware.

Our algorithm can be applied so that HW area and power can be reduced, but also it can be applied to SW platform and the overall power consumption can be reduced owing to reduction of processing bit depth.

Our noble approach reduced hardware area and power consumption without degradation of image quality in subjective and in objective criteria. So that it is essential in design of custom deep network hardware and software platform.

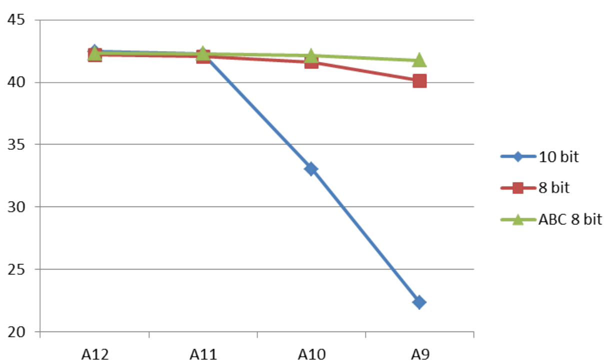

Figure 8. *Results for quantization in HDR+ dataset, PSNR [dB] for original 10 bit, LSB truncated 8 bit and ABC 8 bit*

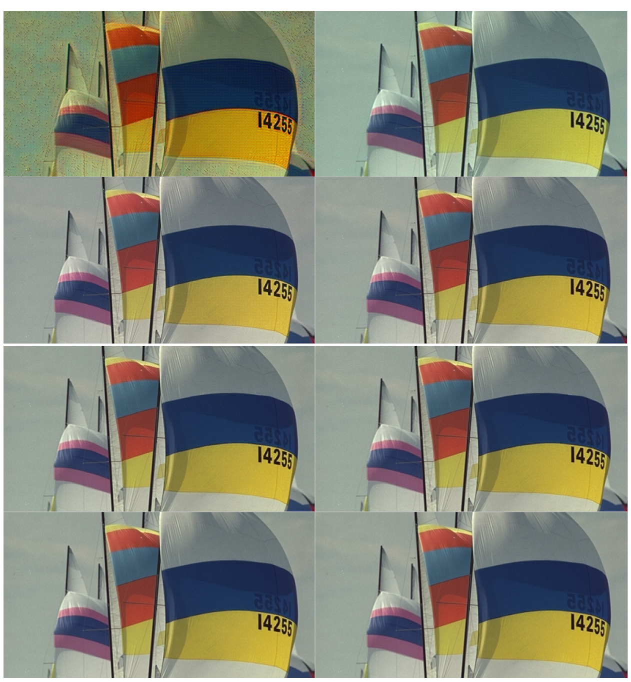

Figure 9. *Quantized network output results on Kodak image number 9: (a) - 6 bit; (b) - 7 bit; (c) - 8 bit; (d) - 9 bit; (e) - 10 bit; (f) - 11 bit; (g) - 16 bit; (h) - float.*

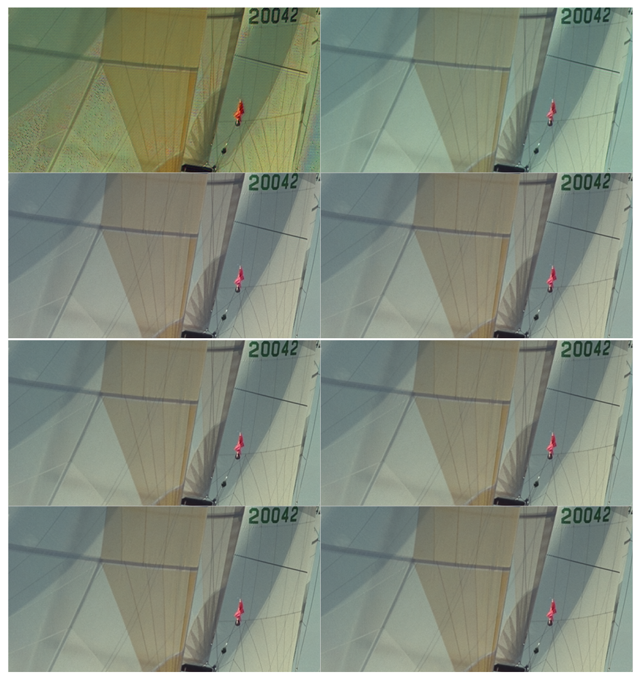

Figure 10. *Quantized network output results on Kodak image number 10: (a) - 6 bit; (b) - 7 bit; (c) - 8 bit; (d) - 9 bit; (e) - 10 bit; (f) - 11 bit; (g) - 16 bit; (h) - float.*

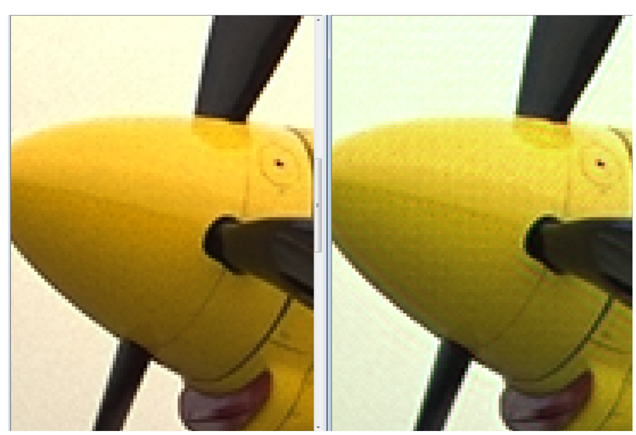

Figure 11. *Quantized network output results on RGBW image for activation 9 bit : (a) - ABC 8 bit; (b) - original 10 bit*

References

[1] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." Advances in neural information processing systems 25 (2012): 1097-1105.

[2] Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv:1409.1556 (2014).

[3] Lee, Inho, et al. "A hardware-like high-level language based environment for 3D graphics architecture exploration." 2003 IEEE International Symposium on Circuits and Systems (ISCAS). Vol. 2. IEEE, 2003.

[4] Achterhold, Jan, et al. "Variational network quantization." International Conference on Learning Representations. 2018.

[5] LeCun, Yann, John S. Denker, and Sara A. Solla. "Optimal brain damage." Advances in neural information processing systems. 1990.

[6] Sze, Vivienne, et al. "Efficient processing of deep neural networks: A tutorial and survey." Proceedings of the IEEE 105.12 (2017): 2295-2329.

[7] Wang, Peisong, et al. "Towards accurate post-training network quantization via bit-split and stitching." International Conference on Machine Learning. PMLR, 2020.

[8] Gong, Yunchao, et al. "Compressing deep convolutional networks using vector quantization." arXiv preprint arXiv:1412.6115 (2014).

[9] Yang, Jiwei, et al. "Quantization networks." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019.

[10] Courbariaux, Matthieu, Yoshua Bengio, and Jean-Pierre David. "Binaryconnect: Training deep neural networks with binary weights during propagations." Advances in neural information processing systems. 2015.

[11] Rastegari, Mohammad, et al. "Xnor-net: Imagenet classification using binary convolutional neural networks." European conference on computer vision. Springer, Cham, 2016.

[12] Li, Fengfu, Bo Zhang, and Bin Liu. "Ternary weight networks." arXiv preprint arXiv:1605.04711 (2016).

[13] Zhu, Chenzhuo, et al. "Trained ternary quantization." arXiv preprint arXiv:1612.01064 (2016).

[14] Zhou, Shuchang, et al. "Dorefa-net: Training low bitwidth convolutional neural networks with low bitwidth gradients." arXiv preprint arXiv:1606.06160 (2016).

[15] Li, Xin, Bahadir Gunturk, and Lei Zhang. "Image demosaicing: A systematic survey." Visual Communications and Image Processing 2008. Vol. 6822. International Society for Optics and Photonics, 2008.

[16] Menon, Daniele, and Giancarlo Calvagno. "Color image demosaicking: An overview." Signal Processing: Image Communication 26.8-9 (2011): 518-533.

[17] Cok, David R. "Signal processing method and apparatus for producing interpolated chrominance values in a sampled color image signal." U.S. Patent No. 4,642,678. 10 Feb. 1987.

[18] Adams Jr, James E. "Interactions between color plane interpolation and other image processing functions in electronic photography." Cameras and Systems for Electronic Photography and Scientific Imaging. Vol. 2416. International Society for Optics and Photonics, 1995.

[19] Adams Jr, James E. "Interactions between color plane interpolation and other image processing functions in electronic photography." Cameras and Systems for Electronic Photography and Scientific Imaging. Vol. 2416. International Society for Optics and Photonics, 1995.

[20] Dubois, Eric. "Frequency-domain methods for demosaicking of Bayer-sampled color images." IEEE Signal Processing Letters 12.12 (2005): 847-850.

[21] Hao, Pengwei, et al. "A geometric method for optimal design of color filter arrays." IEEE Transactions on Image Processing 20.3 (2010): 709-722.

[22] Mukherjee, Jayanta, R. Parthasarathi, and Sachin Goyal. "Markov random field processing for color demosaicing." Pattern Recognition Letters 22.3-4 (2001): 339-351.

[23] Keren, Daniel, and Margarita Osadchy. "Restoring subsampled color images." Machine Vision and applications 11.4 (1999): 197-202.

[24] Zhang, Chao, et al. "Universal demosaicking of color filter arrays." IEEE Transactions on Image Processing 25.11 (2016): 5173-5186.

[25] Gharbi, Michaël, et al. "Deep joint demosaicking and denoising." ACM Transactions on Graphics (ToG) 35.6 (2016): 1-12.

[26] Tan, Runjie, et al. "Color image demosaicking via deep residual learning." IEEE Int. Conf. Multimedia and Expo (ICME). Vol. 2. No. 4. 2017.

[27] Tan, Daniel Stanley, Wei-Yang Chen, and Kai-Lung Hua. "DeepDemosaicking: Adaptive image demosaicking via multiple deep fully convolutional networks." IEEE Transactions on Image Processing 27.5 (2018): 2408-2419.

[28] Syu, Nai-Sheng, Yu-Sheng Chen, and Yung-Yu Chuang. "Learning deep convolutional networks for demosaicing." arXiv preprint arXiv:1802.03769 (2018).

[29] Kim, Irina, et al. "On recent results in demosaicing of Samsung 108MP CMOS sensor using deep learning." 2021 IEEE Region 10 Symposium (TENSYMP). IEEE, 2021.

[30] Kim, Irina, et al. "Under display camera quad bayer raw image restoration using deep learning." Electronic Imaging 2021.7 (2021): 67-1.

[31] Banner, Ron, et al. "Post-training 4-bit quantization of convolution networks for rapid-deployment." arXiv preprint arXiv:1810.05723 (2018).

[32] Jacob, Benoit, 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. 2018.

[33] Chung, Kuo-Liang, Tzu-Hsien Chan, and Szu-Ni Chen. "Effective three-stage demosaicking method for RGBW CFA images using the iterative error-compensation based approach." Sensors 20.14 (2020): 3908.

[34] Kwan, Chiman, and Jude Larkin. "Demosaicing of bayer and CFA 2.0 patterns for low lighting images." Electronics 8.12 (2019): 1444.

[35] Kwan, Chiman, and Bryan Chou. "Further improvement of debayering performance of RGBW color filter arrays using deep learning and pansharpening techniques." Journal of Imaging 5.8 (2019): 68.

[36] Hasinoff, Samuel W., et al. "Burst photography for high dynamic range and low-light imaging on mobile cameras." ACM Transactions on Graphics (ToG) 35.6 (2016): 1-12.

[37] Loui, Alexander, et al. "Kodak's consumer video benchmark data set: concept definition and annotation." Proceedings of the international workshop on Workshop on multimedia information retrieval. 2007.

[38] Kim Irina, Lim Dongpan, Seo Youngil, Lee Jeongguk, Choi Wooseok and Song Seongwook. "Image Deblurring Using Deep Multi-Scale Distortion Prior." 2022 IEEE International Conference

on Image Processing (ICIP). IEEE, 2022.

[39] Seo, Youngil, et al. ”On quantization of convolutional neural networks for image restoration.” Electronic Imaging 34 (2022): 1-5.

[40] Kim, Irina, et al. ”DePhaseNet: A deep convolutional network using phase differentiated layers and frequency based custom loss for RGBW image sensor demosaicing.” Electronic Imaging 34 (2022): 1-5.

Author Biography

Youngil Seo received his B.S in Electrical Engineering from Hanyang University and M.S. in Electrical Engineering and Computer Science from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, in 2001 and 2003, respectively. From 2001 to 2009, he developed telemetics system in LG Electronics. Since 2009, he has been with Samsung Electronics where he developed Video codec, GPU, Sensor IP and so on. His main research interests include image processing systems and deep learning now.

Seongwook Song received his B.S. and M.S. degrees in electrical engineering from Seoul National University, in 1997 and 1999, respectively, and the Ph.D. degree in electrical and computer engineering from the University of Illinois at Urbana–Champaign, in 2004. He has been with Samsung Electronics since 2003, to develop 2G, 3G and 4G chipsets. His main research interests include advanced signal processing for digital communications, multimedia and deep learning systems for digital cameras.