Towards Eight-bit Quantization for 3D U-Net Medical Image Segmentation via ROI-Based Calibration and Background-Aware Shift

Soosung Kim
Department of Electrical and Computer Engineering
Chungang University
Seoul, Korea
kimsoosung@cau.ac.kr

Mincheol Cha
Department of Electrical and Computer Engineering
Seoul National University
Seoul, Korea
chamj61047@snu.ac.kr

Chaehag Yi
Department of Next Generation Semiconductor Convergence and Open Sharing System, Seoul National University
Seoul, Korea
chaehag@snu.ac.kr

Xuan Truong Nguyen
Department of Next Generation Semiconductor Convergence and Open Sharing System, Seoul National University
Seoul, Korea
truongnx@snu.ac.kr

***Abstract*—3D U-Net medical image segmentation is essential for precise diagnosis and treatment planning but comes with massive bandwidth requirements and considerable computational costs. Eight-bit fixed-point quantization is crucial for efficient 3D U-Net inference on modern deep learning accelerators. However, quantizing a 3D U-Net model is non-trivial and frequently leads to unfavorable accuracy degradation. We empirically observed that large range imbalances between tumor/non-tumor voxels and foreground/background areas are the main contributing factors. Based on these findings, we propose an eight-bit quantization method with a simple yet effective ROI-based calibration and a background shift that captures outliers in tumor areas to avoid accuracy degradation. Moreover, our method incorporates instance normalization folding, which eliminates computation-intensive operations during inference. Experimental results on the KITS19 dataset show that the proposed method with integer-only arithmetic achieves a negligible accuracy drop, i.e., 0.4%p in the dice score, compared to the FP32 model.**

***Index Terms*—3D U-Net Kidney/Tumor Image Segmentation, Eight-bit Quantization, Data Calibration, InstanceNorm Folding**

I. Introduction

Kidney cancer affects over 400,000 people each year, creating a critical need for surgical approaches that are customized to the complex and varied shapes of each tumor. Given the effectiveness of convolutional neural networks, 3D U-Net models have become widely adopted for segmenting and classifying tissues into background, kidney, and tumor regions [1], [6]. Unfortunately, a 3D U-Net frequently requires massive memory bandwidth and a large computational cost. To tackle these challenges, eight-bit fixed-point quantization - which is supported by modern deep learning hardware [2], [9]–[12] - is crucial for efficient 3D U-Net inference. 3D CT images, however, consist of a highly dynamic range variance among tumor and tumorless voxels and between background and foreground areas, which frequently leads to significant accuracy degradation in quantization. To address this problem, this work proposes an effective eight-bit quantization for 3D U-Net models with the following contributions:

- Hardware-friendly design: To facilitate model deployment on hardware, our design targets integer-only arithmetic with an inter-layer binary scaling scheme. The target design also supports both int8 and uint8.
- Accuracy: We empirically observed that tumor areas are more sensitive for accuracy. Therefore, our quantization searches for an ROI-based range to avoid an accuracy drop. Additionally, our method processes a background-aware shift, which enhances accuracy and increases sparsity simultaneously.
- Instance Normalization Folding: To avoid computational-expensive normalization during inference, we adopt an instance normalization folding. Similar to batch normalization, running means and variances among voxels are learned during training and fused with weights and biases before inference.

II. Background

A. 3D Image Patches and Voxels

During training and inference, 3D images are usually divided into patches or voxels, i.e., sized (128, 128, 128) in size, which are fed into a network. The results are combined after each patch is processed.

B. Instance Normalization Folding

During training, mean and variance parameters are learned and normalized among different patches similar to batch normalization [3]. To reduce computational complexity and

memory usage during inference, instance normalization folding fuses these parameters into weights and biases prior to model inference.

C. Post-Training Quantization and Scales

In typical post-training quantization (PTQ) methods utilizing two scale factors for per-layer scaling [4], [9]–[11], the input and weights of a convolutional layer are each multiplied by their respective scale factors $\text{Scale}_{\text{Act}}^{(n)}$ and $\text{Scale}_{\text{Weight}}^{(n)}$. Subsequently, at the end of the layer, the output is divided by the product of these scale factors. When a bias term is present, the scale factor applied to the bias is $\text{Scale}_{\text{Act}}^{(n)} \times \text{Scale}_{\text{Weight}}^{(n)}$.

III. Proposed Method

A. Hardware-Friendly Quantization

1) Binary Scaling: In hardware implementations, utilizing values that are powers of two offers significant advantages due to the reduction in computational cost, as multiplication operations can be efficiently replaced by shift operations. Contrary to the approach in [2], where scaling factors are set to powers of two, our method refines the quantization process by ensuring that the ratio of scaling factors between consecutive layers is a power of two. Specifically, we set the ratio

$$\text{Scale}_{\text{Act}}^{(n+1)} / (\text{Scale}_{\text{Act}}^{(n)} \times \text{Scale}_{\text{weight}}^{(n)})$$

to be a power of two. This strategy simplifies the implementation of quantization by facilitating the use of efficient shift operations within the processing element (PE).

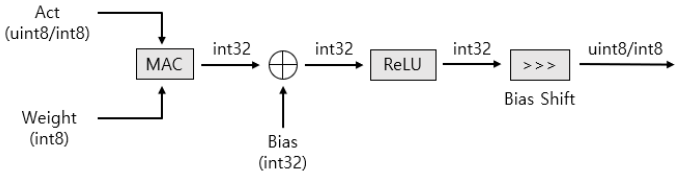


Fig. 1: Proposed Quantization Scheme

2) Adaptive Quantization: When passing through ReLU, all negative values are clipped to zero, allowing both the input and the output of convolutional layers to be represented as uint8. However, since the convtranspose layer does not include ReLU, its output must be received as int8, and consequently, the subsequent convolutional layer's input should also be int8. This approach of using different data types for each layer enhances granularity compared to the method of uniformly quantizing both weights and activations into one data type for all layers [1].

B. Range Optimization

1) Calibration sample: In post-training quantization (PTQ) approaches, the significance of calibration data is paramount as it directly influences the quantization scale factors and consequently, the overall precision of the model as noted in studies like [5] [6] [7]. In this case, by examining the characteristics of patches containing the region of interest (ROI), namely tumors, as opposed to those that do not, we can determine which patches should be selected as calibration samples. Specifically, a common feature among patches containing tumors was that they exhibited higher voxel averages. Consequently, patches with the highest average values could be considered most representative for calibration purposes.

2) Sparsity-Aware Quantization: In the case of 3D CT images, the proportion of voxels corresponding to the background is significantly large. Typically, background values are measured as very small according to the Hounsfield scale. During preprocessing, these background values are clipped to -2.3407. Consequently, by adding 2.3407 to all preprocessed CT voxel values, many voxels are effectively shifted to zero, thus increasing sparsity. This implies that zero-skipping techniques can be leveraged in subsequent hardware design.

Through the use of bias correction techniques, such as those used by [8] to address quantization errors, IFM sparsity for the first layer can be achieved without performance degradation. ($\alpha = 2.3407$)

$$y = w_{fused} \cdot (\text{input} + \alpha) + b_{fused} - \frac{w_{conv} \cdot \alpha}{\sqrt{\text{Var}[x] + \epsilon}} \cdot \gamma$$

C. Instance Normalization Folding

To accommodate the hardware real-time inference speed requirements, we trained the model with the running mean and the running variance activated. As a result, there was a slight trade-off in accuracy. To mitigate the reduction in quantization accuracy, we implemented the strategy of decreasing the number of quantization steps within each layer. Layer Fusion is a technique that integrates Instance Normalization operations into the weights and biases of a CONV layer. The fused weights and biases are computed using the following formulas.

$$w_{fused} = w_{conv} \cdot \frac{\gamma}{\sqrt{Var[x] + \epsilon}}$$

$$b_{fused} = \frac{b_{conv} - E[x]}{\sqrt{Var[x] + \epsilon}} \cdot \gamma + \beta$$

Here, w_{conv} and b_{conv} represent the weight and bias of a CONV layer, while γ and β represent the weight and bias of InstanceNorm layer. E[x] and Var[x] are defined as the running mean and variance calculated on an instance during training. In hardware, the fused layer reduces latency by using just one multiplication and one addition.

IV. Experimental Results

A. Determination of a calibration sample

As illustrated in Fig. 2, two observations can be made: 1) patches containing tumors, represented by red, generally exhibit higher average voxel values, and 2) looking into the red dots, a correlation can be found between the tumor percentage and the average voxel values per patch. These findings suggest that these characteristics are typical of tumors. Consequently, the patches within the circled area, showing the

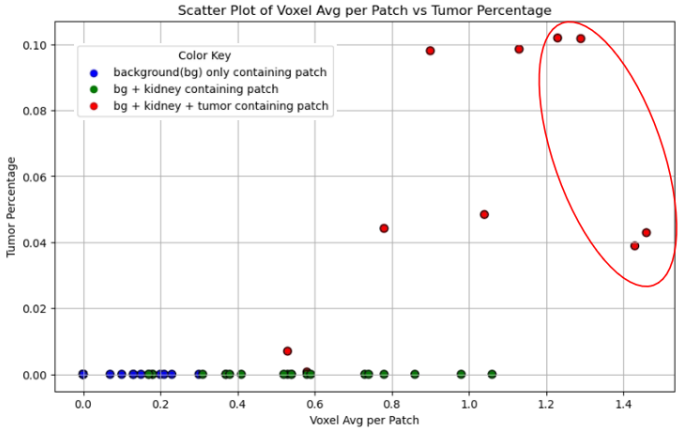


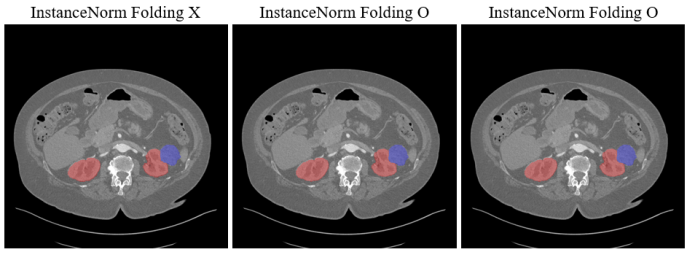
Fig. 2: Scatter Plot of Voxel Avg per Patch vs. Tumor Percentage for a training image containing 50 patches

highest average voxel values and tumor percentages, can serve as a representative calibration sample for tumor segmentation.

B. Accuracy Results

The accuracy results are from using the top four circled patches as the calibration set for quantization. As illustrated in Fig. 3, a sample test case visually confirms that the quantization has been performed effectively.

InstanceNorm Folding	8-bit Quantization	Mean Dice Score
X	X	91.1
O	X	88.1
O	O	87.7



float32	**float32**	**int8**
kidney 0.972087	kidney 0.965635	kidney 0.965635
tumor 0.950772	tumor 0.935209	tumor 0.935209
mean_dice 0.9614295	**mean_dice 0.950422**	**mean_dice 0.950422**

Fig. 3: Visualization of kidney tumor segmentation for 1 random test dataset

V. Conclusion

We present an optimized quantization process for 3D U-Net models used in kidney tumor segmentation, introducing an effective ROI-based calibration and a background-aware shift. This approach minimized accuracy degradation to a negligible 0.4%p drop in the dice score, and significantly enhanced the efficiency of the hardware implementation. Our results on the KiTS19 dataset validate the practicality of our methods. Future research will aim to extend these techniques to other medical imaging tasks and evaluate their effectiveness in clinical settings.

VI. Acknowledgment

This work was supported in part by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2024-2020-0-01461) supervised by the IITP(Institute for Information & communications Technology Planning & Evaluation) and in part by the Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. RS-2024-00399936, Development of a Benchmark Test (BMT) Platform Technology for Edge AI Semiconductor). This research was also supported by the Next Generation Semiconductor Convergence and Open Sharing System (COSS) program.

References

[1] Jianwei Li, Tianchi Zhang, Ian En-Hsu Ten, Dongkuan Xu, "FP8-BERT: Post-Training Quantization for Transformer," arXiv:2312.05725 [cs.AI]

[2] Minsik Kim, Kyoungseok Oh, Youngmock Cho, Hojin Seo, Xuan Truong Nguyen, and Hyuk-Jae Lee, "A Low-Latency FPGA Accelerator for YOLOv3-Tiny With Flexible Layerwise Mapping and Dataflow," IEEE Transactions on Circuits and Systems I: Regular Papers, Volume: 71, Issue: 3, March. 2024, pp. 1158-1171

[3] Edouard Yvinec, Arnaud Dapogny, Kevin Bailly, "To Fold or Not to Fold: a Necessary and Sufficient Condition on Batch-Normalization Layers Folding," arXiv:2203.14646

[4] Dai, Steve, et al. "Vs-quant: Per-vector scaled quantization for accurate low-precision neural network inference." Proceedings of Machine Learning and Systems 3 (2021): 873-884.

[5] Zhang, Rongzhao, and Albert CS Chung. "EfficientQ: An efficient and accurate post-training neural network quantization method for medical image segmentation." Medical Image Analysis 97 (2024): 103277.

[6] Williams, Miles, and Nikolaos Aletras. "On the Impact of Calibration Data in Post-training Quantization and Pruning." Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2024.

[7] Hubara, Itay, et al. "Accurate post training quantization with small calibration sets." International Conference on Machine Learning. PMLR, 2021.

[8] Nagel, Markus, et al. "Data-free quantization through weight equalization and bias correction." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019.

[9] Kyeongjong Lim, Gyuri Kim, Taehyung Park, Xuan Truong Nguyen, and Hyuk-Jae Lee, "An Energy-Efficient YOLO Accelerator Optimizing Filter Switching Activity," IEEE International Symposium on Circuits and Systems (ISCAS), May 2022.

[10] Subin Ki*, Juntae Park*, and Hyun Kim, "Dedicated FPGA Implementation of the Gaussian TinyYOLOv3 Accelerator," IEEE Transactions on Circuits and Systems II: Express Briefs, Volume: 70, Issue: 10, Pages:3882-3886, Oct. 2023.

[11] Jicheon Kim, Chunmyung Park, Eunjae Hyun, Xuan Truong Nguyen, and Hyuk-Jae Lee, "A Scalable Multi-Chip YOLO Accelerator With a Lightweight Inter-Chip Adapter," IEEE International Symposium on Circuits and Systems (ISCAS), May. 2024.

[12] Jicheon Kim, Chunmyung Park, Eunjae Hyun, Xuan Truong Nguyen, and Hyuk-Jae Lee, "A Highly-Scalable Deep-Learning Accelerator With a Cost-Effective Chip-to-Chip Adapter and a C2C-Communication-Aware Scheduler," IEEE Journal on Emerging and Selected Topics in Circuits and Systems (JETCAS), Sep. 2024.

[13] Hu Cao, Yueyue Wang, Joy Chen, Dongsheng Jiang, Xiaopeng Zhang, Qi Tian, Manning Wang, "Swin-Unet: Unet-Like Pure Transformer for Medical Image Segmentation," Computer Vision – ECCV 2022 Workshops: Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part III, Pages 205 - 218.