Article Information

Title	Efficient and Fault-tolerant Object Localization and Classification Based on an Ensemble of Dual Ternary YOLIC Models
Authors	Masahiro Ishii, Kai Su, Yoichi Tomioka, Hiroshi Saito
Citation	2024 IEEE 17th International Symposium on Embedded Multicore/Many-core Systems-on-Chip (MCSoC), Kuala Lumpur, Malaysia, 2024, pp. 302-309, doi: 10.1109/MCSoC64144.2024.00057.
Copyright	
Note	This is the author's accepted version of the paper published in *Proceedings of IEEE International Symposium on Embedded Multicore/Many-core Systems-on-Chip (2024)*. The final version is available at IEEE Xplore DOI: https://doi.org/10.1109/MCSoC64144.2024.00061

Efficient and Fault-tolerant Object Localization and Classification Based on an Ensemble of Dual Ternary YOLIC Models

Masahiro Ishii, Kai Su, Yoichi Tomioka, Hiroshi Saito
Graduate School of Computer Science and Engineering
University of Aizu
Aizu-Wakamatsu City, Japan
{m5281017,d8232114,ytomioka,hiroshis}@u-aizu.ac.jp

***Abstract*—Misses in artificial intelligence inference due to hardware failures can cause serious accidents in mission-critical systems involving automated guided vehicles, delivery robots, and so on. It is desirable to realize a fault-tolerant AI system that continues to operate normally even when a failure occurs. Conventional fault-tolerant techniques have the problems of increased computational cost and power consumption. In this paper, we proposed an accurate and fault-tolerant object localization and classification based on an ensemble of dual modular redundancy of ternary You Only Look at Interested Cells (YOLIC) models. In our experiments, we evaluate the proposed method with two types of road surface risk detection datasets. We demonstrate that the fault-tolerant model consisting of three ternary (ternary-weight) models achieves comparable accuracy and reduces the computational cost by 89.5% (79.7%) compared to the triple modular redundancy (TMR) of the floating-point model of the same structure.**

***Index Terms*—fault-tolerant, quantization, ensemble learning, dual modular redundancy, neural network**

I. Introduction

In recent years, automated guided vehicles, delivery robots, and automatic driving based on neural networks have been developed actively. In addition, technologies that utilize the Internet of Things (IoT) with artificial intelligence (AI) are being incorporated into factory automation and transportation robots. In this situation, automatic recognition by AI plays an important and responsible role.

Using AI accelerators is one of the promising options in edge AI systems to meet the strict conditions of power consumption and real-time constraints. On the other hand, the accelerators can be affected by hardware failures in inference [1]. The safety and reliability-related properties of neural networks are widely questioned. As CMOS technology downscales, circuits become more sensitive to soft errors caused by atmospheric neutrons, radioactive impurities, voltage instability, and temperature fluctuations, which can lead to inference failures in edge AI devices [2]. Another possibility of failure is Bias temperature instability (BTI), which is a deterioration of MOS transistors due to aging. Reference [3] reported that BTI might cause non-negligible delayed degeneration. Such faults in mission-critical AI systems can lead to serious accidents that threaten our lives. There are demands for the realization of a fault-tolerant AI system that continues to operate normally even when a failure occurs.

A traditional fault-tolerant technique of placing three identical circuits, inputting the same data into them, and taking majority voting of their outputs is called triple modular redundancy (TMR). For example, In the aerospace field, TMR is used in flight control systems and avionics to detect faults and switch to normal operation [4]. In this approach, we execute the same process independently in three modules, such as AI accelerators, and compare the outputs to determine the correct result. The advantage of this approach is to provide correct results even if one accelerator outputs incorrect results. While this method provides high reliability, the problem is that redundancy increases the computational cost, area, and power consumption.

References [5] and [6] have proposed more efficient fault-tolerant neural network models that can contribute to the reduction of hardware resources and power consumption. Reference [5] has reduced the overhead to detect adversarial attacks to a neural network accelerator by applying the dual modular redundancy (DMR) technique only to neurons whose faults degenerate the inference accuracy significantly. Reference [6] has proposed a fault-tolerant ensemble network that applies a modular redundancy to a part of neural network layers to detect faulty accelerators and realize accurate inference based on an ensemble of neural network models run in non-faulty accelerators. Because these approaches do not realize modular redundancy for whole parts of neural networks, there is a risk of missing serious soft errors. Moreover, although the effectiveness of fault-tolerant techniques has been shown in the field of image classification, more advanced tasks, such as object localization, have not been sufficiently investigated.

In this paper, we propose a hardware-friendly, accurate, and fault-tolerant object localization and classification model that is based on a combination of quantization, ensemble model,

This work was supported by JST, PRESTO Grant Number JPMJPR22C7, Japan. This paper is partly based on results obtained from "Research and Development Project of the Enhanced Infrastructures for Post-5G Information and Communication Systems" (JPNP20017), commissioned by the New Energy and Industrial Technology Development Organization (NEDO).

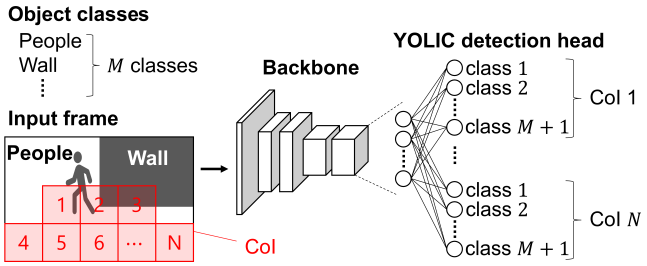


Fig. 1. The structure of a YOLIC model.

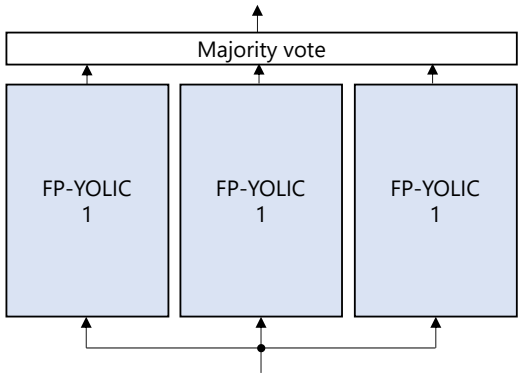


Fig. 2. Fault-tolerance of YOLIC models based on conventional TMR.

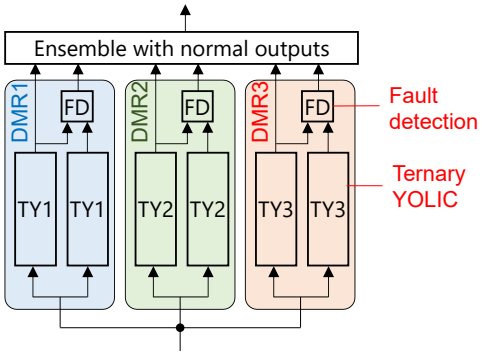


Fig. 3. Proposed fault-tolerant model consisting of three DMR units of ternary YOLIC models.

and dual modular redundancy. The contributions of this paper are as follows:

1) We propose a novel fault-tolerant object localization and classification model consisting of an ensemble of DMR of ternary You Only Look at Interested Cells (YOLIC) [7] models.
2) We demonstrate that our fault-tolerant model can achieve lower computational cost than the conventional TMR of a floating-point YOLIC model when no faults occur while achieving comparable accuracy.

The remainder of this paper is organized as follows. In Section II, we explain the YOLIC approach for object localization and classification. In Section III, we explain the proposed method to convert a YOLIC model to an efficient fault-tolerant YOLIC model. In Section IV, we show the experimental results with road risk detection dataset and road surface damage detection dataset. We conclude this paper in Section V.

II. You Only Look at Interested Cells (YOLIC)

YOLIC has been proposed in [7], which is an efficient method for object localization and classification on edge devices. YOLIC places cells called Cells of Interest (CoI) in the image and scrutinizes each CoI for the presence of object parts. In contrast to traditional object detection methods that require extensive searches of the entire image, YOLIC passively waits for objects to appear in predefined cells. This can reduce the computational cost for object localization and detection. By focusing on labeling at the necessary granularity, the labeling effort can be reduced compared to semantic segmentation. Reference [8] demonstrates that YOLIC achieves faster and more accurate object localization and classification compared with YOLOv5 [9] and YOLOv8 [10] for road risk detection.

As shown in Figure 1, a YOLIC model consists of a backbone and a YOLIC detection head. The backbone is based on convolutional neural networks and extracts features from a frame. The YOLIC detection head consists of fully-connected layers and decides if a risk of each class exists in each cell. The YOLIC model has $N \times (M+1)$ outputs; for each of N CoIs, there are $M+1$ neurons corresponding to M object classes and the background class. We employ a sigmoid activation function at the end of the YOLIC classification head.

In YOLIC, a binary cross-entropy loss function during the training phase is employed as:

$$C = N \times (M+1), \tag{1}$$

$$\text{Loss} = -\frac{1}{C}\sum_{i=1}^{N}\sum_{j=1}^{M+1}\left[y_{ij}\log\left(p_{ij}\right) + \left(1-y_{ij}\right)\log\left(1-p_{ij}\right)\right]. \tag{2}$$

where y_{ij} and p_{ij} denote the true label and predicted value for the j-th object class in the i-th CoI, respectively. The true label y_{ij} is set to 1 if an object of the j-th object class is overlapped with the i-th CoI. For example, in Figure 1, the true label of CoI 1 is set to 1 for the people class. The true labels of COI 2 are set to 1 for the people and wall classes. In the inference phase, we detect an object of the j-th object class if p_{ij} is greater than or equal to a specified threshold and the predicted value $p_{i(M+1)}$ for the background class is less than a specified threshold. That is, YOLIC can detect multiple classes for each CoI. In experiments, we used 0.5 as the threshold as in [8].

III. Fault-tolerant model based on ensemble of Ternary DMR

A. Structure of fault-tolerant YOLIC model

In this section, we explain an approach for transforming a full-precision YOLIC (FP-YOLIC) model into a fault-tolerant,

accurate, and efficient YOLIC model. Figure 2 represents a conventional TMR approach that takes the majority votes of the outputs of the three identical models. It increases the number of operations by more than three times while the accuracy is maintained. Ternary quantization is a promising approach for reducing the computational cost and miniaturizing the circuit for TMR. However, it can decrease the inference accuracy depending on the target problems. Typically, the more complex the problem, the more challenging aggressive quantization becomes.

On the other hand, Figure 3 represents the proposed fault-tolerant model for object localization and classification. It is an ensemble model consisting of multiple DMR units. Each DMR unit consists of two identical YOLIC models in which weight and activation are quantized. The weight is quantized into ternary values: -1, 0, and 1 while the activation is quantized into n-bit unsigned integer values or ternary values. Depending on the problems, we select the appropriate bit width for activation. Each DMR unit can detect if one of the two YOLIC models included in the DMR unit fails. We dynamically realize an ensemble of DMR units that do not detect a fault. Therefore, even if a failure occurs, we can maintain sufficiently high accuracy.

Let N_e be the number of DMR units in an ensemble model. Let y_{ij}^k be the predicted value of the k-th DMR unit for the j-th object class in the i-th CoI. For multi-class object localization and classification, we employ a majority votes of the outputs of N_e DMR units. If y_{ij}^k is greater than or equal to a specified threshold, the k-th DMR unit vote in the detection bin for the j-th object in the i-th CoI ($1 \leq k \leq N_e$). We detect an object of the j-th object class in the i-th CoI if the number of votes in the corresponding detection bin is greater than or equal to $N_e/2$. Note that N_e must be an odd number. For example, when one DMR unit detects a fault in an ensemble model consisting of five DMR units, we make an ensemble model consisting of three normal DMR units to maintain a high accuracy.

In the case of binary-class object localization and classification (i.e., $M = 1$), we change the activation function of each YOLIC model and the ensemble method to realize a more flexible ensemble model consisting of any number of models. Instead of the sigmoid activation function, we employ the softmax function at the end of each YOLIC model as follows.

$$p_{ij}^k = \frac{\exp(z_{ij}^k)}{\sum_{j\in\{1,2\}} \exp(z_{ij}^k)} \tag{3}$$

where z_{ij}^k is the logits of a YOLIC model of the k-th DMR unit. We employ a cross-entropy loss function calculated by

$$\text{Loss}_2 = \frac{1}{N} \sum_{i=1}^{N} \sum_{j\in\{1,2\}} y_{ij}^k \log p_{ij}^k. \tag{4}$$

In the inference phase, we calculate the average of outputs of N_e DMR units instead of majority votes. Note that we can accept ensemble models consisting of an even number of DMR units.

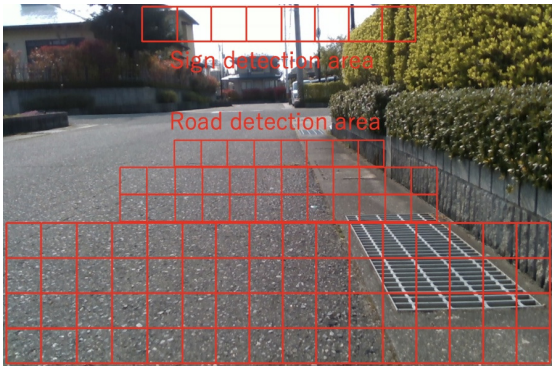


Fig. 4. Cell configuration for road risk detection.

B. Training

To realize the ternary YOLIC model, quantization-aware training [11] is employed. The neural network calculates ternary weight and quantized activation from the original floating-point weights and activations during training. Ternary weights and quantized activations are used in the forward path. In the backward path, we propagate gradients from outputs to inputs with a straight-through estimator (STE) [11] for the quantization layer; the gradient propagated to the output is passed to the input as it is. The original floating-point parameters are updated based on the obtained gradients.

We train and construct a fault-tolerant ensemble model as follows. First, we train a floating-point YOLIC model. Second, we train multiple ternary YOLIC models with different random seeds so that different data augmentations are applied. The pre-trained floating-point YOLIC model's parameters are used as initial parameters. Lastly, we construct a fault-tolerant ensemble model by combining DMR units of the obtained ternary YOLIC models.

IV. Experimental Results

A. Dataset and Models

In this paper, we select two types of object localization and classification problems: road risk detection and road surface damage detection. For road risk detection, we use the dataset provided in [8]. In this dataset, there are 11 classes: Column, Dent, Fence, People, Vehicle, Wall, Weed, ZebraCrossing, Traffic Cone, and Traffic Sign. In Figure 4, We show cell configuration for road risk detection defined in [8]. There is a traffic sign detection area at the top of the image, and 8 CoIs are used to approximate the position of the traffic sign. A total of 96 CoIs are assigned to the road detection area at the bottom of the image. Therefore, 104 cells are used for object detection.

Table I shows our YOLIC model that employs VGG11-BN [12] as a backbone. The input frame size is 224×224. The classification head of VGG11-BN is replaced with a series of global average pooling, a fully-connected layer with sigmoid activation. The parameters trained for IMAGENET are used

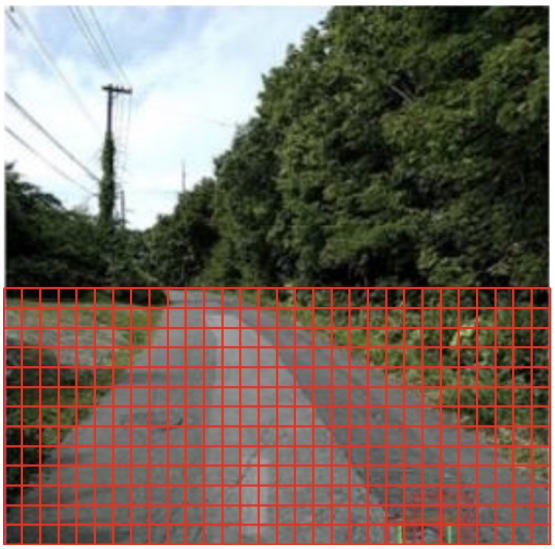


Fig. 5. Cells for road surface damage detection.

TABLE I
YOLIC NETWORK CONFIGURATION.

stage	layer	filter size	output size
stage1	Conv	(3,3)	(320,640,64)
	Batchnorm		(320,640,64)
	Maxpool	(2,2)	(160,320,64)
stage2	Conv	(3,3)	(160,320,128)
	Batchnorm		(160,320,128)
	Conv	(3,3)	(160,320,128)
	Maxpool	(2,2)	(80,160,128)
stage3	Conv	(3,3)	(80,160,256)
	Batchnorm		(80,160,256)
	Conv	(3,3)	(80,160,256)
	Maxpool	(2,2)	(40,80,256)
stage4	Conv	(3,3)	(40,80,512)
	Batchnorm		(40,80,512)
	Conv	(3,3)	(40,80,512)
	Maxpool	(2,2)	(20,40,512)
stage5	Conv	(3,3)	(20,40,512)
	Batchnorm		(10,20,2)

TABLE II
DEFINITION OF TP, FN, FP, AND TN.

		Predicted Class	
		Positive	Negative
Actual Class	Positive	True Positive (TP)	False Negative (FN)
	Negative	False Positive (FP)	True Negative (TN)

as initial parameters in training. We quantize the activation of ternary YOLIC model into 8 bits for road risk detection.

For road surface damage detection, we used the GRDDC2020 dataset [13]. There are eight classes for road damage, and our model is trained with a focus on the detection of the D40 class involving rutting bumps, and potholes that can be a risk to personal mobiles and delivery robots. Figure 5 shows an example of a simple and regular CoI configuration for road surface damage detection. We target the detection to the lower half of the image. We divide the lower half into 10×20 CoIs as shown in Figure 5. We also employed VGG11-BN [14] as the backbone of the YOLIC model. However, there are two changes. First, the input image size is set to 640×640. Second, the YOLIC detection head is replaced with a convolutional layer to avoid overfitting in training with GRDDC2020 dataset. The structure of the YOLIC model is shown in Table I. We quantize the activation of ternary YOLIC model into ternary values with 2 bits for road surface damage detection.

B. Environment and setting

In the experiments, we used a local runtime of Google Colaboratory and PyTorch 1.11.0 as a deep learning framework on a desktop PC with AMD Ryzen 5600X 6-Core Processor and NVIDIA GeForce RTX 3090 GPU. The optimizer for neural network training is set to Adam. The number of training epochs is set to 150 in road risk detection and 300 in road damage detection. The learning rate is set to 0.001. In road risk detection, data augmentation is performed by randomly changing the brightness, contrast, saturation, and hue of the image within a range of ±50%. In road damage detection, data augmentation is performed by randomly flipping and cropping images.

C. Evaluation indices

We evaluated the proposed method using two evaluation indices. The first is F1-score. F1-score represents the harmonic mean of precision and recall. The second is the number of operations with consideration of required bit width, which are denoted by BOPs. The number of BOPs is calculated by counting the multiplications and additions required for inference with consideration of bit width. We also evaluated how much computational cost can be reduced by the proposed method compared to the conventional TMR model.

True Positive (TP), False Negative (FN), False Positive (FP), and True Negative (TN) are defined as Table II. Recall, Precision, and F1-score can be calculated as follows:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{5}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{6}$$

$$\text{F1-score} = \frac{2\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{7}$$

Precision is the ratio of true positives to the sum of true positives and false positives. Recall is the ratio of true positives to the sum of true positives and false negatives. F1-score represents the harmonic mean of precision and recall.

We also evaluated BOPs [15] of each YOLIC model. BOPs stands for computational cost when a model infers. The activation and weight bits are used to calculate BOPs. To calculate an element of an output feature map, we require MNK^2 multiply-accumulate (MAC) operations where N is the number of input channels, M is the number of output

channels, and K is the filter size. Let B_w and B_a be the bit widths of weight and activation, respectively. The bit width of the weight and activation multiplication result is B_w and B_a, respectively. The maximum value of the accumulation result is about $2^{(B_w+B_a)}NK^2$. Therefore, the bit width of the accumulation result becomes $\log_2(2^{(B_a+B_w)}NK^2)$. The multiplier and adder of wider bit width require more hardware resources to be implemented. The estimation of BOPs assumes that the cost of multiplication and addition is proportionate to their bit widths. Therefore, the calculation cost of MAC is defined as $B_a + B_w + \log_2(NK^2)$.

To obtain the output feature map of $H \times W \times M$ by convolution of an input feature map of $H \times W \times N$ and M filters of $K \times K$ with padding, we require $HWMK^2N$ MAC operations. Therefore, The number of BOPs is calculated by

$$\text{BOPs} = HWMK^2N\left(B_aB_w + B_a + B_w + \log_2 NK^2\right) \quad (8)$$

D. Road Risk Detection

In accordance with the explanation provided in Section III, the proposed fault-tolerant ensemble model can detect a permanent fault of hardware and temporal fault by computation of DMR. Although quantization can reduce the total computational cost, that may reduce the accuracy of the model. In addition, the redundancy of DMR in the ensemble model may increase the computational cost. Therefore, it is necessary to evaluate the accuracy and the number of BOPs of the proposed fault-tolerant ensemble model. We evaluated the TMR of the original FP-YOLIC models, the TMR of ternary YOLIC models, and the proposed fault-tolerant ensemble consisting of DMR of the ternary YOLIC model.

We prepared eight original FP-YOLIC models for evaluation. We selected one FP-YOLIC model, and converted it to eight ternary YOLIC models with different random seeds. We generated 56 ensemble models consisting of three ternary models using all possible combinations of the eight models and evaluated their accuracy. Table III shows the F1-score and BOPs of each fault-tolerant model. This table shows the minimum, average, and maximum accuracy values for eight FP-YOLIC models, eight ternary models, and 56 fault-tolerant ensemble models. The table also shows the weight bit, activation bit, the number of BOPs for each model, and the percentage of the computational cost compared to traditional methods.

Note that we optimistically estimate the BOPs for FP-YOLIC using Equation (8) though floating-point operations are more complicated. The result shows that the ensemble model reduced the BOPs by about 79.7% of a FP-YOLIC model. This ensemble model improves the accuracy lowered by quantization to be comparable to the accuracy of the original model.

Figure 6 is an example of road risk detection by FP-YOLIC and fault-tolerant ensemble model. Figure 6(a) and (b) represent the results of the FP-YOLIC model and the proposed fault-tolerant model, respectively. In these images, the colored lines represent detected objects, where yellow indicates people, orange indicates vehicles, including bicycles, and green indicates dents. The proposed fault-tolerant model detects road risk as well as the FP-YOLIC model with less computational cost.

E. Road surface damage detection

We performed experiments on the road surface damage dataset as same as road risk detection. We evaluated eight quantized models and 56 ensemble models as well as road risk detection.

Table IV shows the F1-score and BOPs for each model. The result shows that the ensemble model is the best for the minimum, average, and maximum items. This ensemble model also reduced the BOPs by about 89.5% of an FP-YOLIC model. This shows the effect of ensemble learning of ternary models, which improves the F1-score lowered by ternary quantization to more than that of the original model.

Figure 7 is an example of road surface damage detection. Figures 7(a) and (b) represent the output of FP-YOLIC and the proposed fault-tolerant YOLIC model, respectively. The green line in this image represents the ground truth, and the red line represents the detected cell. Although road surface detection, such as pot-hole detection, is still challenging, the proposed fault-tolerant model achieves as reliable detection as the TMR of FP-Yolic models with much lower computational cost.

V. Conclusion

In this paper, we have proposed a method to realize object localization and classification that has low computational cost and is resistant to failures. The proposed fault-tolerant ensemble model consisting of DMR of ternary YOLIC models can detect permanent hardware faults and transient faults. We demonstrate that the proposed method achieves comparable accuracy to the original floating point models while reducing computational cost compared to traditional TMR approaches.

According to our experiments, the quality of the fault-tolerant ensemble models has varied, though their average has been comparable with the average quality of floating-point models. Therefore, how to reduce the variation and obtain a better ensemble model with shorter training time is in our future work. We also plan to implement and evaluate the proposed fault-tolerant models with FPGA devices.

Acknowledgement

We would like to thank Prof. Qiangfu Zhao, from the University of Aizu in Japan, for the useful discussions and his valuable advice.

References

[1] U. Zahid, G. Gambardella, N. J. Fraser, M. Blott, and K. Vissers, "Fat: Training neural networks for reliable inference under hardware faults," in *2020 IEEE International Test Conference (ITC)*, pp. 1–10, IEEE, 2020.

[2] W. Li, X. Ning, G. Ge, X. Chen, Y. Wang, and H. Yang, "Ftt-nas: Discovering fault-tolerant neural architecture," in *2020 25th Asia and South Pacific Design Automation Conference (ASP-DAC)*, pp. 211–216, 2020.

TABLE III
THE RESULTS FOR ROAD RISK DETECTION.

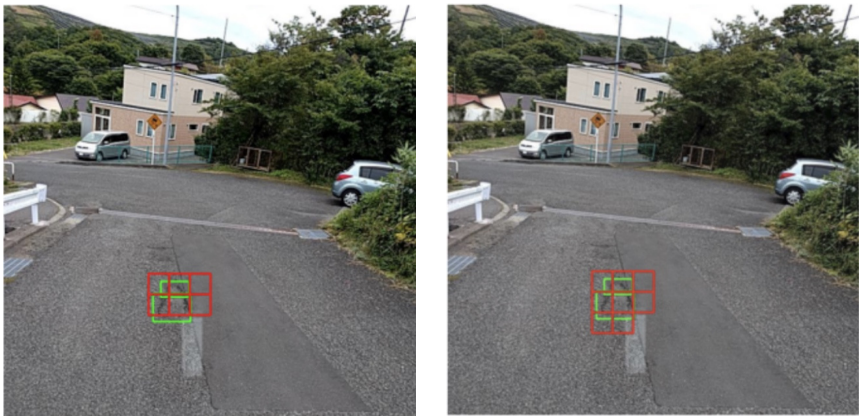
model	weight bit	activation bit	min. F1-score	ave. F1-score	max. F1-score	BOPs (Ratio)
TMR of FP-YOLIC	32	32	0.8609	0.8632	0.8657	33583G (100%)
TMR of ternary YOLIC	2	8	0.8537	0.8543	0.8548	3399G (10.1%)
Ours	2	8	0.8596	0.8603	0.8613	6801G (20.3%)

(a) FP-YOLIC (b) Ours

Fig. 6. Examples of road risk detection resuls.

TABLE IV
THE RESULTS FOR ROAD SURFACE DAMAGE DETECTION.

model	weight bit	activation bit	min. F1-score	ave. F1-score	max. F1-score	BOPs (Ratio)
TMR of FP-YOLIC	2	2	0.6181	0.6270	0.6372	33583G (100.0%)
TMR of ternary YOLIC	2	2	0.5842	0.6024	0.6332	1770G (5.3%)
Ours	2	2	0.6187	0.6375	0.6597	3539G (10.5%)



(a) FP-YOLIC (b) Ours

Fig. 7. Examples of road surface damage detection results.

[3] W. Wang, S. Yang, S. Bhardwaj, S. Vrudhula, F. Liu, and Y. Cao, "The impact of nbti effect on combinational circuit: Modeling, simulation, and analysis," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 18, no. 2, pp. 173–183, 2010.

[4] A. Lund, Z. Hammadeh, P. Kenny, V. Bansal, A. Kovalov, H. Watolla, A. Gerndt, and D. Lüdtke, "Scosa system software: the reliable and scalable middleware for a heterogeneous and distributed on-board computer architecture," *CEAS Space Journal*, vol. 14, 05 2021.

[5] Y. Li, Y. Liu, M. Li, Y. Tian, B. Luo, and Q. Xu, "D2nn: a fine-grained dual modular redundancy framework for deep neural networks," in *Proceedings of the 35th Annual Computer Security Applications Conference*, pp. 138–147, 2019.

[6] S. Koeda, Y. Tomioka, and H. Saito, "Fault-tolerant ensemble cnns increasing diversity based on knowledge distillation," in *2023 IEEE 16th International Symposium on Embedded Multicore/Many-core Systems-on-Chip (MCSoC)*, pp. 399–405, 2023.

[7] K. Su, H. Wang, I. M. Chowdhury, Q. Zhao, and Y. Tomioka, "You only look at interested cells: Real-time object detection based on cell-wise segmentation," *2020 11th International Conference on Awareness Science and Technology (iCAST)*, pp. 1–6, 2020.

[8] K. Su, Y. Tomioka, Q. Zhao, and Y. Liu, "Yolic: An efficient method for object localization and classification on edge devices," *Image and Vision Computing*, 2024.

[9] G. Jocher, "YOLOv5 by Ultralytics," May 2020. https://github.com/ultralytics/yolov5.

[10] G. Jocher, A. Chaurasia, and J. Qiu, "Ultralytics YOLO," Jan. 2023.

https://github.com/ultralytics/ultralytics.

[11] M. Courbariaux, I. Hubara, D. Soudry, R. El-Yaniv, and Y. Bengio, "Binarized neural networks: Training deep neural networks with weights and activations constrained to+ 1 or-1," *arXiv preprint arXiv:1602.02830*, 2016.

[12] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[13] D. Arya, H. Maeda, S. K. Ghosh, D. Toshniwal, H. Omata, T. Kashiyama, and Y. Sekimoto, "Global road damage detection: State-of-the-art solutions," in *2020 IEEE International Conference on Big Data (Big Data)*, (Los Alamitos, CA, USA), pp. 5533–5539, IEEE Computer Society, dec 2020.

[14] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings* (Y. Bengio and Y. LeCun, eds.), 2015.

[15] C. Baskin, N. Liss, E. Schwartz, E. Zheltonozhskii, R. Giryes, A. M. Bronstein, and A. Mendelson, "Uniq: Uniform noise injection for non-uniform quantization of neural networks," *ACM Transactions on Computer Systems*, vol. 37, p. 1–15, Nov. 2019.