2024 하계 세미나

Quantization for computer vision tasks

Sogang University
Vision & Display Systems Lab, Dept. of Electronic Engineering

Presented By
양진철

Outline

- Intro
  - What is quantization?

- Papers
  - Reg-PTQ: Regression-specialized Post-training Quantization for Fully Quantized Object Detector (CVPR 2024)
  - PTQ4SAM: Post-Training Quantization for Segment Anything (CVPR 2024)

Intro

- What is quantization?
  - 모델 최적화를 위한 motivation
    - Performance ↑ → Model size ↑
      - 컴퓨터 비전에서 모델들은 모델 사이즈를 크게 가지면서 성능을 향상
        - →모델 학습의 시간, latency 및 비용 증가
    - Edge device
      - Edge device의 부족한 메모리 용량
    - Applications such as real-time intelligent
      - health care monitoring, autonomous driving, …
  - Method for optimizing models
    - Quantization, Pruning, Knowledge Distillation, Efficient Network Design
  - Quantization은 파라미터의 값(weight, activation)의 표현 정밀도를 낮추는 과정
    - Floating point (FP32) value → INT value
  - Basic equations

Quantization : $x_q = \mathrm{clamp}(\left\lfloor \frac{x}{s} \right\rceil + z, 0, 2^b - 1)$

DeQuantization : $\hat{x} = s \cdot (x_q - z)$

scale factor $s = \frac{\beta - \alpha}{2^b - 1}$

Zero-point $z = \left\lfloor -\frac{\min(x)}{s} \right\rceil$

Intro

- What is quantization?

quantization

How to quantize?
- Dynamic
- Static (calibration, clipping)

Hardware-aware?
- Integer-only
- Simulation (fake quantization)

What to quantize?
- Weight
- Activation map

Uniform vs. Non-uniform?
- Asymmetric
- Symmetric
- Power of two

When to quantize?
- Post-training quantization
- Quantization-aware training

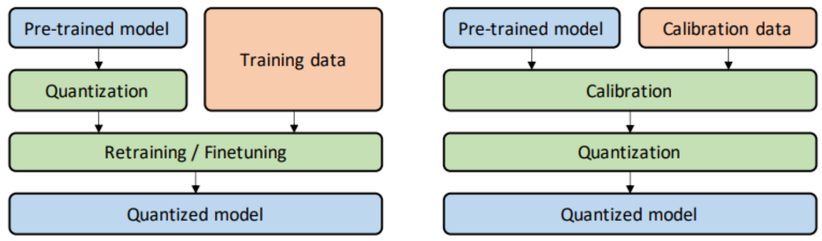
How much to quantize?

Binary, Ternary, INT K, FP16, Mixed-precision

Intro

- What is quantization?
  - Fine-tuning methods : PTQ vs QAT
    - Post-Training Quantization (PTQ)
      - Fine-tuning 없이 pre-trained model에서 모든 weight, activation quantization 파라미터를 quantization하는 방식
      - Inference에서 quantization하는 방법
      - QAT와 비교하여 낮은 accuracy
    - Quantization-Aware Training (QAT)
      - Fine-tuning을 하면서 loss를 최소로 하는 최적의 파라미터 찾는 방식
      - Loss를 최소로 하는 최적의 파라미터 찾기 위해 fine-tuning에 많은 시간과 비용을 들이는 단점 존재
      - PTQ와 비교하여 높은 accuracy 달성

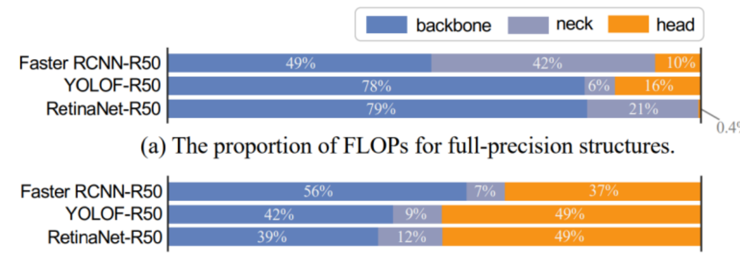
Overview of QAT and PTQ



< Left : QAT, Right : PTQ >

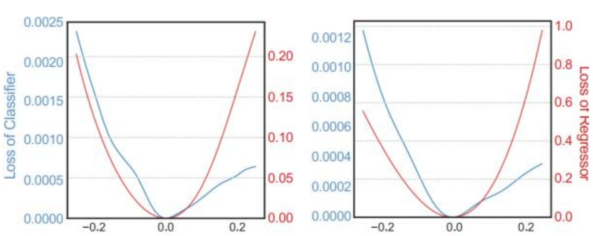
Reg-PTQ[1)]

- Introduction
  - 기존 classification task에서 제안한 방법론의 한계
    - Obeject detection task에 기존 PTQ 방법 적용 시 성능 하락
  - 기존 obeject detection task에서 제안한 방법론의 한계
    - Detector의 head를 quantization 하지 않음
  - Obeject detection 모델에 대한 분석 및 새로운 quantization 방법 제안
- Analysis
  - 1) Regressor is more sensitive to perturbation than classifier.
  - 2) Minimizing local quantization error selects sub-optimal scaling factors for regressor.
  - 3) Regressor has non-uniform weight distributions, which differs from the classifier.



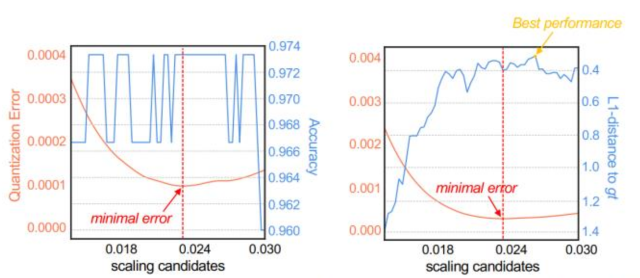
(a) The proportion of FLOPs for full-precision structures.

(b) The proportion of memory footprint for full-precision structures.



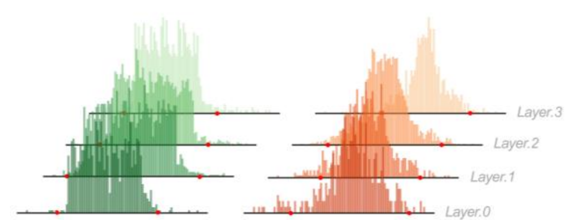
(a) Perturbation sampled from Gaussian distribution

(b) Perturbation sampled from uniform distribution



(a) Quantization error and accuracy of quantized classifier

(b) Quantization error and L1-distance to *gt* (ground truth) of quantized regressor



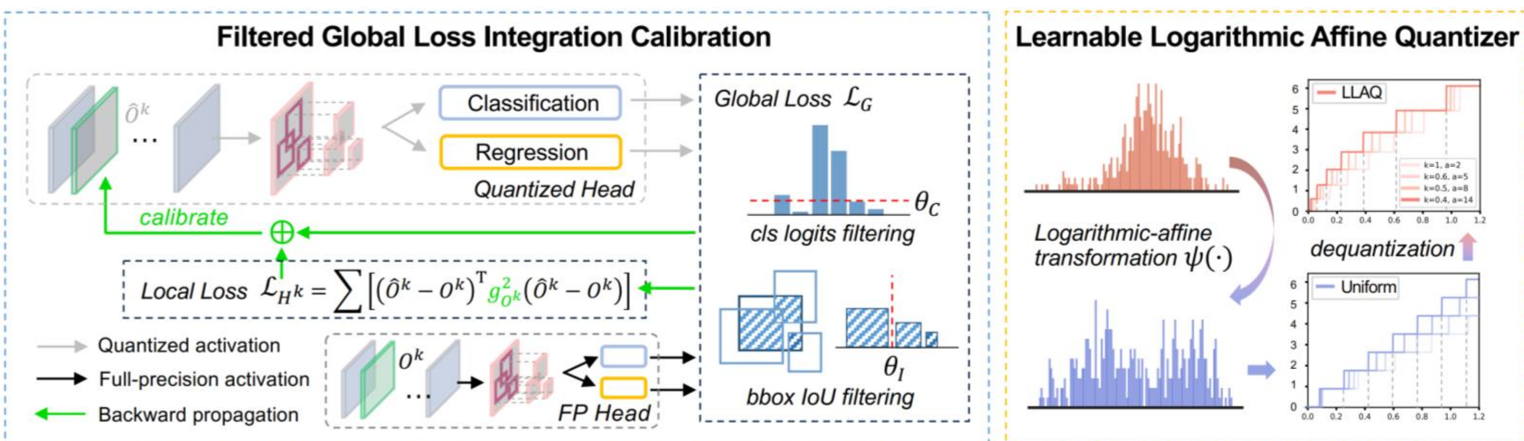
(a) Weight in classifier

(b) Weight in regressor

1) Ding, Yifu, et al. "Reg-PTQ: Regression-specialized Post-training Quantization for Fully Quantized Object Detector." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2024.

Reg-PTQ[1)]

- Method
  - Reg-PTQ[1)] method
    - Filtered Global Loss Integration Calibration (FGIC)
      - Analysis 2), local quantization error를 최소화하는 방법이 최적의 scale factor를 구하지 못함
      - 흔히 사용되는 Hessian-guided metric calibration 방법은 문제가 존재
    - Learnable Logarithmic Affine Quantizer (LLAQ)
      - Analysis 3), regression of object location의 weight distribution
    - Regression head에 LLAQ → FGIC를 통해 파라미터 미세 조정



Reg-PTQ[1)]

- Method
  - Filtered Global Loss Integration Calibration (FGIC)
    - Hessian-guided metric calibration

O^k: k-th layer output
n:number of bounding box
λ: hypher parameter

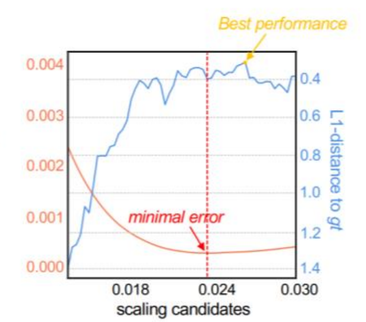
$$L_{H^k} = \sum_i \left[\left(\hat{O}_i^k - O_i^k\right)^{\mathrm{T}} \left(\frac{\partial L}{\partial O_i^k}\right)^2 \left(\hat{O}_i^k - O_i^k\right) \right]$$

    - Global Loss Integration Calibration (GIC)

$$L_G = \frac{1}{n}\sum_{i=1}^{n}\Big(\underbrace{L_{CE}(\overbrace{\hat{y}_i}^{\text{classification logit}}, y_i)}_{\text{classification loss}} + \lambda \underbrace{L_p(\overbrace{\hat{b}_i}^{\text{regressed bbox}}, b_i)}_{\text{regression loss}}\Big)$$

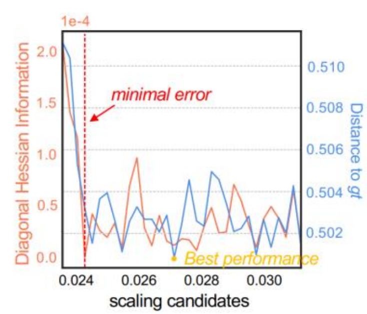
Calibration을 위한 total loss

$$L_{tot}^k = L_{H^k} + L_G$$



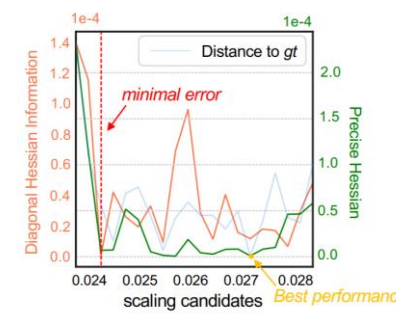
(b) Quantization error and L1-distance to *gt* (ground truth) of quantized regressor

Hessian-guided metric



(a) Diagonal Hessian Information and distance to *gt* (ground truth)

FGIC



(b) Diagonal Hessian Information and precise Hessian

1) Ding, Yifu, et al. "Reg-PTQ: Regression-specialized Post-training Quantization for Fully Quantized Object Detector." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2024.

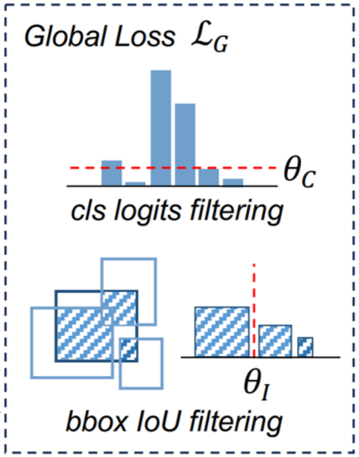
Reg-PTQ[1)]

- Method
  - Filtered Global Loss Integration Calibration (FGIC)
    - Global Loss Integration Calibration (GIC)
      - Global loss L_G 의 문제
        - ✓ Detection head는 classification score를 포함한 수 많은 bounding box 출력
        - ✓ 낮은 confidence score와 IoU는 최적의 scale factor를 구하는데 방해가 됨
      - Two-step bounding boxes filtering mechanism
        - ✓ 높은 confidence score와 IoU를 선택하기 위한 mechanism

$$b' = b \cdot \Gamma_{HC} \cdot \Gamma_{HI} \qquad \hat{b}' = \hat{b} \cdot \Gamma_{HC} \cdot \Gamma_{HI}$$

        - ✓ Γ는 bbox가 필터링되는지를 나타내는 position indicator

$$\Gamma_{HC} = \begin{cases} 1, & \text{if y} \geq \theta_c \\ 0, & \text{otherwise} \end{cases} \qquad \Gamma_{HI} = \begin{cases} 1, & \text{if IoU}(b, \hat{b}) \geq \theta_I \\ 0, & \text{otherwise} \end{cases}$$



    - Global Loss L_G

$$L_G = \frac{1}{n}\sum_{i=1}^{n}(L_{CE}(\hat{y}_i, y_i) + \lambda L_p(\hat{b}_i, b_i)) \quad \Rightarrow \quad L_G = \frac{1}{n}\sum_{i=1}^{n}(L_{CE}(\hat{y}_i, y_i) + \lambda L_p(\hat{b} \cdot \Gamma_{HC} \cdot \Gamma_{HI}, b \cdot \Gamma_{HC} \cdot \Gamma_{HI}))$$

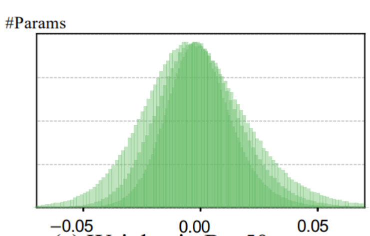
Reg-PTQ[1)]

- Method
  - Learnable Logarithmic Affine Quantizer (LLAQ)
    - Regression of object location의 weight distribution
      - Weight distribution 중앙에 집중 → laplace, gaussian distribution과 유사한 형태
      - Non-uniform한 distribution를 처리를 위한 quantization 방법 필요
    - 확률 밀도 함수를 지수 함수에서 선형 공간으로 변환
      - Laplace distribution 고려 → $f(x|\mu,\lambda) = \frac{1}{2\lambda} \mathrm{e}^{-\frac{|x-\mu|}{\lambda}}$, μ : location, λ: scale parameter
      - Logarithmic-affine transformation ψ → 로그 공간으로 투영하기 위함

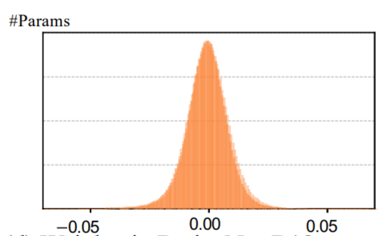
$$\psi(x) = k^* \log_e x + \mathrm{a}^*, k^*: \text{scale}, a^*: \text{offset (learnable parameter)}$$

      - Location μ에 따라 두 부분으로 나누어 Logarithmic-affine transformation 을 적용

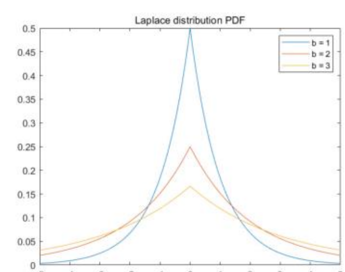
$$\psi(f(x|\mu,\lambda)) \begin{cases} \frac{k^+}{\lambda}(\mu - x) - k^+ \log_e 2\lambda + a^+, & \text{if } \mathrm{x} \geq \mu, \\ \frac{k^-}{\lambda}(x - \mu) - k^- \log_e 2\lambda + a^-, & \text{otherwise} \end{cases}$$

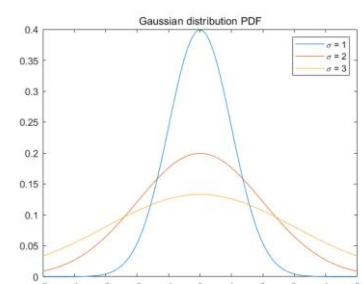


(c) Weights in Res50 convs



(d) Weights in RetinaNet-R18 convs



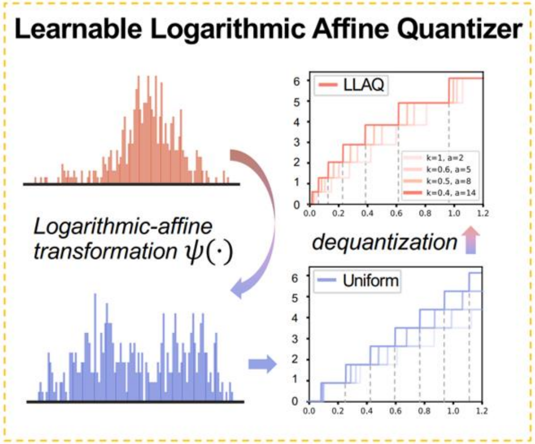


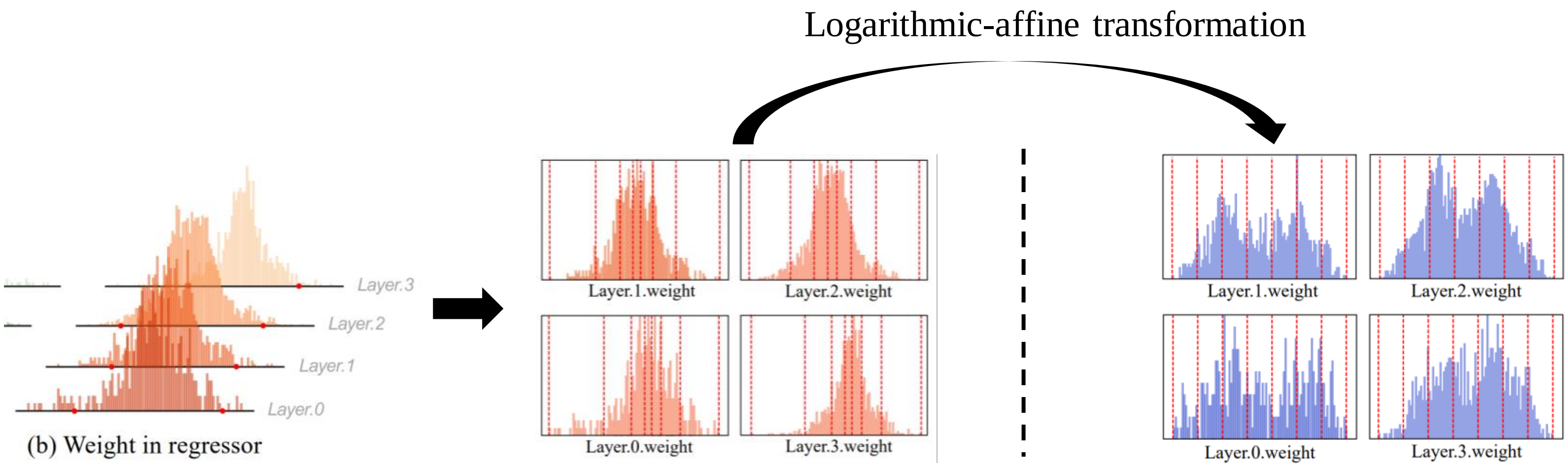
Reg-PTQ[1)]

- Method
  - Learnable Logarithmic Affine Quantizer (LLAQ)
    - Logarithmic-affine transformation

$$\psi(f(x|\mu,\lambda)) \begin{cases} \frac{k^+}{\lambda}(\mu - x) - k^+ \log_e 2\lambda + a^+, & \text{if x} \geq \mu, \\ \frac{k^-}{\lambda}(x - \mu) - k^- \log_e 2\lambda + a^-, & \text{otherwise} \end{cases}$$

※ Weight in regressor를 uniform distribution 에 유사하게 변환





< Non-uniform weight 에 대한 quantization scale 표현 >

< Weight distribution >

Reg-PTQ[1)]

- Experimental results
  - Comparison with other PTQ methods on various detectors with ResNet-50/101 as the backbone on COCO dataset

Method	#Bit$_{(W/A)}$	RetinaNet		YOLOF	Faster RCNN		Mask RCNN	
		ResNet-50	ResNet-101	ResNet-50	ResNet-50	ResNet-101	ResNet-50	ResNet-101
Full-precision	32/32	37.4	38.9	37.5	38.5	39.8	39.2	40.8
BRECQ	2/4	14.0	18.7	10.8	12.5	13.0	11.0	12.0
PD-Quant	2/4	19.3	20.6	15.4	1.7	6.1	11.8	10.8
QDrop	2/4	19.9	22.9	17.4	17.8	19.9	18.2	19.9
Reg-PTQ (Ours)	**2/4**	**23.9**	**24.8**	**19.3**	**19.1**	**21.5**	**19.1**	**20.7**
AdaRound	3/3	19.3	20.7	7.7	21.2	22.8	21.6	22.6
AdaQuant	3/3	21.1	19.9	13.3	4.8	5.8	4.5	4.4
BRECQ	3/3	22.8	24.6	18.4	16.7	16.5	15.9	15.2
PD-Quant	3/3	24.5	25.6	22.2	14.0	14.0	18.7	17.3
QDrop	3/3	26.5	26.8	25.8	23.6	24.1	24.4	24.7
Reg-PTQ (Ours)	**3/3**	**28.1**	**28.3**	**27.3**	**28.1**	**29.1**	**28.4**	**28.8**
AdaRound	4/4	20.5	20.8	17.1	0.6	23.8	24.3	24.8
AdaQuant	4/4	33.5	34.5	25.6	12.8	14.5	12.0	14.6
BRECQ	4/4	34.2	35.8	29.0	28.8	30.8	31.7	30.1
PD-Quant	4/4	33.2	33.4	31.4	25.7	28.3	27.6	27.5
QDrop	4/4	34.1	35.1	33.4	33.7	34.4	34.5	35.6
SubSetQ	4/4	33.4	35.0	31.8	33.3	35.4	34.9	36.8
Reg-PTQ (Ours)	**4/4**	**36.7**	**35.9**	**34.3**	**36.7**	**36.2**	**36.4**	**37.2**
AdaQuant	4/8	36.5	38.1	35.0	16.9	19.2	14.2	18.4
BRECQ	4/8	36.8	38.6	36.2	20.0	22.0	21.2	23.4
PD-Quant	4/8	36.8	38.5	36.5	24.1	24.2	27.4	26.9
QDrop	4/8	37.0	38.5	36.7	37.6	38.9	38.2	39.9
SubSetQ	4/8	36.7	38.3	36.2	36.1	38.7	38.1	39.8
Reg-PTQ (Ours)	**4/8**	**37.4**	**38.6**	**36.8**	**37.8**	**39.1**	**38.3**	**40.0**

Reg-PTQ[1)]

- Experimental results
  - Ablation studies

– FGIC 하이퍼파라미터 조정을 통한 효과 입증

baseline: 23.0		θ_C 0 (w/o)	5e-5	2e-4	1e-3	1e-2
θ_I	0 (w/o)	23.5	23.9	23.8	23.5	23.2
	0.1	23.8	23.9	23.9	23.8	23.5
	0.5	23.8	23.8	23.8	23.8	23.2

Table 2. Ablation study of FGIC and sensitivity analysis of its hyperparameters, θ_C and θ_I, on RetinaNet ResNet-50 on COCO under W2A4 quantization. Baseline means solely using local loss.

- LLAQ 효과 입증

Model	Quantizer	W2A4	W3A3	W4A4
One-Stage (Bbox Head)	Uniform	23.0	27.2	35.2
	LLAQ	23.6	28.0	35.7
Two-Stage (Rpn+Roi Heads)	Uniform	22.3	28.8	34.3
	LLAQ	23.7	31.7	36.4

Table 3. Comparison of uniform and LLAQ quantizers under various bitwidth on COCO. Models used here are RetinaNet ResNet-50 and Faster RCNN ResNet-50.

– Efficiency and storage reduction on single NVIDIA Tesla T4 implemented with TVM

$\#\text{Bit}_{(W/A)}$	Quantize Backbone & Neck FLOPs (G)	Quantize Backbone & Neck Storage (M)	Fully Quantize FLOPs (G)	Fully Quantize Storage (M)
2/4	25.48	21.78	12.14	5.97
4/4	35.24	23.46	22.65	8.70
4/8	54.75	23.46	43.95	8.70

(a) Faster RCNN ResNet-50. The full-precision one has 171.8 GFLOPs and 46.91 M Storage while processing one sample.

$\#\text{Bit}_{(W/A)}$	Quantize Backbone & Neck FLOPs (G)	Quantize Backbone & Neck # Storage (M)	Fully Quantize FLOPs (G)	Fully Quantize Storage (M)
2/4	8.06	37.11	7.89	14.9
4/4	14.73	39.08	14.46	18.42
4/8	28.07	39.08	27.84	18.42

(b) RetinaNet ResNet-50. The full-precision one has 108.10 GFLOPs and 66.60 M Storage while processing one sample.

Table 4. The FLOPs (G) and the Storage (M) of different detectors under different bit-width settings.

DataType	$\text{Latency}_{(ms)}$	$\text{Storage}_{(MB)}$
Float32	796.4	129.7
INT16	438.4	68.6
INT4*	132.8	38.6
INT4	84.5	22.8

Table S5. Efficiency and storage reduction on single NVIDIA Tesla T4 implemented with TVM. **DataType** denotes the weights and activation datatype. INT4* means we only quantize backbone and FPN neck to 4-bit but leave the heads full-precision. Other results without * means full quantization with uniform bitwidth.

서강대학교 SOGANG UNIVERSITY

VDS LAB

Reg-PTQ[1)]

- Experimental results
  - Visualization of detection results by full-precision (FP) detectors and 3-bit quantized models

FP QDrop Ours

Person
0.91 / 0.50 / 0.71

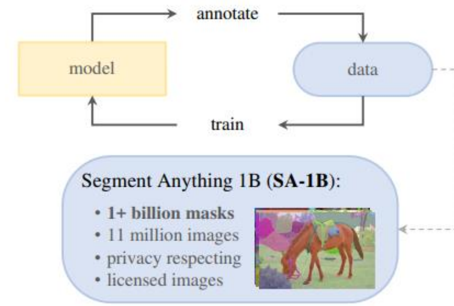
Car
0.90 / 0.64 / 0.79

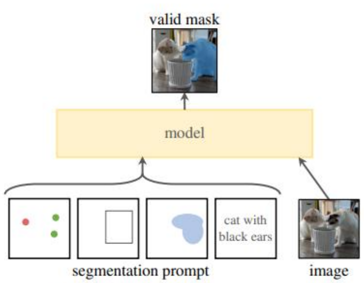
Pizza
0.86 / 0.47 / 0.82

1) Lv, Chengtao, et al. "PTQ4SAM: Post-Training Quantization for Segment Anything." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2024.
2) Kirillov, Alexander, et al. "Segment anything." *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023.

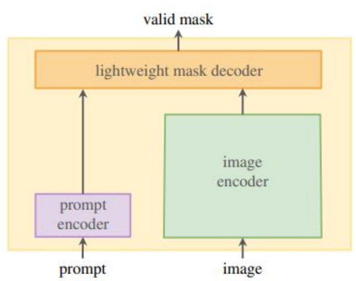
PTQ4SAM[1)]

- Background
  - Segment Anything Model (SAM[2)])
    - Segmentation을 위한 범용적인 foundation 모델 → zero-shot
      - Foundation 모델 : 대규모 데이터셋으로 pretraining 시킨 거대한 모델
      - Prompt와 image 를 입력으로 하여 mask를 출력하는 task
    - Model
      - Image encoder
        - ✓ Pre-trained ViT를 encoder로 하여 image를 입력으로 하여 image embedding을 출력
      - Prompt encoder
        - ✓ point || box || text 와 같은 prompt를 입력으로 하여 token 출력
      - Mask decoder
        - ✓ Prompt self-attention과 cross-attention 을 양방향으로 활용 (img to token / token to img attn)

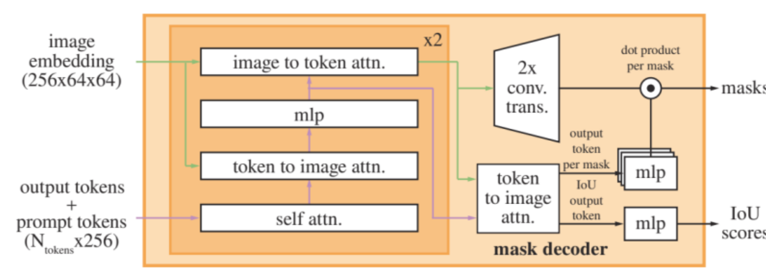




(a) **Task**: promptable segmentation



(b) **Model**: Segment Anything Model (**SAM**)

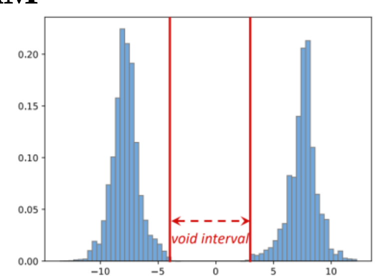


1) Lv, Chengtao, et al. "PTQ4SAM: Post-Training Quantization for Segment Anything." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2024.
2) Kirillov, Alexander, et al. "Segment anything." *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023.

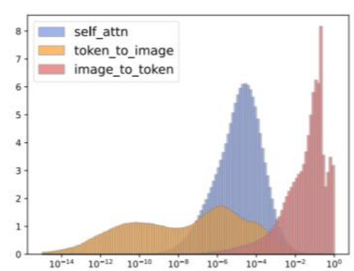
PTQ4SAM[1)]

- Introduction
  - 기존 classification task에서 제안한 방법론의 한계
    - Segment Anything Model (SAM[2)])에 기존 PTQ 방법 적용 시 성능 하락
  - SAM[2)] 에 대한 분석 및 새로운 quantization 방법 제안
- Analysis
  - Bimodal distribution
    - ViT backbone 모델과 달리 bimodal distribution을 나타내는 activation map이 존재
    - Post-Key-Linear 에서 주로 나타남
  - Post-Softmax distribution
    - 다양한 attention mechanism 으로 인해 post-Softmax 이후의 다양한 distribution

SAM

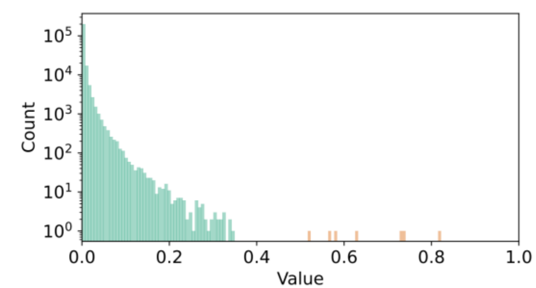


< Bimodal distribution >



< Post-Softmax distribution >

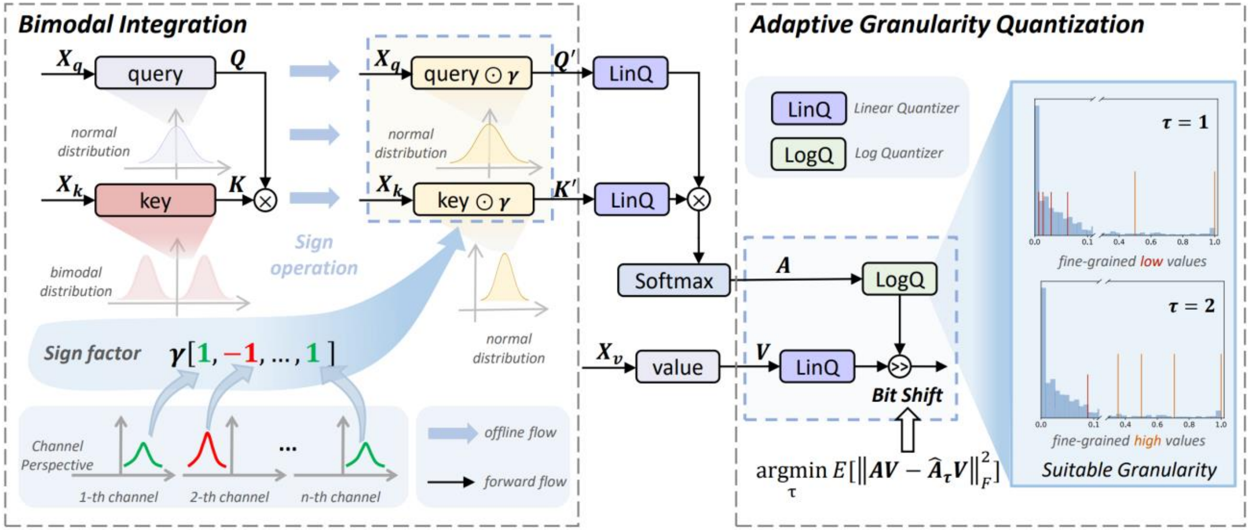
DeiT-S



< Post-Softmax distribution >

PTQ4SAM[1)]

- Method
  - PTQ4SAM[1)] method
    - Bimodal Integration (BIG)
      - Post-Key-Linear 에서의 bimodal distribution
      - 2개의 peaks와 중앙의 빈 간격은 성능 하락의 요인
    - Adaptive Granularity Quantization (AGQ)
      - Post-Softmax 에서의 복잡한 distribution
      - 3가지 방법에 동일한 quantzation 방법 적용 시 주요한 정보를 잃어버릴 가능성 존재



서강대학교 SOGANG UNIVERSITY

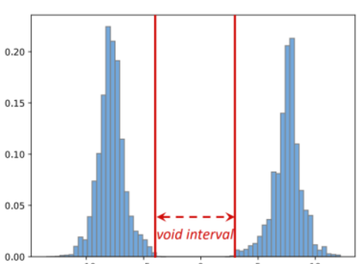
VDS LAB

PTQ4SAM[1)]

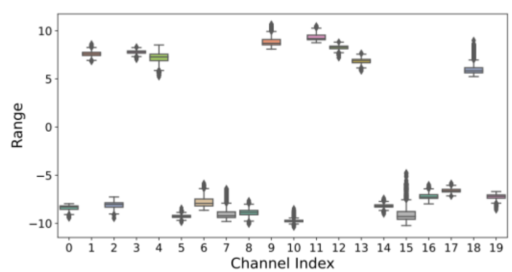
- Method
  - Bimodal Integration (BIG) strategy
    - 두 가지 관점에서의 심층 분석
      - Per-tensor perspective : 두 개의 피크를 포함하며 중심을 기준으로 대칭적
      - Per-channel perspective : 채널별 값들은 고정된 피크에 존재하여 비대칭적
    - Bimodal Integration (BIG)
      - 채널 별 sign factor γ 채택
        - ✓ Bimodal distribution을 normal distribution으로 변환 시켜주는 factor
        - ✓ γ는 각 채널의 평균 값을 고려하여 sign factor를 계산한다고 가정

$$\gamma_j = \begin{cases} +1, & \text{if } \mathrm{mean}\left(K_{:,j}\right) \geq 0 \\ -1, & \text{otherwise} \end{cases}$$

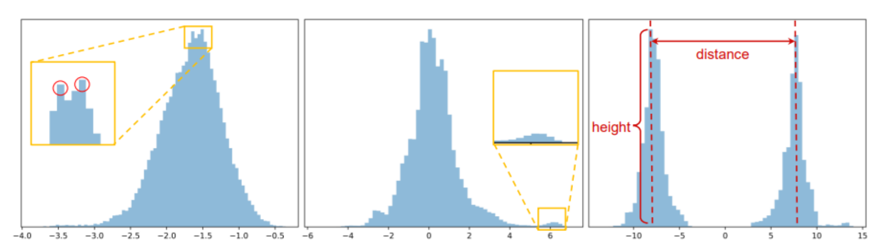
      - Bimodal Discovery
        - ✓ BIG를 사용하기 위한 제약 충족



< Per-tensor perspective>



< Per-channel perspective>

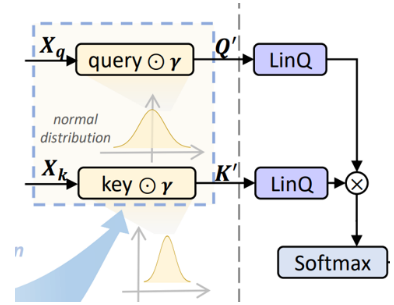


< Three typical examples in BIG strategy >

1) Lv, Chengtao, et al. "PTQ4SAM: Post-Training Quantization for Segment Anything." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2024.

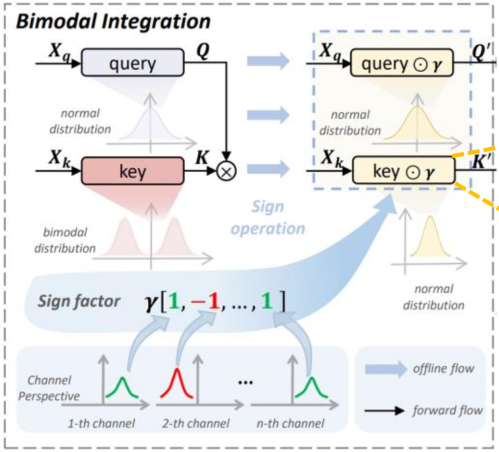
PTQ4SAM[1)]

- Method
  - Bimodal Integration (BIG) strategy
    - Bimodal Integration (BIG)
      - BIG strategy를 사용한 Query, Key 계산 및 연산

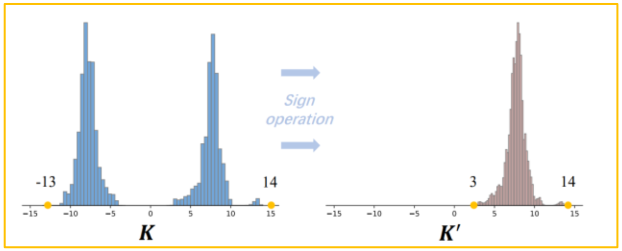


$$\left\{\begin{array}{l} QK^{\mathrm{T}} = \underbrace{\left(X_qW_q + b_q\right)}_{\text{normal distribution}}\underbrace{\left(X_kW_k + b_k\right)^{\mathrm{T}}}_{\text{bimodal distribution}} \\ \\ \gamma_{\mathrm{j}} = \begin{cases} +1, & \text{if } \mathrm{mean}\left(K_{:,j}\right) \geq 0 \\ -1, & \text{otherwise} \end{cases} \end{array}\right.$$

$$\begin{aligned} QK^{\mathrm{T}} &= \left(\left(X_qW_q + b_q\right)\odot\gamma\right)\left(\left(X_kW_k + b_k\right)^{\mathrm{T}}\right)\odot\gamma^{\mathrm{T}}\big) \\ &= \underbrace{\left(X_qW'_q + b'_q\right)}_{\text{normal distribution}}\underbrace{\left(X_kW'_k + b'_k\right)^{\mathrm{T}}}_{\text{normal distribution}} \end{aligned}$$



< BIG strategy >



< The distribution of key activations before and after BIG strategy >

1) Lv, Chengtao, et al. "PTQ4SAM: Post-Training Quantization for Segment Anything." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2024.

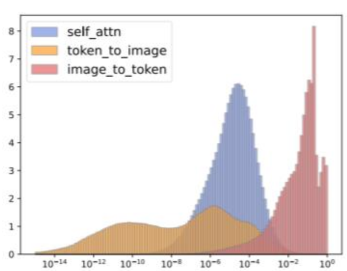
PTQ4SAM[1)]

- Method
  - Adaptive Granularity Quantization (AGQ) strategy
    - Softmax activation function
      - Softmax 함수는 attention score를 확률로 변환하여 0 ~ 1 사이의 값을 가짐
    - SAM의 attention mechanism
      - Self-attention mechanism
      - Cross-attention in two directions → SAM의 mask decoder에 존재
        - ✓ Token-to-image cross-attention
        - ✓ Image-to-token cross-attention
    - Softmax activation quantization

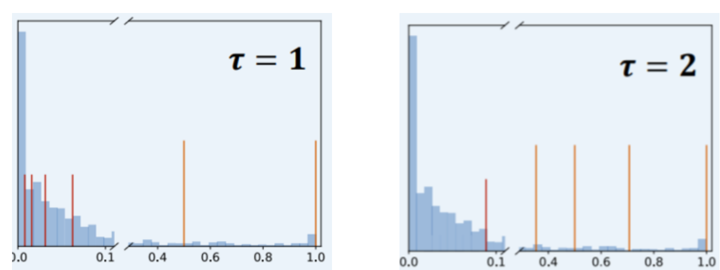
$\{\tau : 2^0, 2^1, 2^2, \dots\}$

Quantize : $a_q = \text{clamp}(\lfloor -\log_2 \frac{a}{s_a} \rceil, 0, 2^k - 1)$ → Quantize : $a_q = \text{clamp}(\lfloor -\log_{2^{\frac{1}{\tau}}} \frac{a}{s_a} \rceil, 0, 2^k - 1)$

DeQuantize : $\hat{a} = s_a \cdot 2^{-a_q}$ → DeQuantize : $\hat{a} = s_a \cdot 2^{-\frac{a_q}{\tau}}$



< Post-Softmax distribution >



< τ에 따른 quantization results >

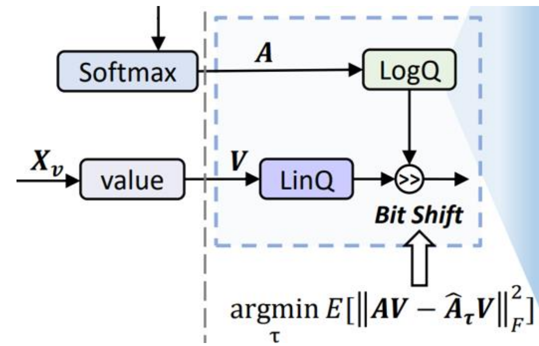
PTQ4SAM[1)]

- Method
  - Adaptive Granularity Quantization (AGQ) strategy
    - Softmax activation quantization

$$\hat{a} = s_a \cdot 2^{-\frac{a_q}{\tau}} = s_a \cdot \underbrace{2^{\lfloor -\frac{a_q}{\tau} \rfloor}}_{\text{정수}} \cdot \underbrace{2^{-\frac{(-a_q)\%\tau}{\tau}}}_{\text{소수}}$$

Bit shift

$$\hat{a} = s_a \cdot 2^{-\frac{(-a_q)\%\tau}{\tau}} \gg \left\lceil \frac{a_q}{\tau} \right\rceil$$

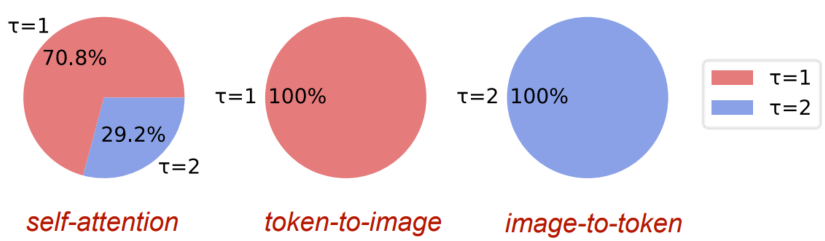


- Softmax activation 과 Value 연산

$$\hat{a} \cdot \hat{v} = s_a \cdot s_v \cdot 2^{-\frac{(-a_q)\%\tau}{\tau}} \cdot v_q \gg \left\lceil \frac{a_q}{\tau} \right\rceil$$

- 최적의 τ를 구하기 위한 목적 함수 정의
  - Real value 인 attention map A와 Value V간의 행렬 곱과 quantized A와 real value V 곱의 error 측정

$$\operatorname{argmin} E[\|AV - \hat{A}_\tau V\|_F^2]$$



< Pie charts depicting the optimal τ across various attention mechanisms in SAM-L>

1) Lv, Chengtao, et al. "PTQ4SAM: Post-Training Quantization for Segment Anything." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2024.
2) Choukroun, Yoni, et al. "Low-bit quantization of neural networks for efficient inference." *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*. IEEE, 2019.
3) Wei, Xiuying, et al. "Qdrop: Randomly dropping quantization for extremely low-bit post-training quantization." *arXiv preprint arXiv:2203.05740* (2022).

PTQ4SAM[1)]

- Experimental results
  - Quantization results of instance segmentation on COCO dataset among different detectors

Detector	Methods	SAM-B			SAM-L			SAM-H		
		FP	W6A6	W4A4	FP	W6A6	W4A4	FP	W6A6	W4A4
Faster R-CNN [45]	MinMax [17]	33.4	9.2	-	36.4	32.9	-	37.2	31.9	-
	Percentile [59]		10.9	-		33.5	-		32.0	-
	OMSE [5]		11.9	-		33.9	5.4		33.1	7.4
	PTQ4SAM-S		**15.4**	-		**35.7**	**18.1**		**36.0**	**24.1**
	AdaRound [42]		23.1	-		34.3	8.7		33.7	14.5
	BRECQ [26]		24.1	-		34.2	10.7		33.7	15.1
	QDrop [56]		29.3	13.0		35.2	22.6		36.3	32.3
	PTQ4SAM-L		**30.3**	**16.0**		**35.8**	**28.7**		**36.5**	**33.5**
YOLOX [9]	MinMax [17]	37.0	10.7	-	40.4	37.5	-	41.0	36.1	-
	Percentile [59]		12.0	-		38.0	-		36.3	-
	OMSE [5]		13.5	-		38.4	6.1		37.5	7.8
	PTQ4SAM-S		**17.4**	-		**40.0**	**20.6**		**40.3**	**26.7**
	AdaRound [42]		26.4	-		38.9	11.1		38.3	16.7
	BRECQ [26]		26.1	-		38.9	12.0		38.3	16.3
	QDrop [56]		33.6	13.3		39.7	25.3		40.4	35.8
	PTQ4SAM-L		**34.3**	**18.4**		**40.3**	**31.6**		**40.7**	**37.6**
H-Deformable-DETR [18]	MinMax [17]	38.2	10.9	-	41.5	38.6	-	42.0	37.3	-
	Percentile [59]		12.3	-		39.0	-		37.5	-
	OMSE [5]		15.0	-		39.6	6.2		38.6	7.7
	PTQ4SAM-S		**17.9**	-		**41.0**	**20.9**		**41.3**	**27.3**
	AdaRound [42]		27.2	-		39.9	8.0		39.4	16.3
	BRECQ [26]		27.9	-		39.9	11.1		39.5	15.5
	QDrop [56]		34.3	13.2		40.5	25.8		41.4	36.5
	PTQ4SAM-L		**35.1**	**17.3**		**41.2**	**32.1**		**41.6**	**38.4**
DINO [69]	MinMax [17]	44.5	11.2	-	48.6	44.7	-	49.1	42.8	-
	Percentile [59]		14.0	-		45.4	-		43.1	-
	OMSE [5]		16.6	-		45.9	6.8		44.5	8.3
	PTQ4SAM-S		**20.4**	-		**47.7**	**23.1**		**48.1**	**30.5**
	AdaRound [42]		31.2	1.2		46.6	8.8		46.0	18.2
	BRECQ [26]		31.8	3.6		46.6	12.3		46.0	17.6
	QDrop [56]		38.9	11.2		47.5	27.5		48.3	41.7
	PTQ4SAM-L		**40.4**	**14.4**		**48.3**	**36.6**		**48.7**	**43.9**

- OMSE[2)] : statistic-based quantization; activation에서 channel-wise quantization을 하지 않고, quantized tensor와 floating point tensor와의 L2 distance를 이용하여 loss 계산
- Qdrop[3)] : learning-based quantization; 최적화된 모델의 평탄성을 증가시키기 위해 reconstruction 과정에서 drop 추가
- PTQ4SAM-S : proposed method + OMSE
- PTQ4SAM-L : proposed method + Qdrop

1) Lv, Chengtao, et al. "PTQ4SAM: Post-Training Quantization for Segment Anything." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2024.
2) Jia, Ding, et al. "Detrs with hybrid matching." *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2023.
3) Ge, Z. "Yolox: Exceeding yolo series in 2021." *arXiv preprint arXiv:2107.08430* (2021).

PTQ4SAM[1)]

- Experimental results
  - Ablation studies
    - 제안한 방법 BIG, AGQ 효과 입증

Quantization results of instance segmentation on COCO dataset H-Deformable-DETR[2)] detector

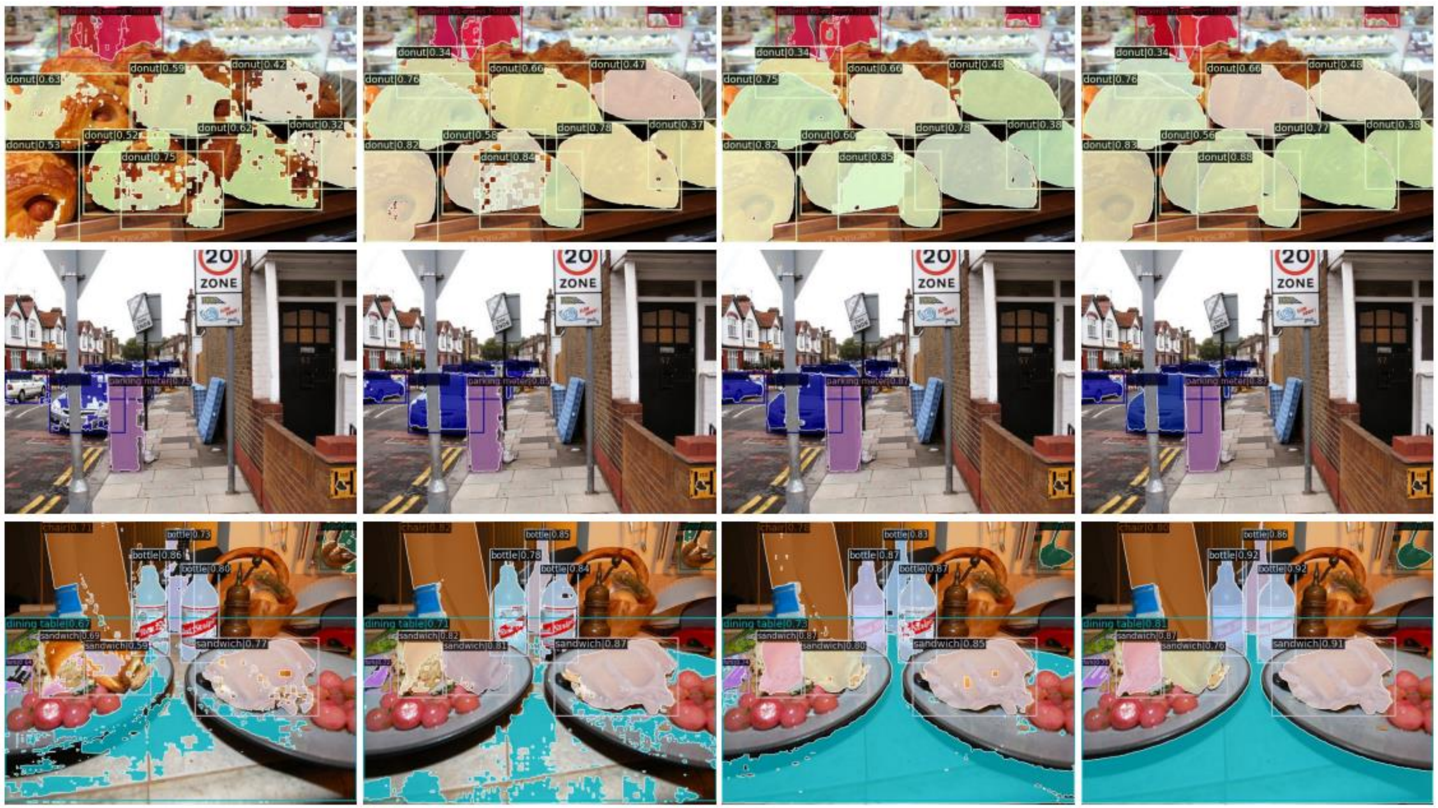
Row ID	Model	BIG	AGQ	FP	W6A6	W4A4
1	SAM-L	×	×	41.5	40.5	25.8
2		✓	×		40.6	29.2
3		×	✓		41.2	27.3
4		✓	✓		**41.2**	**32.1**

Quantization results of instance segmentation on COCO dataset YOLOX[3)] detector

#bits	Quantizer	SAM-B	SAM-L	SAM-H
Full-precision	-	37.0	40.4	41.0
W6A6	Uniform	33.6	39.7	40.4
	Log2	33.3	40.2	40.6
	AGQ (ours)	**33.9**	**40.3**	**40.6**
W4A4	Uniform	13.3	25.3	35.8
	Log2	14.1	26.5	37.3
	AGQ (ours)	**15.0**	**27.8**	**37.6**

PTQ4SAM[1)]

- Experimental results
  - Visualization of instance segmentation on 4-bit SAM-L.



BRECQ QDrop Ours FP

Donut

Parking meter
0.75/0.85/0.87/0.87

Dining table

Thank you