SCALING UP QA-NAS FOR EFFICIENT DEEP LEARNING ON THE EDGE

CODAI’23 Workshop

Yao Lu, Hiram Rayo Torres Rodriguez, Sebastian Vogel, Nick van de Waterlaat, Pavol Jancura

SEPTEMBER 21, 2023

SECURE CONNECTIONS FOR A SMARTER WORLD

PUBLIC

OVERVIEW

- Introduction & Related Work
- Quantization-Aware Block-wise NAS (Homogeneous)
- Quantization-Aware Block-wise NAS (FB-MP)
- Conclusions

OVERVIEW

- Introduction & Related Work
- Quantization-Aware Block-wise NAS (Homogeneous)
- Quantization-Aware Block-wise NAS (FB-MP)
- Conclusions

INTRODUCTION: MOTIVATIONS

- Applications of DNN models on edge devices
  - Autonomous driving
  - Real-time healthcare devices
  - Speech recognition
  - etc

[1]

[1] https://on-device-ai.com/

INTRODUCTION: MOTIVATIONS

- The **keys** to effective deployment of DNN models on edge devices:

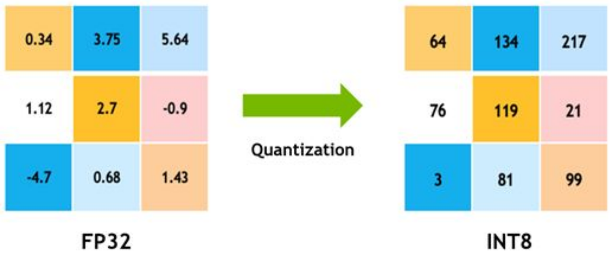
1. Low inference latency
2. Small memory footprint
3. High accuracy

- The **keys** to effective deployment of DNN models on edge devices:

1. Low inference latency
2. Small memory footprint
3. High accuracy

Model Efficiency

Quantization



[2]

[2] https://developer.nvidia.com/blog/achieving-fp32-accuracy-for-int8-inference-using-quantization-aware-training-with-tensorrt/

RELATED WORK: QUANTIZATION

- Represents the weights and activations of DNN models **using fewer bits** (e.g. INT8) than the standard FP32 representation without sacrificing much accuracy.
  - Reduce memory footprint
  - Lower inference latency

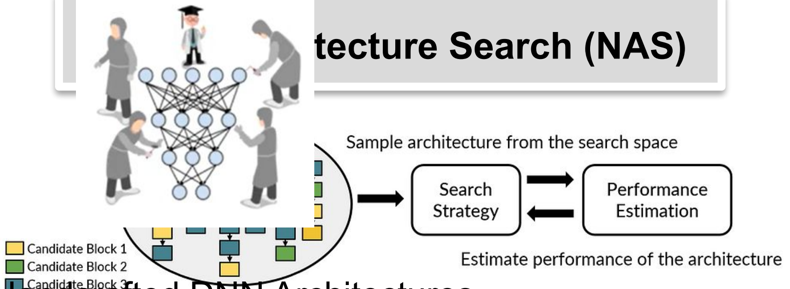
Categories:

According to different bit-width allocation strategies:

- Homogeneous Quantization
- Few-Bit Mixed-Precision (FB-MP) Quantization

- The **keys** to effective deployment of DNN models on edge devices:
  1. Low inference latency
  2. Small memory footprint
  3. High accuracy

tecture Search (NAS)



Handcrafted DNN Architectures

[3]

Sample-based NAS

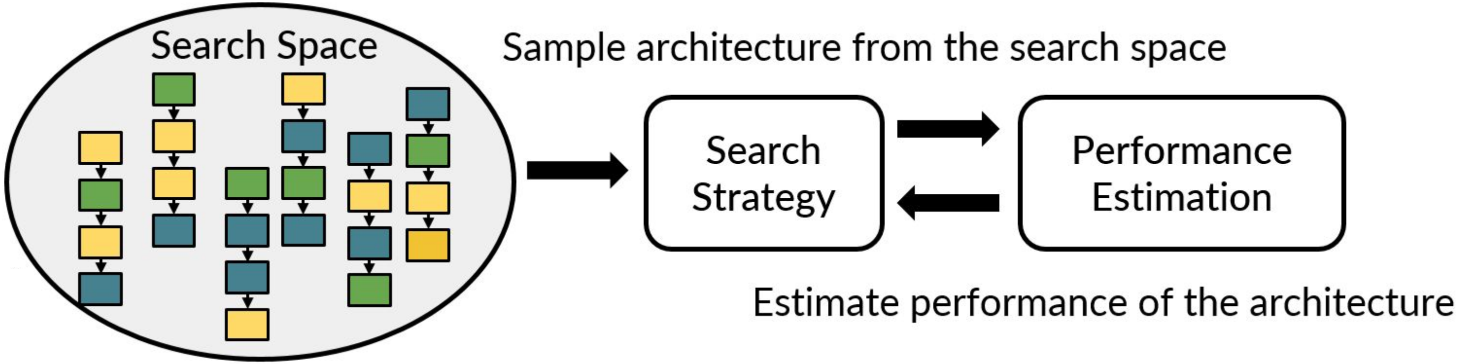
Weight-sharing NAS

Block-wise NAS

[3] https://medium.com/ai-academy-taiwan/提煉再提煉濃縮再濃縮-neural-architecture-search-介紹-ef366ffdc818

RELATED WORK: SAMPLE-BASED NAS

- Sample-based NAS
  - Sample a large number of architectures from the search space and then train each of them from scratch to validate their performance.
  - Scaling to compute-intensive tasks is intractable as the training cost will explode.

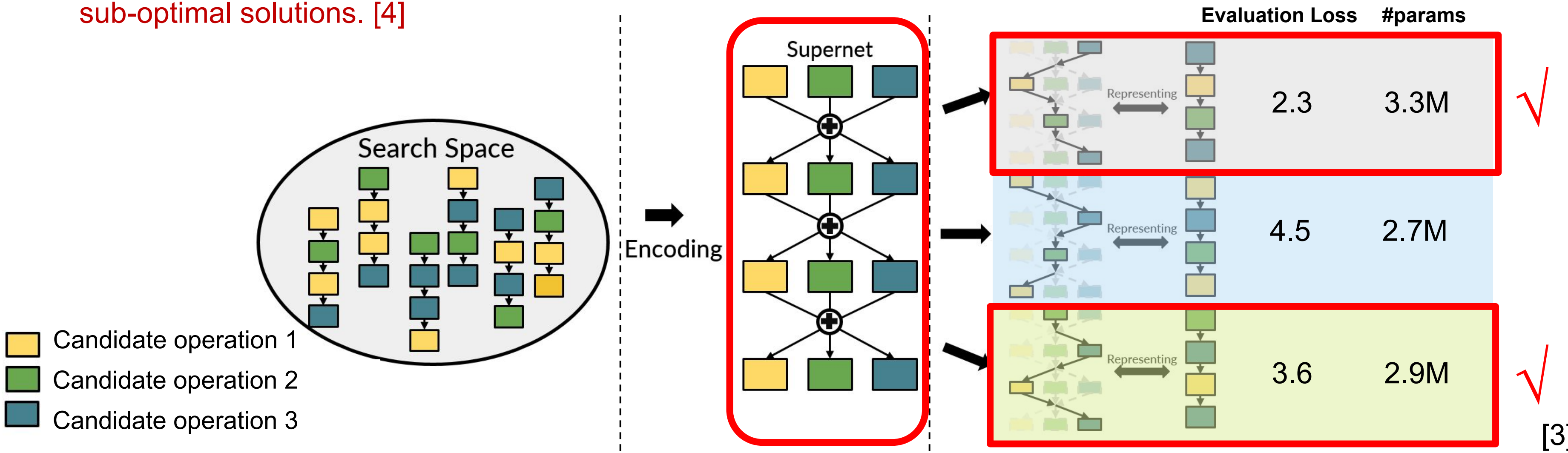


Candidate
Candidate
Candidate operation 3

[3]

RELATED WORK: WEIGHT-SHARING NAS

- Weight-sharing NAS (e.g., FairNAS[4] and SPOS[5])
  - A supernet encompassing all candidate architectures. Only supernet is trained, with candidate subnets sharing weights.
  - Evaluate and rank subnet performance for subsequent search.
  - Promising results have been shown in small search spaces.
  - Subnets can be trained insufficiently in a large search space, leading to incorrect ranking and hence, sub-optimal solutions. [4]



[3]

[4] Xiangxiang Chu, Bo Zhang, and Ruijun Xu. FairNAS: Rethinking Evaluation Fairness of Weight Sharing Neural Architecture Search. 2019.
[5] Z. Guo, X. Zhang, H. Mu, W. Heng, Z. Liu, Y. Wei, and J. Sun, "Single path one-shot neural architecture search with uniform sampling," 2020

RELATED WORK: BLOCK-WISE NAS

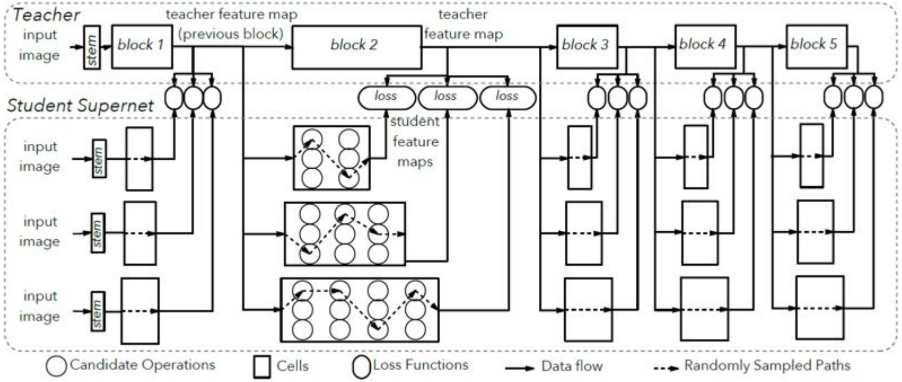
- Block-wise NAS
  - Divide the supernet into several blocks in term of depth and optimize these blocks in isolation.

$$\mathcal{N} = \mathcal{N}_N \cdots \mathcal{N}_{i+1} \circ \mathcal{N}_i \cdots \circ \mathcal{N}_1 \qquad (1)$$

  - The size of search space in each block is exponentially reduced following Eqn. (2), where C denotes number of candidate operations in each layer, d_i denotes the depth of i-th block.

$$Reduction\ rate = C^{d_i} \Big/ \left(\prod_{i=0}^{N} C^{d_i}\right) \qquad (2)$$

  - All candidates in every block are well optimized, thus improving the ranking accuracy.
  - Fails to address quantization



[6]

[6] C. Li, J. Peng, L. Yuan, G. Wang, X. Liang, L. Lin, and X. Chang, “Blockwisely supervised neural architecture search with knowledge distillation,” 2020.

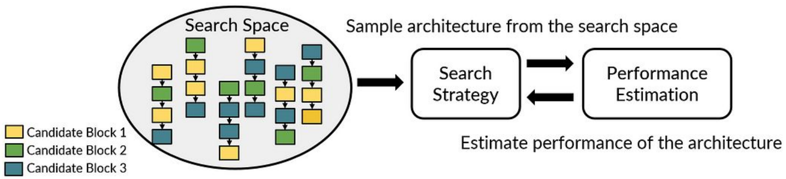
INTRODUCTION: MOTIVATIONS

- The **keys** to effective deployment of DNN models on edge devices:

1. Low inference latency
2. Small memory footprint
3. High accuracy

The best full-precision architecture is **not necessarily** the optimal one after quantization. [9]

Neural Architecture Search (NAS) → **Quantization**



[3]

[2]

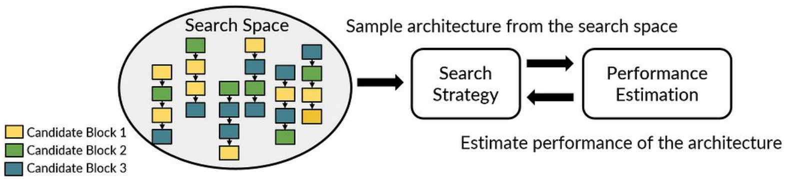
[9] T. Wang, K. Wang, H. Cai, J. Lin, Z. Liu, H. Wang, Y. Lin, and S. Han, "Apq: Joint search for network architecture, pruning and quantization policy," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 2020.

INTRODUCTION: MOTIVATIONS

- The **keys** to effective deployment of DNN models on edge devices:

1. Low inference latency
2. Small memory footprint
3. High accuracy

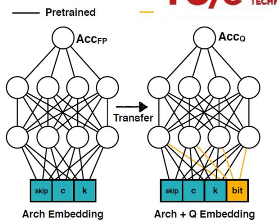
Neural Architecture Search (NAS) ⟷ **Joint** ⟷ **Quantization**



[3]

[2]

RELATED WORK: JOINT QUANTIZATION AND NEURAL ARCHITECTURE SEARCH



[9]

- Common approaches such as APQ [9] and QFA [10]
  - Once-for-all supernet-based NAS which builds an accuracy predictor for quantized performance
- Requires several thousand GPU hours for training
- Fails to scale towards large-scale tasks

 With block-wise NAS, the total search cost can potentially be reduced to **tens of** GPU hours on large-scale tasks, e.g., semantic segmentation.

[9] T. Wang, K. Wang, H. Cai, J. Lin, Z. Liu, H. Wang, Y. Lin, and S. Han, “Apq: Joint search for network architecture, pruning and quantization policy,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 2020.

[10] H. Bai, M. Cao, P. Huang, and J. Shan, “Batchquant: Quantized-for-all architecture search with robust quantizer,” 2021.

CONTRIBUTIONS

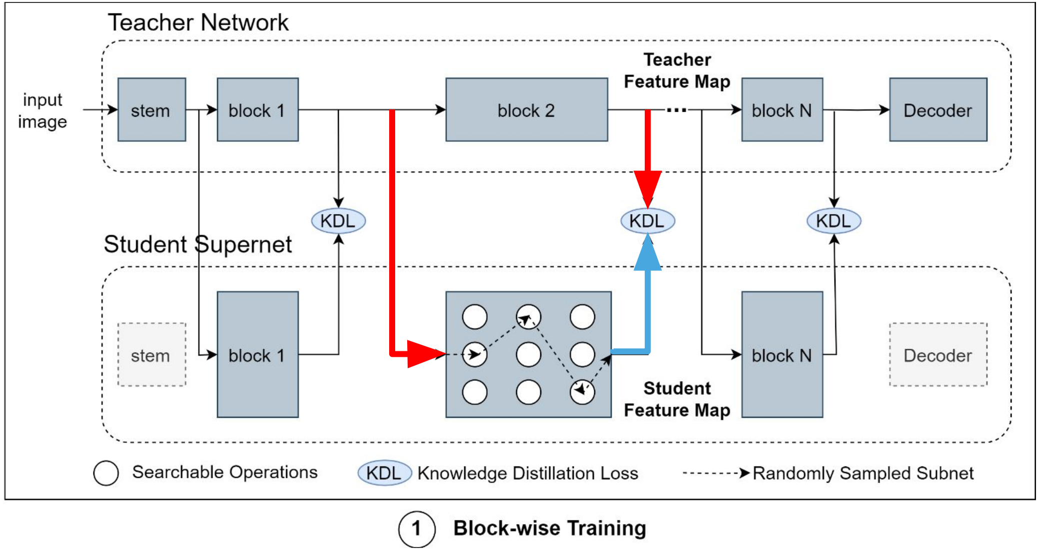
1. Quantization-Aware Block-Wise NAS (**QA-BWNAS**)
   - A simple yet effective approach
2. Automate the design of highly accurate and efficient homogeneous (e.g., INT8) and FB-MP models.
3. Suitable for scaling QA-NAS up to large-scale and compute-intensive tasks.
4. Optimization on search strategy, reducing the search cost from hours to seconds.

OVERVIEW

- Introduction & Related Work
- Quantization-Aware Block-wise NAS (Homogeneous)
- Quantization-Aware Block-wise NAS (FB-MP)
- Conclusions

METHOD: BLOCK-WISE SUPERNET TRAINING VIA KNOWLEDGE DISTILLATION

- Feature-based knowledge distillation
  - Blocks in the student supernet are trained in isolation
    - Input: the previous feature map of a trained teacher model
    - Knowledge Distillation (KD) loss: noise-to-signal-power ratio (NSR)
  - NSR loss of **each** subnet can be evaluated as a proxy of ground truth performance.



Adopted from [6]

$$\mathcal{L}_{NSR}(\mathcal{Y}_n, \hat{\mathcal{Y}}_n) = \frac{1}{C} \sum_{c=0}^{C} \frac{\|\mathcal{Y}_{n,c} - \hat{\mathcal{Y}}_{n,c}\|^2}{\sigma_{n,c}^2} \quad (1)$$

[13]

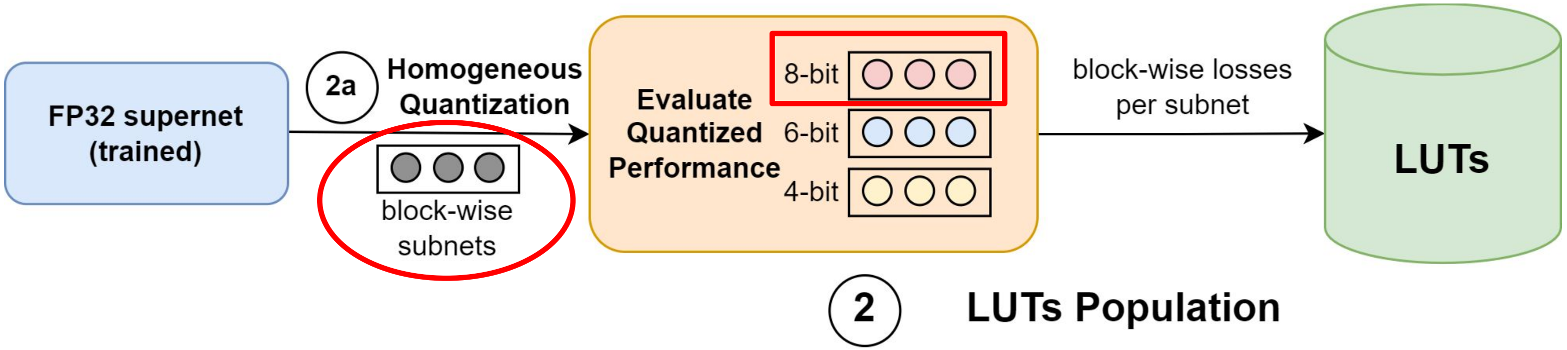
[6] C. Li, J. Peng, L. Yuan, G. Wang, X. Liang, L. Lin, and X. Chang, "Blockwisely supervised neural architecture search with knowledge distillation," 2020.

[13] B. Moons, P. Noorzad, A. Skliar, G. Mariani, D. Mehta, C. Lott, and T. Blankevoort, "Distilling optimal neural networks: Rapid search in diverse spaces," 2021.

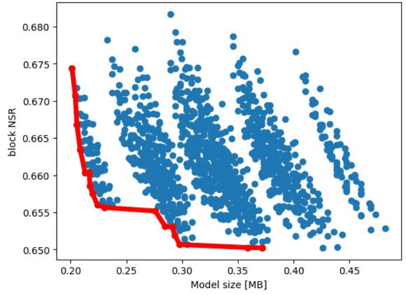
METHOD: NSR LUT POPULATION (HOMOGENEOUS)

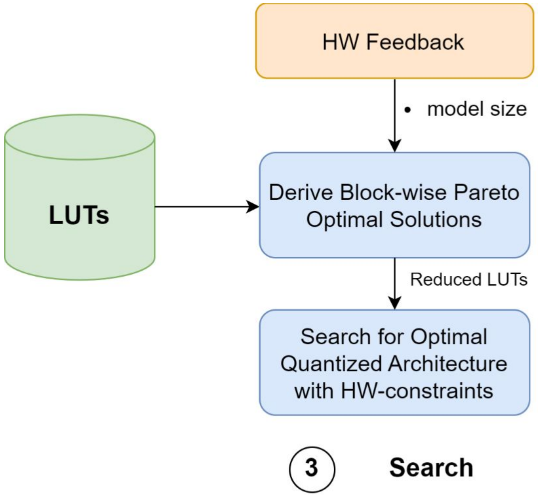
- ***How*** to efficiently introduce quantization in block-wise NAS?
  - Quantize each subnet from the FP32 supernet
  - Evaluate quantized subnets to populate NSR LUTs

$$\mathcal{L}_{NSR}(\mathcal{Y}_n, \hat{\mathcal{Y}_n}) = \frac{1}{C}\sum_{c=0}^{C}\frac{\|\mathcal{Y}_{n,c} - \hat{\mathcal{Y}}_{n,c}\|^2}{\sigma_{n,c}^2} \qquad (1)$$

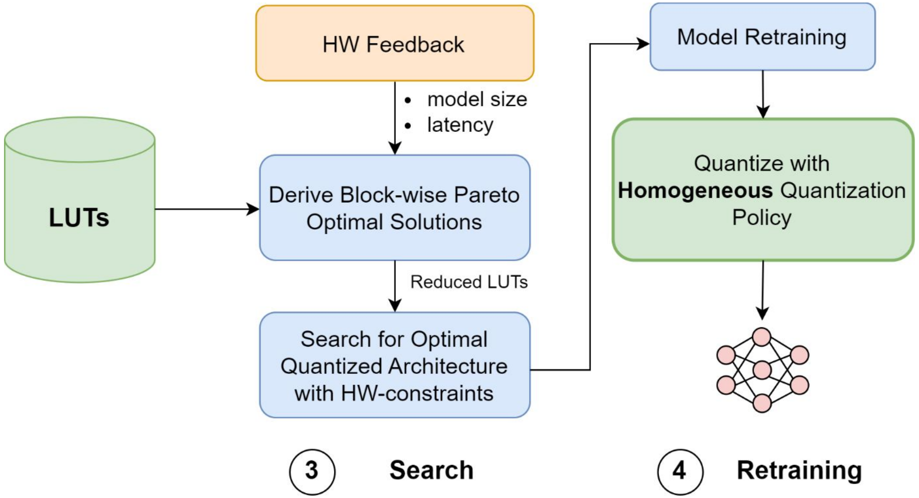


- **Search Strategy**
  - DNA's traversal search [6]:
    - Subtly visits all possible candidates in the search space
    - The search can take approximately 1 hour for one optimal model
  - Our optimization
    - HW-related secondary objectives
      - model size
      - inference latency
    - Searches only within Pareto optimal candidates in each block
    - e.g., Reduces #candidates from 1296 to 17 (4-layer block)
    - Search cost: from several **hours** to a few **seconds**





- **Model Retraining**
  - Retrain the searched architecture to convergence.
  - Quantize the trained model to obtain its low-precision performance.



IMPLEMENTATION DETAILS

- *Dataset:* **Cityscapes**

- *Teacher model:* DeepLabv3 [12]
  - *SOTA model, the encoder is MobileNet V2.*

- *Searchable architectures*
  - MBConv block
  - Kernel size: {3, 5, 7}
  - Expansion ratios: {3, 6}

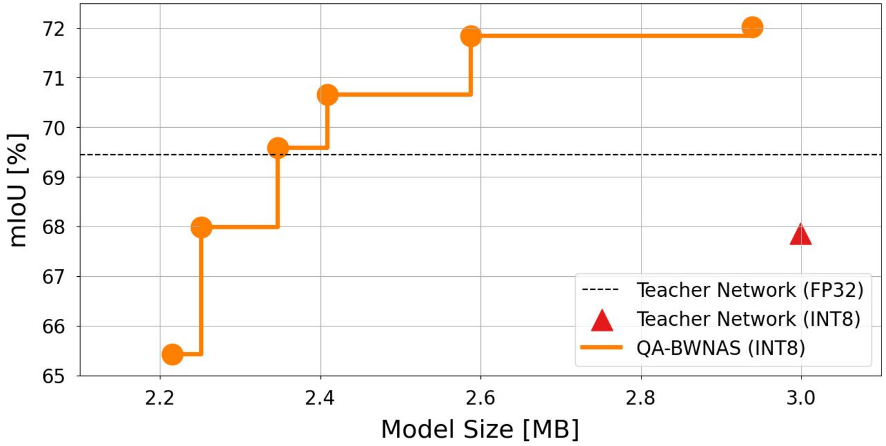
- *Bit widths*
  - Homogeneous quantization: {8}

TABLE I
SUPERNET DESIGN AND BLOCK DETAILS. "L#" AND "CH#" REPRESENT THE NUMBER OF LAYERS AND CHANNELS OF EACH BLOCK.

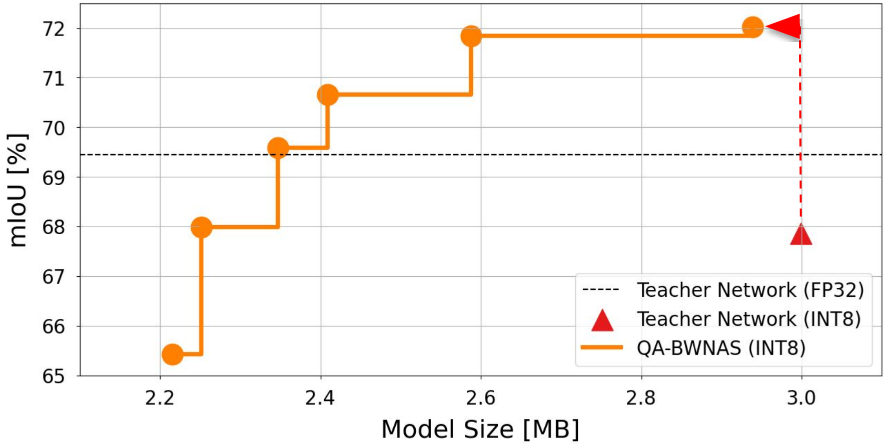
model		teacher		student supernet	
block	stride	L#	CH#	L#	CH#
1	2	2	24	3	24
2	2	3	32	3	32
3	1	4	64	4	64
4	1	3	96	4	96
5	1	3	160	3	160
6	1	1	320	1	320

[12] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," 2017.

- **Results:**
  - QA-BWNAS (homogeneous) yields a Pareto front of solutions, which substantially outperform the teacher network.

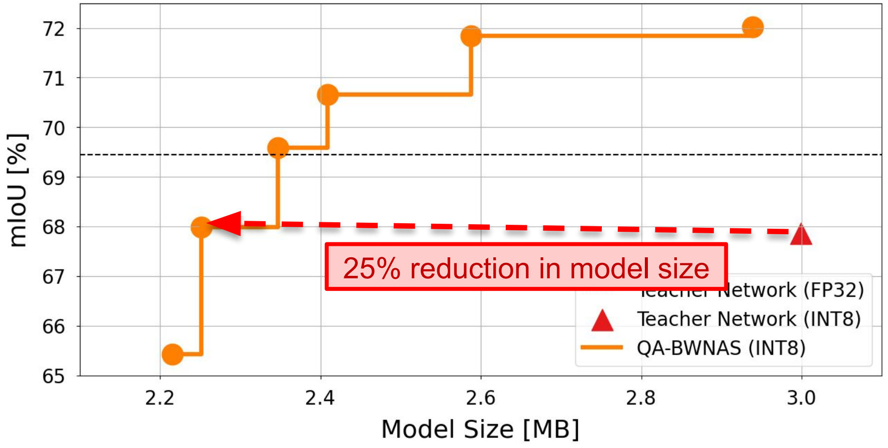


- **Results:**
  - QA-BWNAS (homogeneous) yields a Pareto front of solutions, which substantially outperform the teacher network.
  - 4.2 pp. higher mIoU



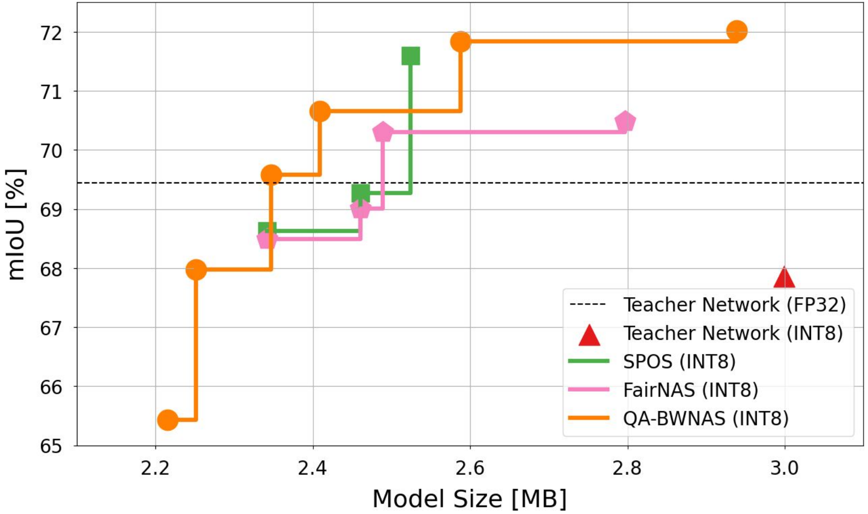
RESULTS: HOMOGENEOUS QUANTIZATION (INT8 & MODEL SIZE)

- **Results:**
  - QA-BWNAS (homogeneous) yields a Pareto front of solutions, which substantially outperform the teacher network.
  - 4.2 pp. higher mIoU
  - 25% smaller model size



RESULTS: HOMOGENEOUS QUANTIZATION (INT8 & MODEL SIZE)

- **Results:**
  - Two SOTA weight-sharing NAS methods
    - FairNAS
    - SPOS
  - Outperform them with little extra compute cost.



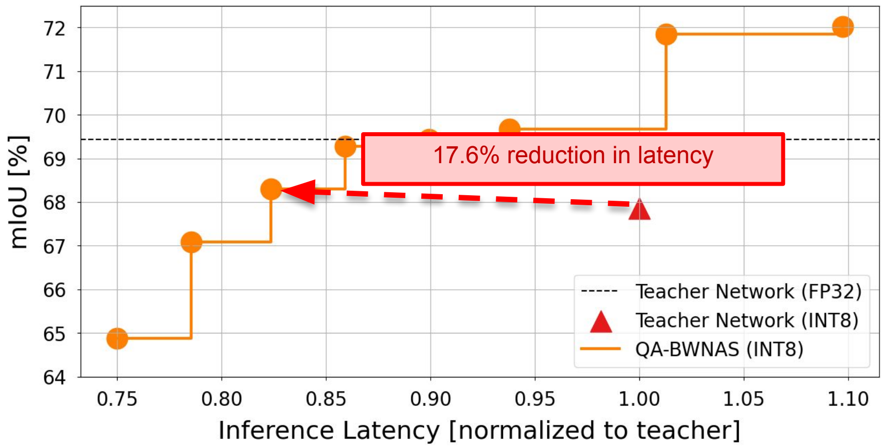
Compute Effort (GPU hours)

Method	Train	LUT Population	Search
QA-BWNAS (INT8)	4.05	14.87	0
FairNAS (INT8)	3.5	-	7.5
SPOS (INT8)	4.5	-	7.5

GPU: NVIDIA RTX8000

RESULTS: HOMOGENEOUS QUANTIZATION (INT8 & INFERENCE LATENCY)

- **Results:**
  - A Pareto front of solutions on i.MX8M Plus.
  - Reduction in inference latency.
    - 17.6% lower



- **Findings:**
  - Accommodate various secondary objectives.

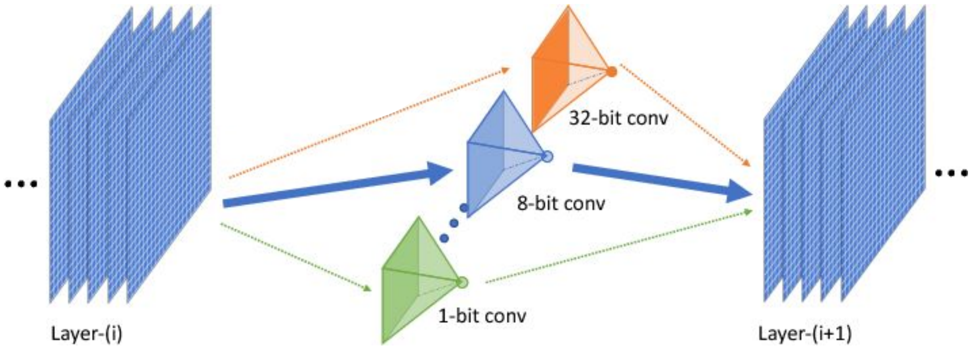
OVERVIEW

- Introduction & Related Work
- Quantization-Aware Block-wise NAS (Homogeneous)
- Quantization-Aware Block-wise NAS (FB-MP)
- Conclusions

METHOD: QUANTIZATION-AWARE BLOCK-WISE NAS (FB-MP)

- Layers/Blocks in DNNs have different sensitivities to quantization. [7]

- Few-Bit Mixed-Precision (FB-MP) quantization
  - Improve model efficiency without causing considerable performance degradation.



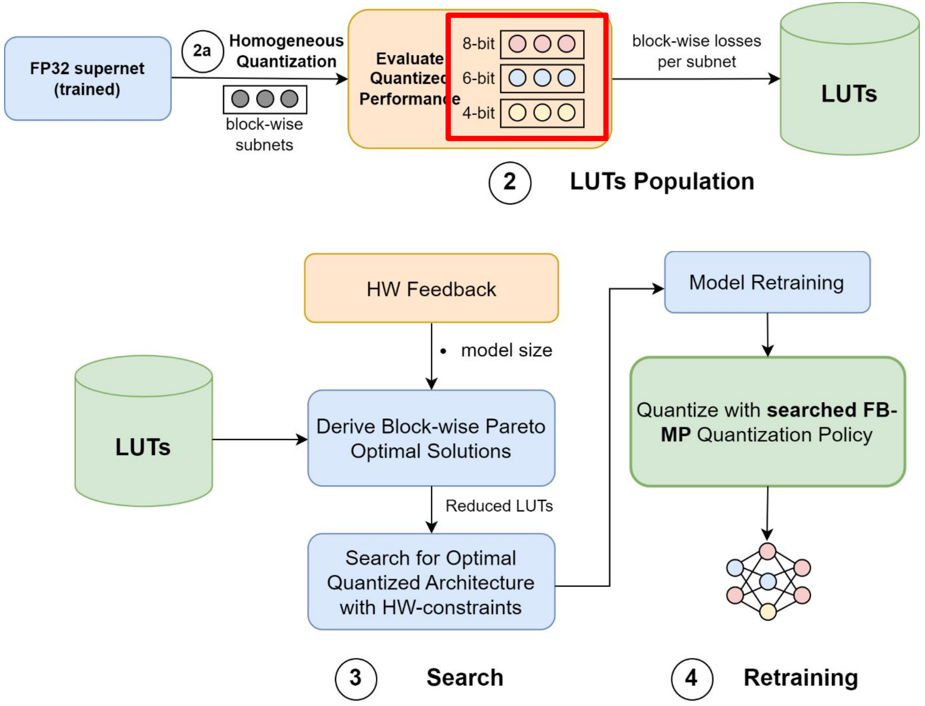
[11]

[7] A. Gholami, S. Kim, Z. Dong, Z. Yao, M. W. Mahoney, and K. Keutzer, “A survey of quantization methods for efficient neural network inference,” 2021.

[11] B. Wu, et al., Mixed Precision Quantization of ConvNets via Differentiable Neural Architecture Search, ICLR 2019.

QA-BWNAS (FB-MP):

- Quantize each subnet with different bit widths
- Concatenate NSR LUTs for searching
- Retrain the found model and quantize it with searched FB-MP policy



IMPLEMENTATION DETAILS

- *Dataset:* Cityscapes

- *Teacher model:* DeepLabv3 [12]
  - *SOTA model, the encoder is MobileNet V2.*

- *Searchable architectures*
  - MBConv block
  - Kernel size: {3, 5, 7}
  - Expansion ratios: {3, 6}

- *Searchable bit-widths*
  - **FB-MP quantization: {4, 6, 8}**

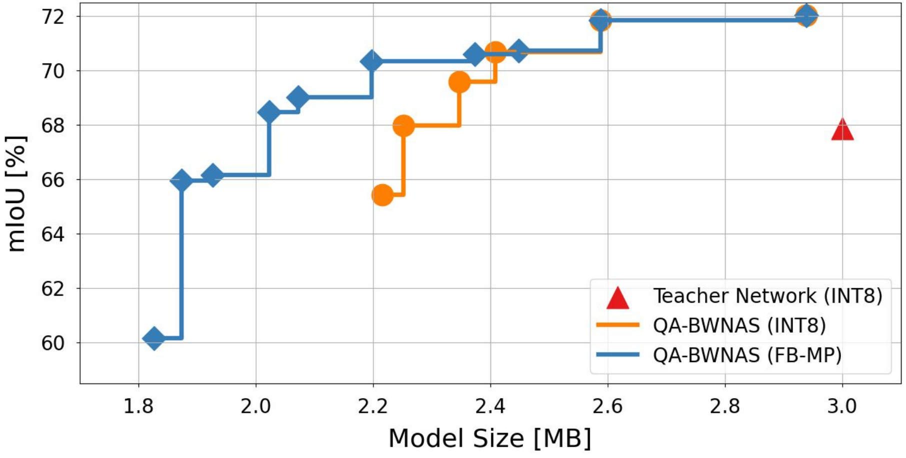
TABLE I
SUPERNET DESIGN AND BLOCK DETAILS. "L#" AND "CH#" REPRESENT THE NUMBER OF LAYERS AND CHANNELS OF EACH BLOCK.

model		teacher		student supernet	
block	stride	L#	CH#	L#	CH#
1	2	2	24	3	24
2	2	3	32	3	32
3	1	4	64	4	64
4	1	3	96	4	96
5	1	3	160	3	160
6	1	1	320	1	320

[12] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," 2017.

RESULTS: FB-MP QUANTIZATION (MODEL SIZE)

- **Results:**
  - Outperform INT8 solutions in terms of mIoU and model size.



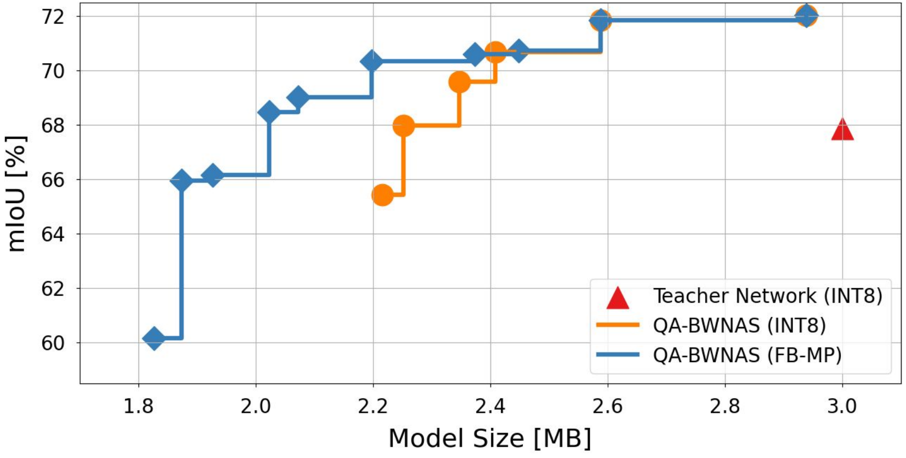
RESULTS: FB-MP QUANTIZATION (MODEL SIZE)

- **Results:**
  - Outperform INT8 solutions in terms of mIoU and model size.
  - Relatively minor increase in compute efforts.

Compute Effort (GPU hours)

Method	Train	LUT Population	Search
QA-BWNAS (FP-MP)±	4.05	44.61	14 × N
QA-BWNAS (FP-MP)	4.05	44.61	0
QA-BWNAS (INT8)	4.05	14.87	0

GPU: NVIDIA RTX8000



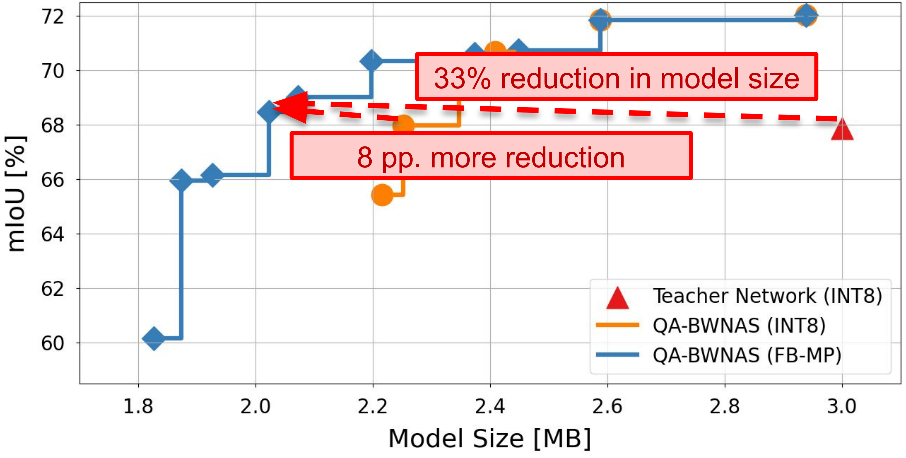
RESULTS: FB-MP QUANTIZATION (MODEL SIZE)

- **Results:**
  - Outperform INT8 solutions in terms of mIoU and model size.
  - Relatively minor increase in compute efforts.
  - 33% smaller model size
    - 8 pp. more reduction

Compute Effort (GPU hours)

Method	Train	LUT Population	Search
QA-BWNAS (FP-MP)±	4.05	44.61	14 × N
QA-BWNAS (FP-MP)	4.05	44.61	0
QA-BWNAS (INT8)	4.05	14.87	0

GPU: NVIDIA RTX8000



OVERVIEW

- Introduction & Related Work
- Quantization-Aware Block-wise NAS (Homogeneous)
- Quantization-Aware Block-wise NAS (Mixed Precision)
- Conclusions

CONCLUSIONS

1. **QA-BWNAS**: A simple yet effective approach.
2. Automate the design of highly accurate and efficient homogeneous (e.g., INT8) and FB-MP models.
3. Suitable for scaling QA-NAS up to large-scale and compute-intensive tasks.
4. Optimization on search strategy, reducing the search cost from hours to seconds.

SECURE CONNECTIONS FOR A SMARTER WORLD

EINDHOVEN UNIVERSITY OF TECHNOLOGY

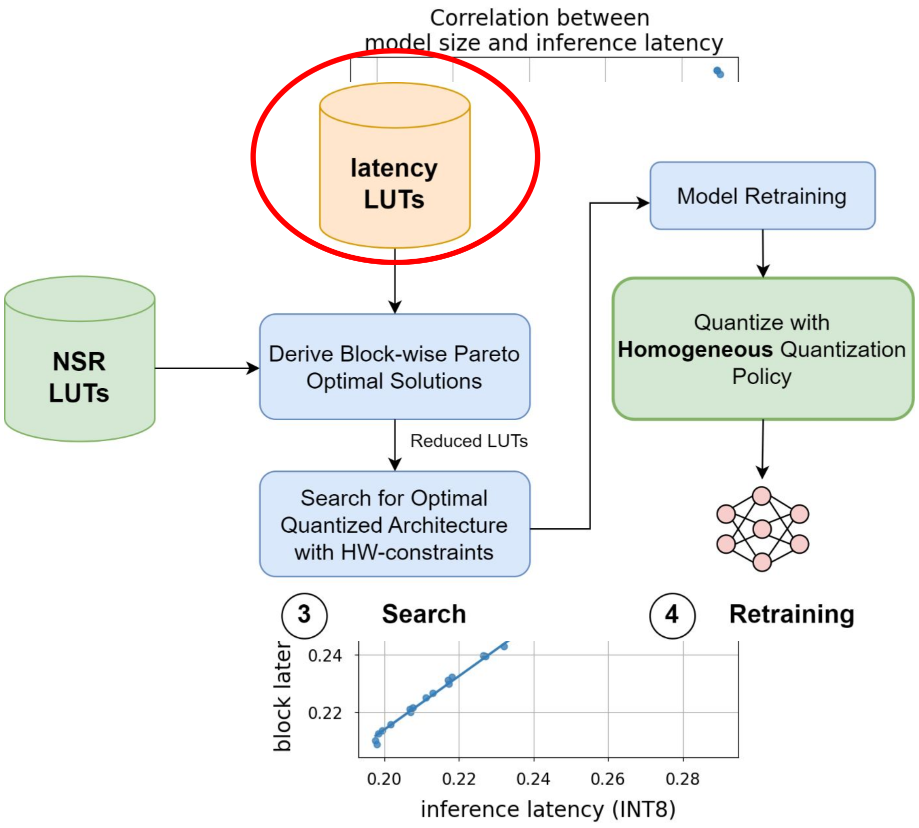
- Backup Slides

METHOD: HOMOGENEOUS QUANTIZATION (INFERENCE LATENCY)

- **Challenge**:
  - Low correlation
  - The best model under model size is likely to be sub-optimal in terms of inference latency.

How to introduce ***latency awareness*** into block-wise NAS?

- **Solution**:
  - Estimate by block latency addition
  - Populate LUTs for quantized subnet latency
  - High correlation (Kendall-Tau = 0.96809)

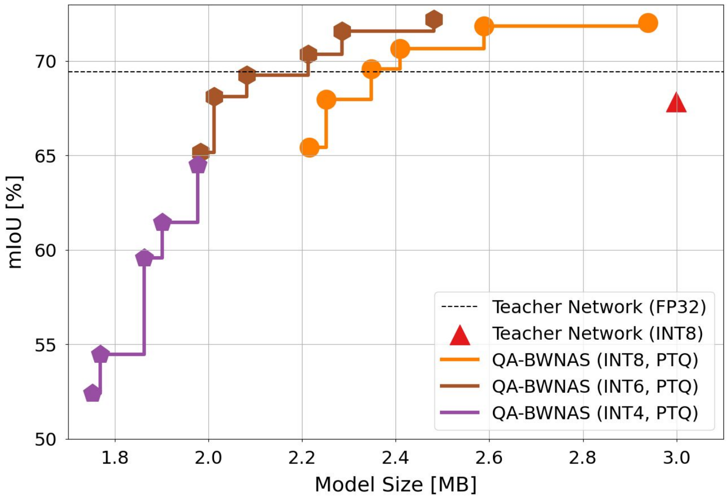


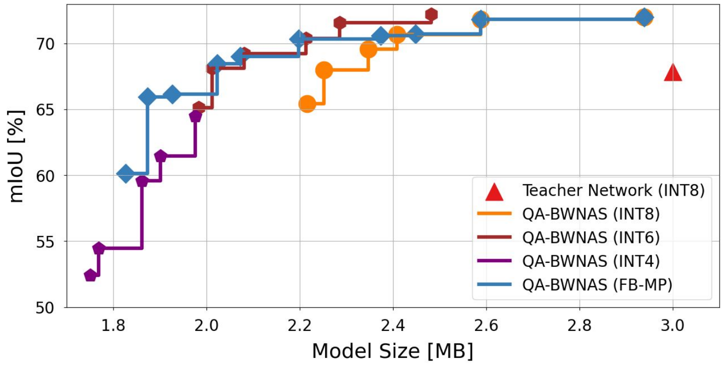
ABLATION STUDY: HOMOGENEOUS LOWER-BIT QUANTIZATION

- Homogeneous QA-BWNAS for lower precision
  - INT6
  - INT4

Observations:

- Reduce model size while retaining task accuracy.





EVIDENCE OF SUB-OPTIMAL ESTIMATION OF NSR ADDITION

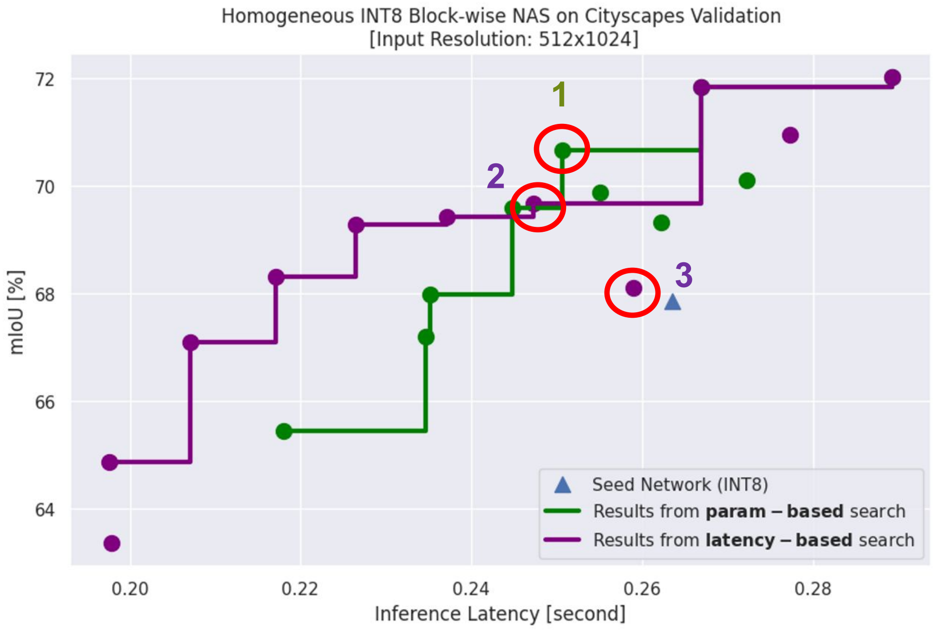
- Limitations of our performance estimation strategy via LUTs:
  - <u>Sub-optimal</u> performance estimation. The correlation between NSR sum and final accuracy is sub-optimal.

For example:

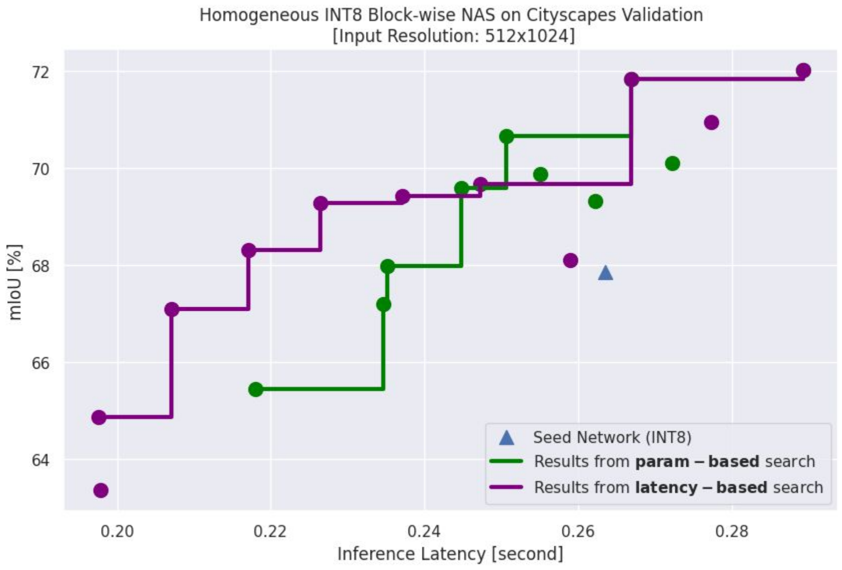
Green 1: 3.640070 (mIoU: 70.66)

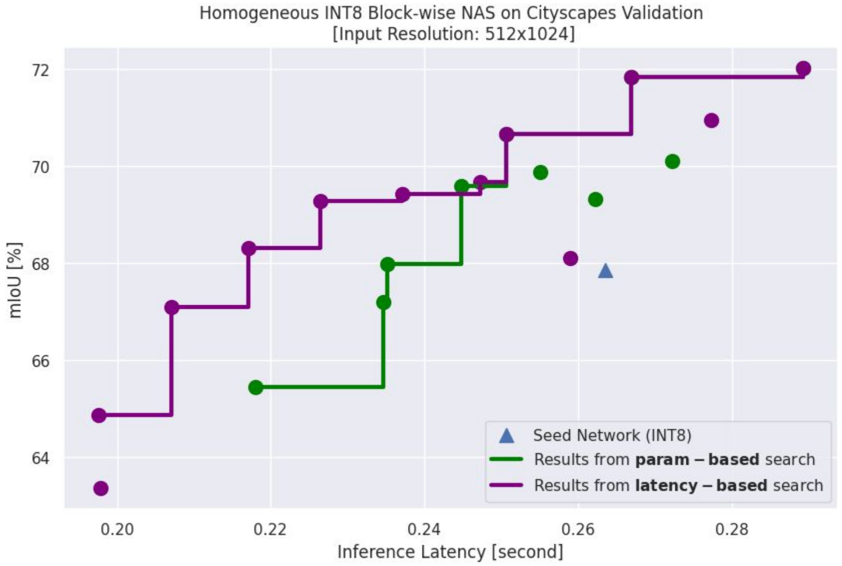
Purple 2: 3.6424480245 (mIoU: 69.67)

Purple 3: 3.6353230685 (mIoU: 68.11)



EVIDENCE OF SUB-OPTIMAL ESTIMATION OF NSR ADDITION





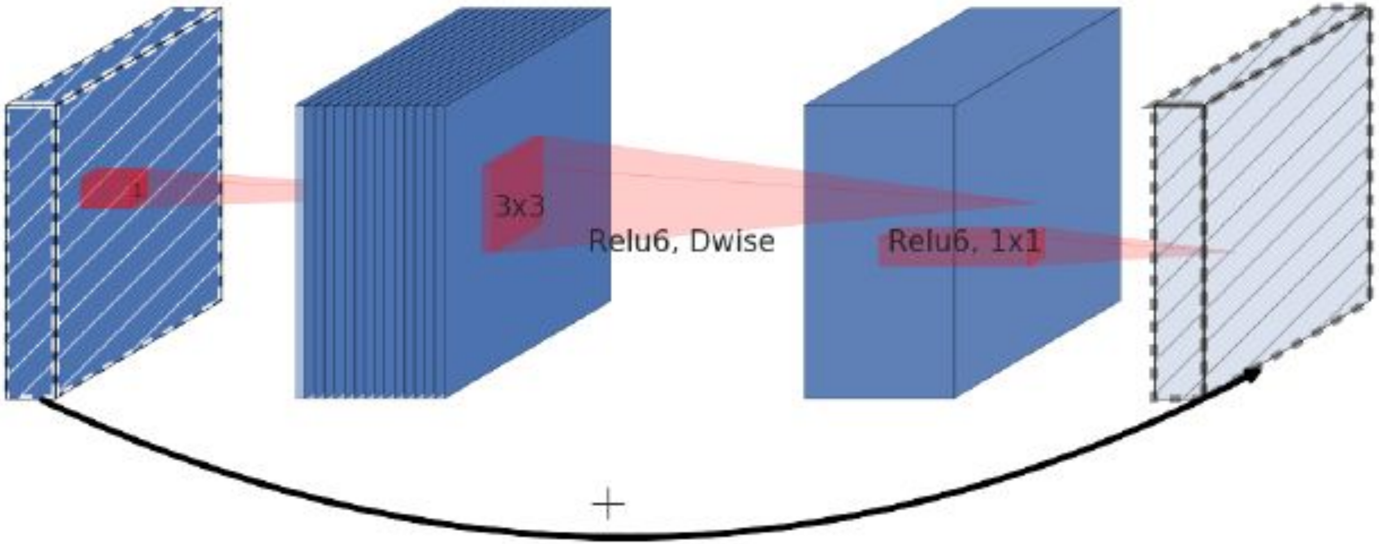
FUTURE WORK

- Direction 1:
  - Accuracy predictor for quantized performance prediction

- Direction 2:
  - Validate its generalizability.
    - Other large-scale/low-scale tasks
    - Other datasets
    - Other networks
    - Different teacher models

[9] T. Wang, K. Wang, H. Cai, J. Lin, Z. Liu, H. Wang, Y. Lin, and S. Han, “Apq: Joint search for network architecture, pruning and quantization policy,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 2020.
[10] H. Bai, M. Cao, P. Huang, and J. Shan, “Batchquant: Quantized-for-all architecture search with robust quantizer,” 2021.
[13] B. Moons, P. Noorzad, A. Skliar, G. Mariani, D. Mehta, C. Lott, and T. Blankevoort, “Distilling optimal neural networks: Rapid search in diverse spaces,” 2021.

Inverted residual block



QUANTIZATION

B. Quantization

The quantization function typically used to map full-precision neural weights and activations to a lower precision is defined as follows [13]:

$$Q(r) = \text{Int}(r/S) - Z \quad (6)$$

where Q is the quantization operator, r is the input tensor (weight or activation), S is the scaling factor, and Z is an integer zero point.

The scaling factor S is mainly to divide the range of a given input tensor r into several partitions by:

$$S = \frac{\beta - \alpha}{2^b - 1} \quad (7)$$

where $[\alpha, \beta]$ denotes the clipping range which is a bounded range used to clip the input values, b is the target quantization bit-width.

The process of selecting the clipping range is called *calibration*. Min-Max is a popular choice to decide the values of α and β, where $\alpha = r_{min}$ and $\beta = r_{max}$. In our work, we apply per-channel Min-Max to choose the clipping range in the calibration process.

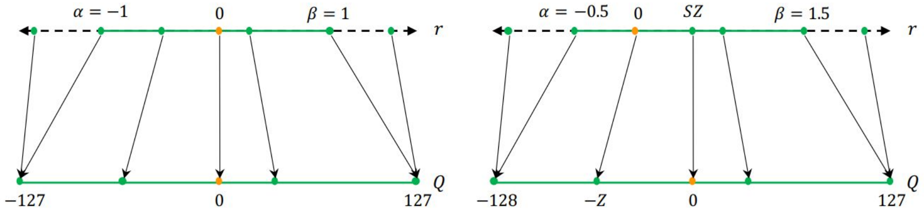


Figure 2: *Illustration of symmetric quantization and asymmetric quantization. Symmetric quantization with restricted range maps real values to [-127, 127], and full range maps to [-128, 127] for 8-bit quantization.*

IMPLEMENTATION DETAILS

- *Dataset:* Cityscapes

- *Teacher model:* DeepLabv3 [12]
  - *SOTA model, the encoder is MobileNet V2.*

- *Searchable architectures*
  - Kernel size of MBConv: {3, 5, 7}
  - Expansion rates: {3, 6}

- *Searchable bit-widths*
  - Homogeneous quantization: {8}
  - Mixed-precision quantization: {4, 6, 8}

TABLE I
SUPERNET DESIGN AND BLOCK DETAILS. "L#" AND "CH#" REPRESENT THE NUMBER OF LAYERS AND CHANNELS OF EACH BLOCK.

model		teacher		student supernet	
block	stride	L#	CH#	L#	CH#
1	2	2	24	3	24
2	2	3	32	3	32
3	1	4	64	4	64
4	1	3	96	4	96
5	1	3	160	3	160
6	1	1	320	1	320

Retraining hyperparameters	
Scheduler	Polynomial
Batch size	8
Learning rate	0.01
Optimizer	SGD with momentum = 0.9
Iterations	80K

Supernet training hyperparameters	
Scheduler	Polynomial
Batch size	8
Learning rate	[0.002, 0.005, 0.005, 0.005, 0.005, 0.002]
Optimizer	SGD with momentum = 0.9
Iterations	[13334, 13334, 13334, 13334, 13334, 13334]