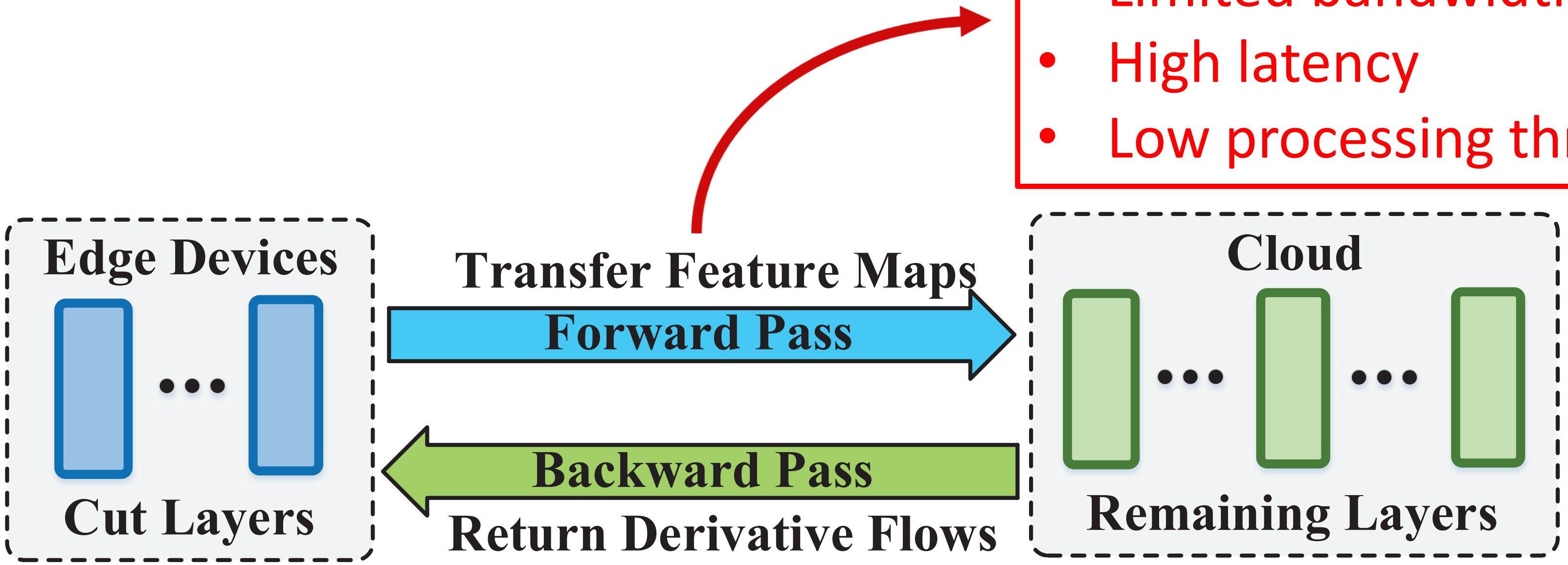
Hierarchical Channel-spatial Encoding for Communication-efficient Collaborative Learning

Qihua Zhou[1], Song Guo[1], Yi Liu[1], Jie Zhang[1],
Jiewei Zhang[1], Tao Guo[1], Zhenda Xu[1], Xun Liu[1], Zhihao Qu[2]

[1]The Hong Kong Polytechnic University, [2]Hohai University

Rise of Collaborative Learning (CL)



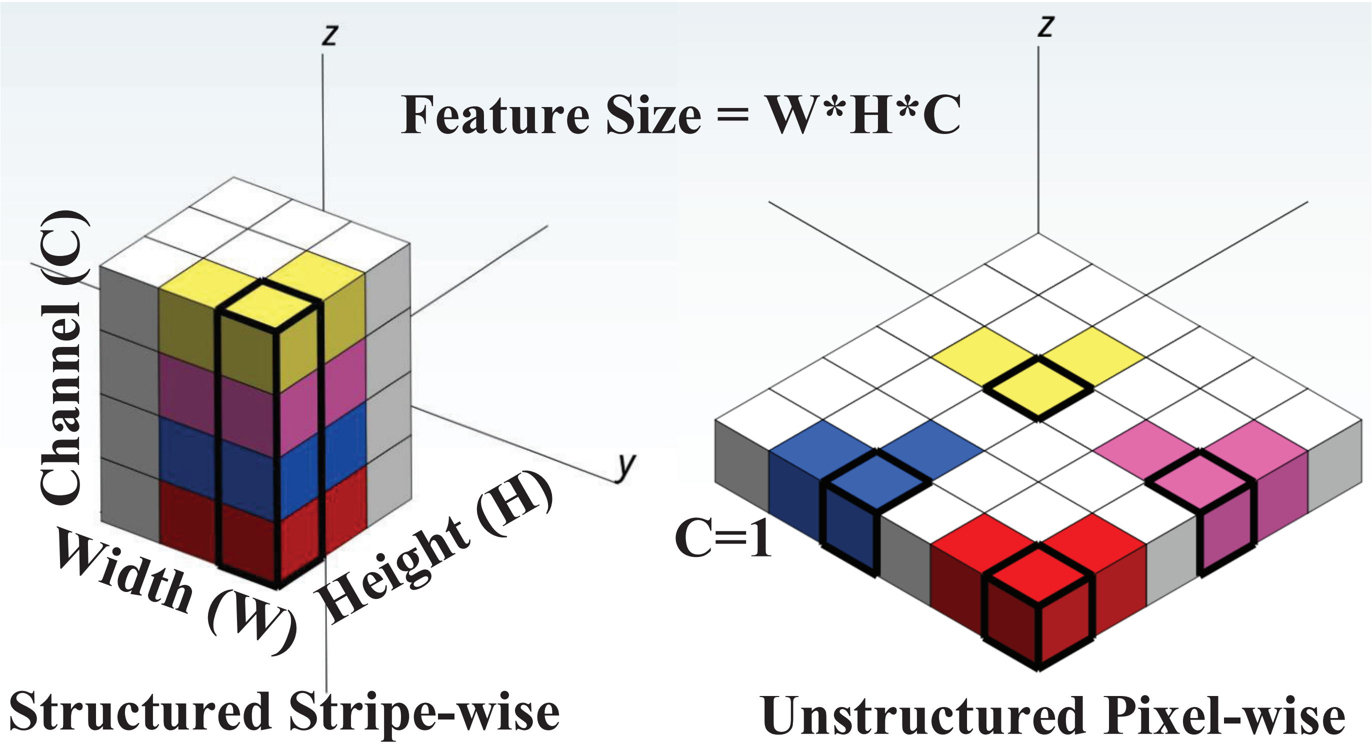
Bottlenecks:
- Huge traffic overhead
- Limited bandwidth
- High latency
- Low processing throughput

Question: how to eliminate the performance bottleneck?
Solution: improve communication efficiency via latent feature encoding.

Limitations of Conventional Encoding Methods

Inspection of previous methods:

- Compress features at pixel level (spatial-wise).
- Ignore the characteristics of feature structure (channel-wise).
- ➔ Why not conduct vector encoding (stripe-wise) for higher compression ratios?



Characteristics of CNN Latent Feature

Observation:

- Output channels generate quite different features when the corresponding filters are orthogonal to each other.

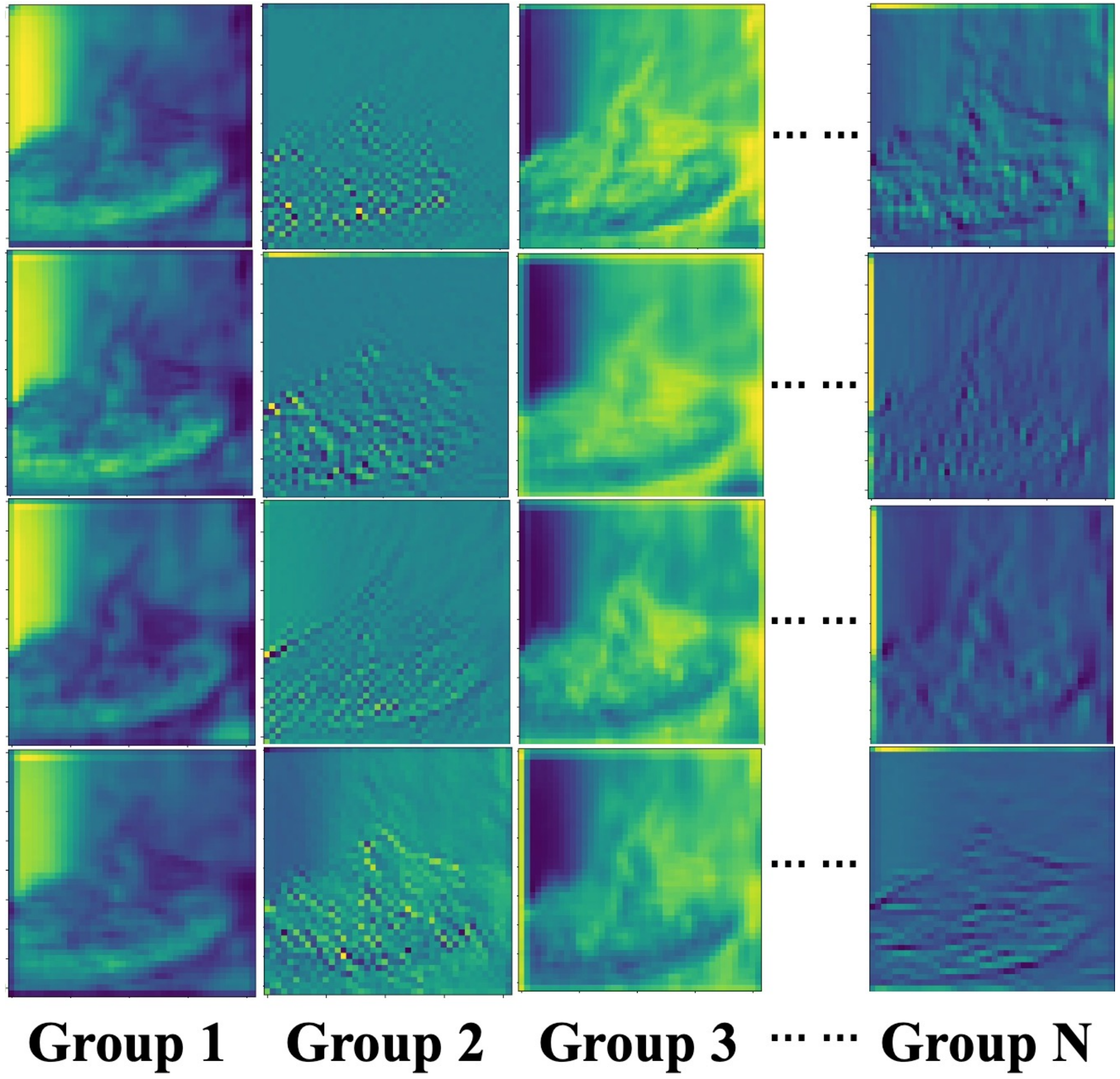
Challenge:

- Simply adopting product or vector quantization along channel dimension does not work well.

Inspiration:

- Grouping the feature maps based on their channel-level similarity can better capture the feature redundancy.

Visualization of Latent Features:

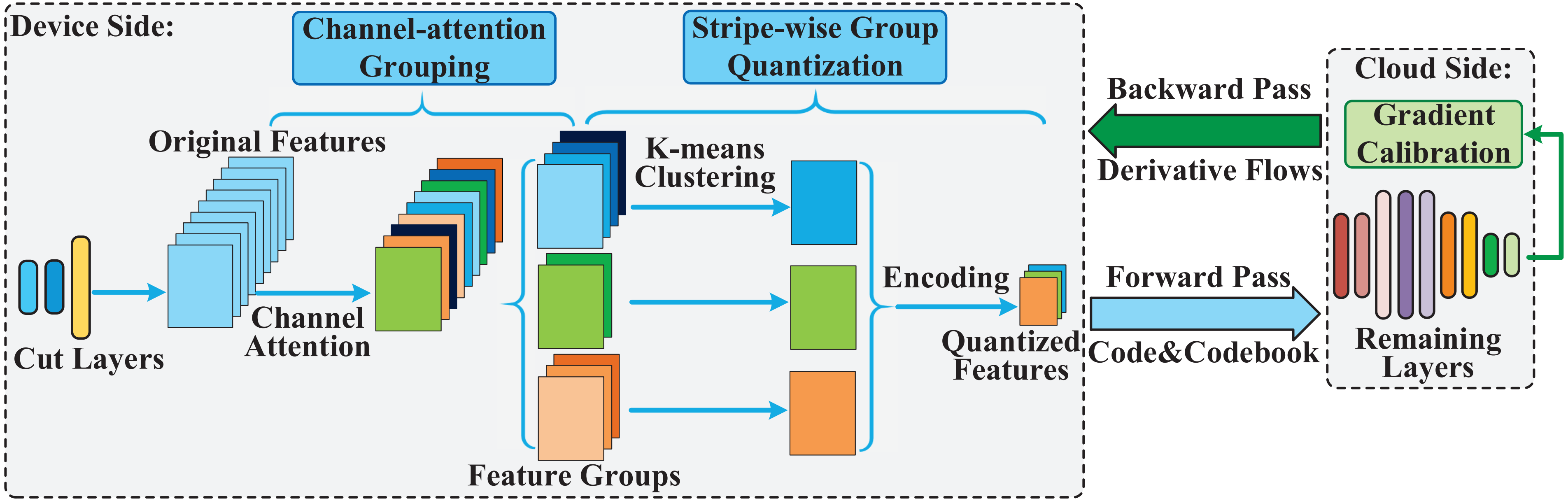


Bridge Gap: Hierarchical Channel-spatial Encoding

Key:

- Capturing such channel-dimension structured information is the key to fundamentally compress feature size.
- It is often omitted by conventional quantization methods designed for parameters, activations or gradients.
- As to each group, we need to find a collection of representative pixels, each of which can replace other pixels similar to it.

Framework Overview:



Core Steps of Strip-wise Group Quantization (SGQ)

Two-step Compression:
- Feature Discretization.
- Pixel Encoding.

$$S_{SGQ} = \sum_{i=1}^{G} (\underbrace{n \cdot WH}_{feature} + \underbrace{32 \cdot 2^n \cdot C_i}_{codebook}),$$

Accuracy Preservation:
- Proper grouping: channel-attention grouping block.
- Guarantee model convergence: gradient calibration.

$$S_{UQ} = \underbrace{n \cdot WH \cdot \sum_{i=1}^{G} C_i}_{feature} + \underbrace{32 \cdot 2^n}_{codebook}$$

Traffic Analysis:
- Achieve a much higher compression ratio over existing methods.

$$\frac{n \cdot WH}{2^{n+5}} > \underbrace{\frac{\sum_{i=1}^{G} C_i - 1}{\sum_{i=1}^{G} C_i - G}}_{\approx 1}$$

Theoretical Convergence Analysis

Convergence order:

$$\frac{1}{T}\sum_{t=0}^{T-1} E\|\nabla f(\boldsymbol{w}_t)\|_2^2 \preceq O(\frac{1}{\sqrt{NT}})$$

Effectiveness :

- SGQ holds the same order of convergence rate as the non-quantized distributed SGD algorithm and exhibits the linear speedup property with respect to the number of devices.

Summary:

- Theoretical results demonstrate that our proposed algorithm is communication-efficient and scalable for the collaborative learning environment.

Evaluation Setup

Platforms

- HUAWEI Atlas 200 DK: Ascend 310 AI processor
- NVIDIA Jetson Nano: Quad-core ARM A57 @ 1.43 GHz
- Remote server: NVIDIA RTX 2080Ti server through 10GbE network

Benchmarks

- Model: AlexNet, VGG-11, ResNet-18/34, ShuffleNet-V2-1.0x/0.5x, MobileNet-V1
- Dataset: CIFAR-10/100 (CF), Fashion MNIST (FM), mini-ImageNet (MI), ImageNet-1K

Baselines

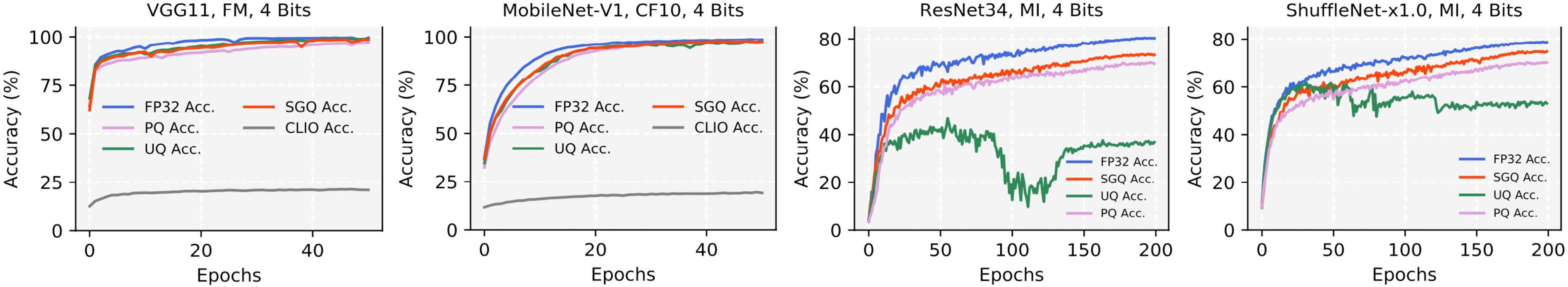
- Vanilla full-precision training (FP32)
- Uniform quantization (UQ)
- Product quantization (PQ)
- Progressive-slicing CLIO (Top-K)

HUAWEI Atlas 200 DK

NVIDIA Jetson Nano

Convergence Results

- **Comparison of convergence curves using different benchmarks and baselines**



- **Summary of average model accuracy (%) using 4-bit compression, compared with FP32**

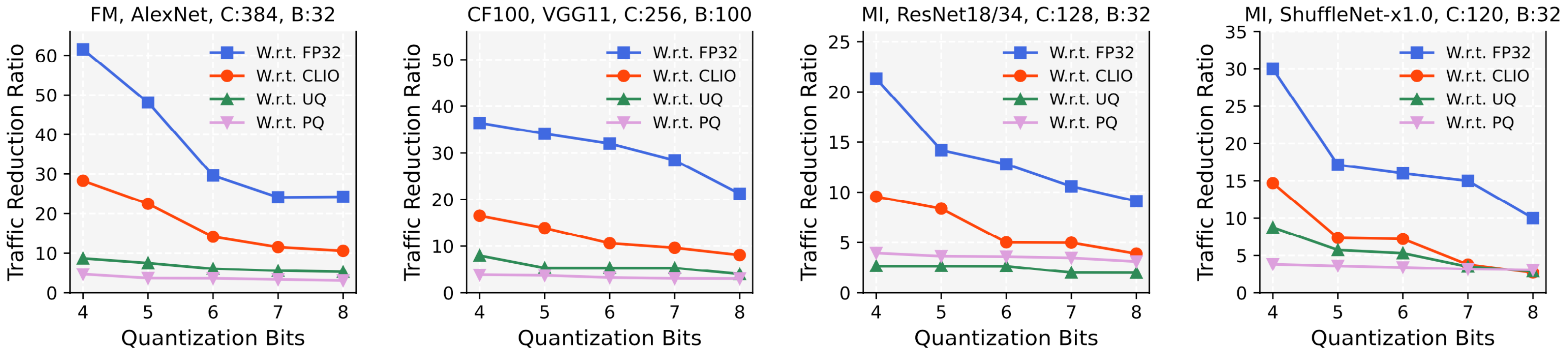
Method	VGG11, FM	MobileNet-V1, CF10	ResNet34, MI	ShuffleNet-x1.0, MI
FP32 (Upper Bound)	97.55	94.74	80.31	78.73
UQ	95.12	92.41	36.89	53.15
PQ	95.94	92.67	69.61	70.16
CLIO	21.02	19.16	13.06	11.10
SGQ	**96.57**	**93.45**	**74.37**	**74.86**

SGQ outperforms other baselines in different training configurations

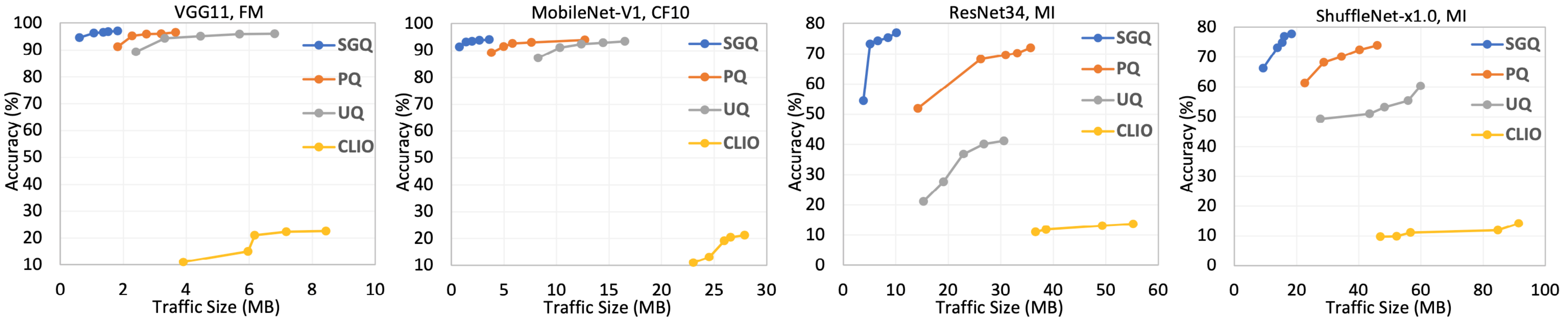
Traffic Saving and Accuracy-size Trade-off

- **Average traffic reduction ratio by using SGQ**

$$\text{Traffic Reduction Ratio} = \frac{\text{Baselines's Traffic}}{\text{SGQ's Traffic}}$$



- **SGQ outperforms existing methods in both model accuracy and traffic size**



Conclusion

SGQ: **Hierarchical Channel-spatial Encoding** for Communication-efficient Collaborative Learning

- **General feature compression method**: effectively leverages the pixel similarity by reorganizing the features into groups based on channel significance.
- **Efficient convergence order**: hold the same convergence order as the Stochastic Gradient Descent method without quantization on feature maps.
- **Scalable collaborative learning framework**: enables model evolution on multiple edge devices and match the requirements of continuous analytics.
- SGQ provides an efficient **accuracy-size trade-off** for collaborative learning applications, while achieving higher **traffic reduction ratio (up to 15.97×)** and higher **image processing speedup (up to 9.22×)** over existing methods.

Thank you!

csqzhou@comp.polyu.edu.hk