Dynamic CNN Accelerator Supporting Efficient Filter Generator with Kernel Enhancement and Online Channel Pruning

Chen Tang, Wenyu Sun, Wenxun Wang, Yongpan Liu

Tsinghua University, Beijing, China

Self Introduction

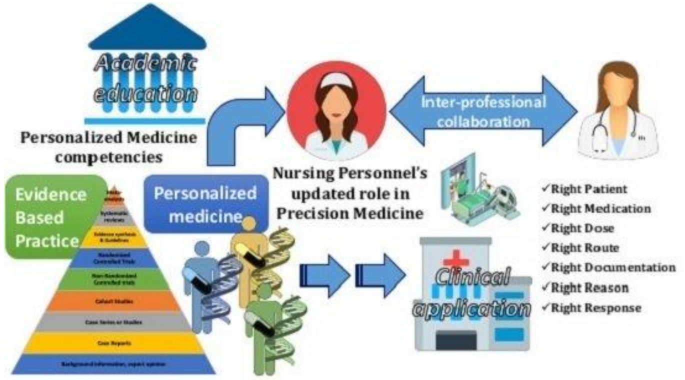
- B.Eng. degree from Tsinghua University, Beijing, China, in 2020
- Ph.D. candidate at Tsinghua University, Beijing, China
- My research interests are in architecture design of accelerators for fast CNN training and inference.
  - Network Pruning
  - Neural architecture search

Outline

- Introduction to Neural Network Sparsity
- Background
  - Dynamic inference
  - Dynamic parameter
- Methods
  - Kernel Enhancement
  - Online Channel Selection
  - Hardware Architecture
- Main Results
- Conclusion

CNN continues to thrive

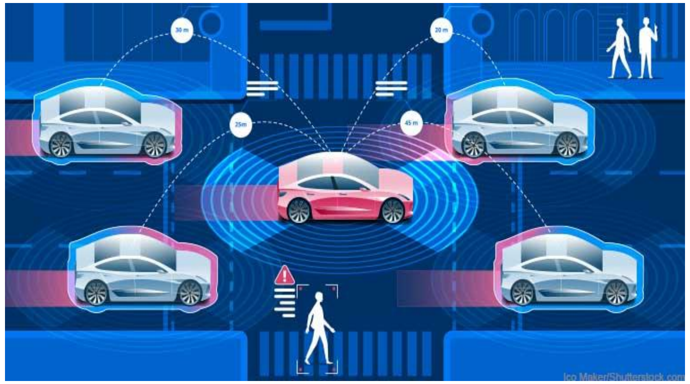
Smart retail



Personalized healthcare

Smart city

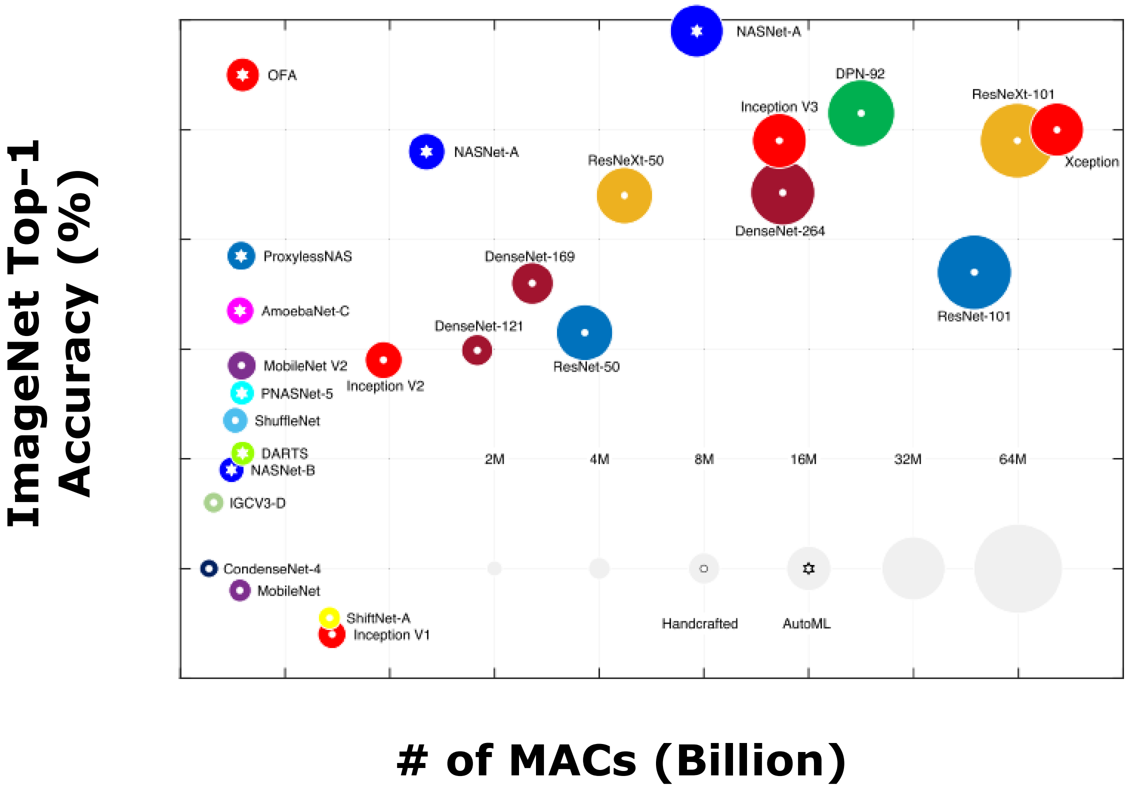
Smart manufacturing



Autonomous driving

Smart teaching

CNN continues to thrive

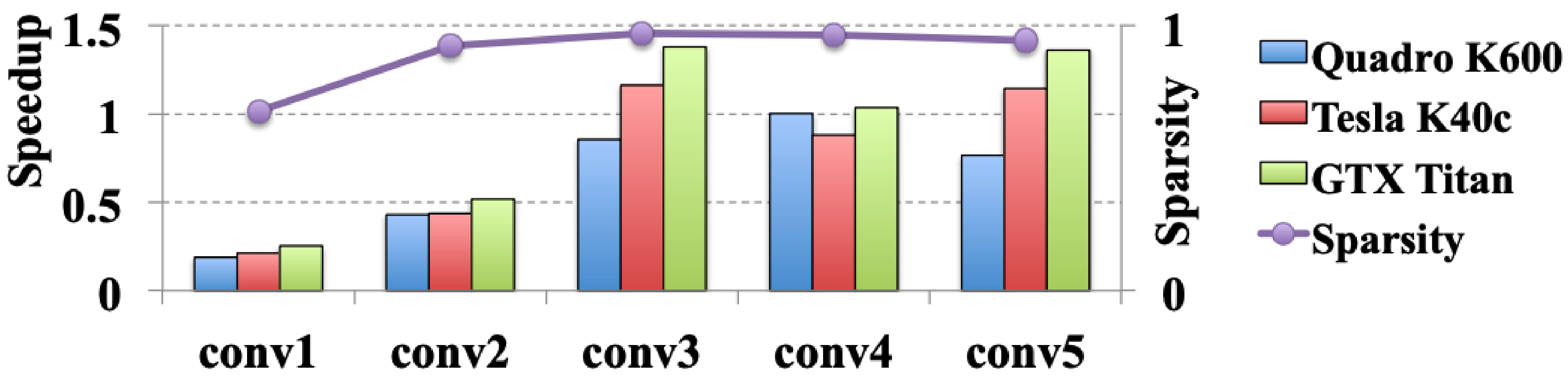


Deng et al., Proc. IEEE, 2020

- Real-time requirement
  - Autonomous driving
  - Image enhancement
  - Video super resolution
  - ……
- Methods
  - Unstructured pruning
  - Structured pruning

Unstructured Sparsity

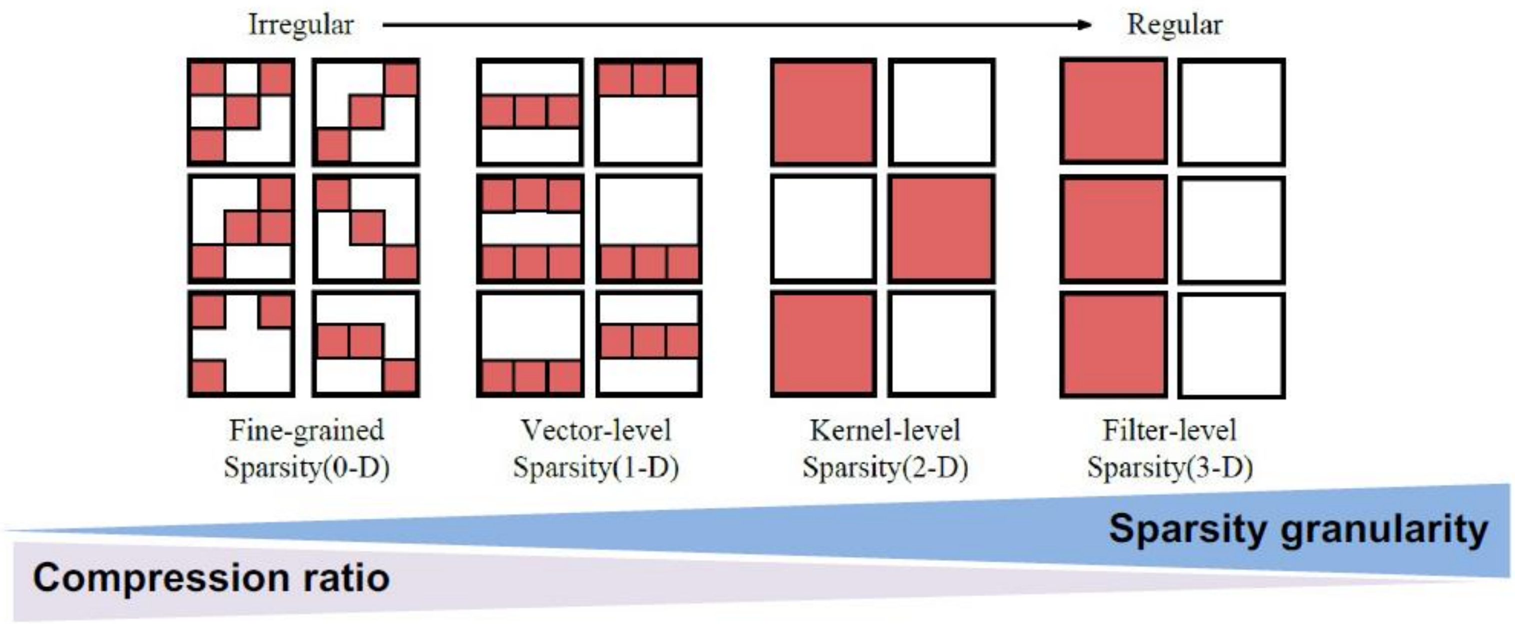
- Unstructured Sparsity does not directly translate to speedup and data compression



Wen et al., NeurIPS, 2016

Structured Sparsity

- Structured pruning leads to more accuracy drop

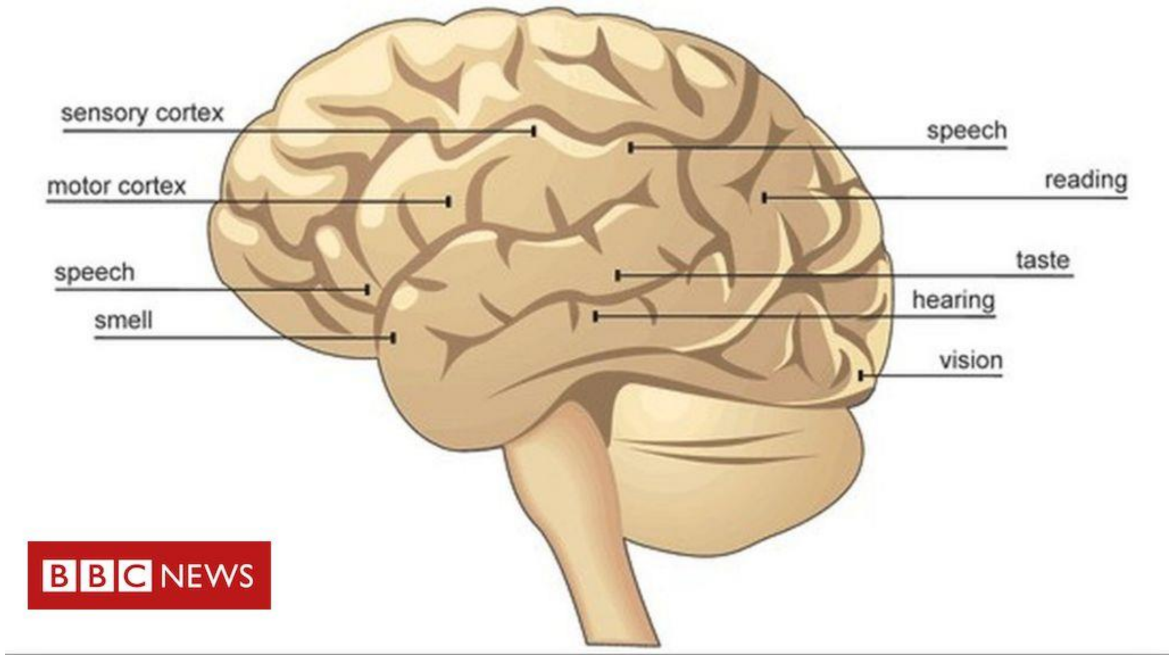


Mao et al., CVPR, 2017

- Is there a new way?

The introduction of Adaptivity in CNN

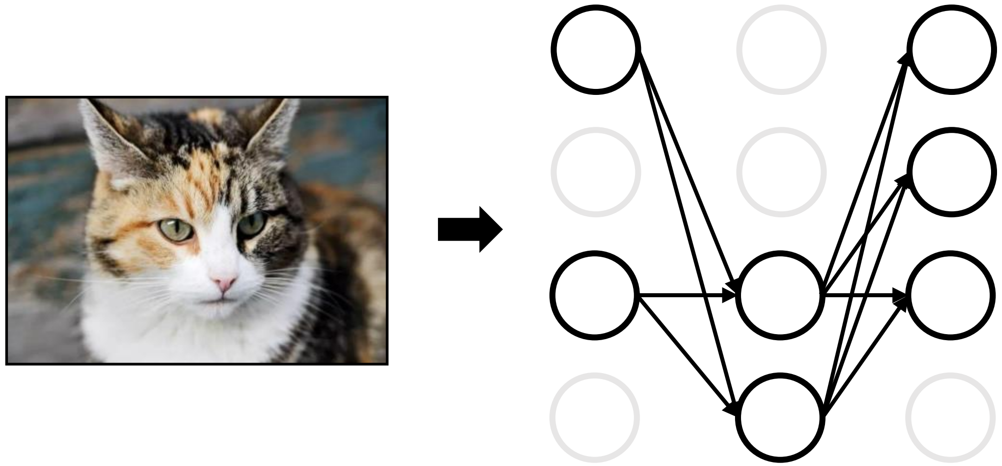
- Motivation

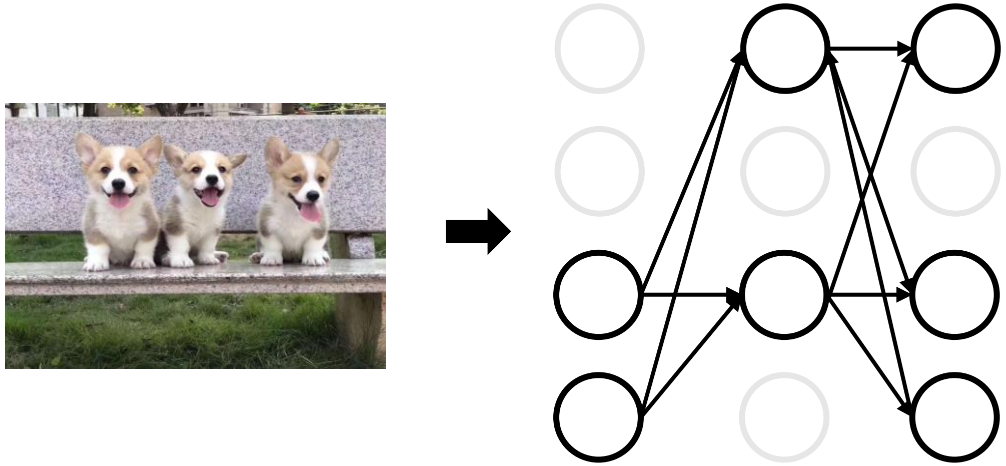


- Advantage
  - Efficiency
  - Representation power
  - Adaptiveness
  - Interpretability
  - Generality

The introduction of Adaptivity in CNN

- Dynamic neural network





- Simple comparison between previous methods and ours

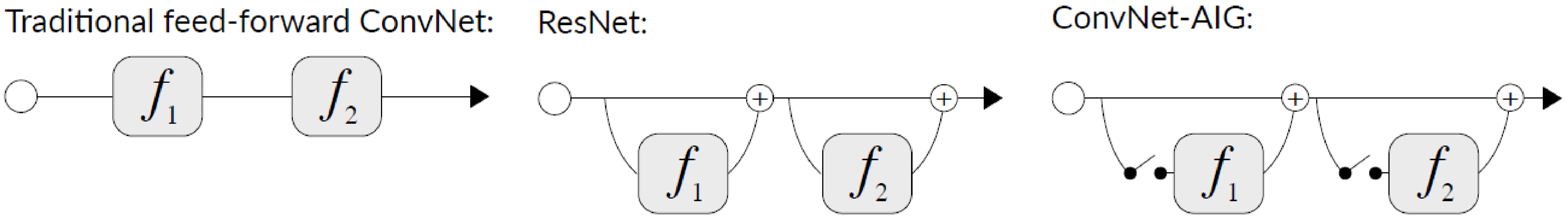
	kernels	inference	compression	accuracy
structured	static	static	✓	low
dy-dnn	static	dynamic	×	medium
ours	dynamic	dynamic	✓	high

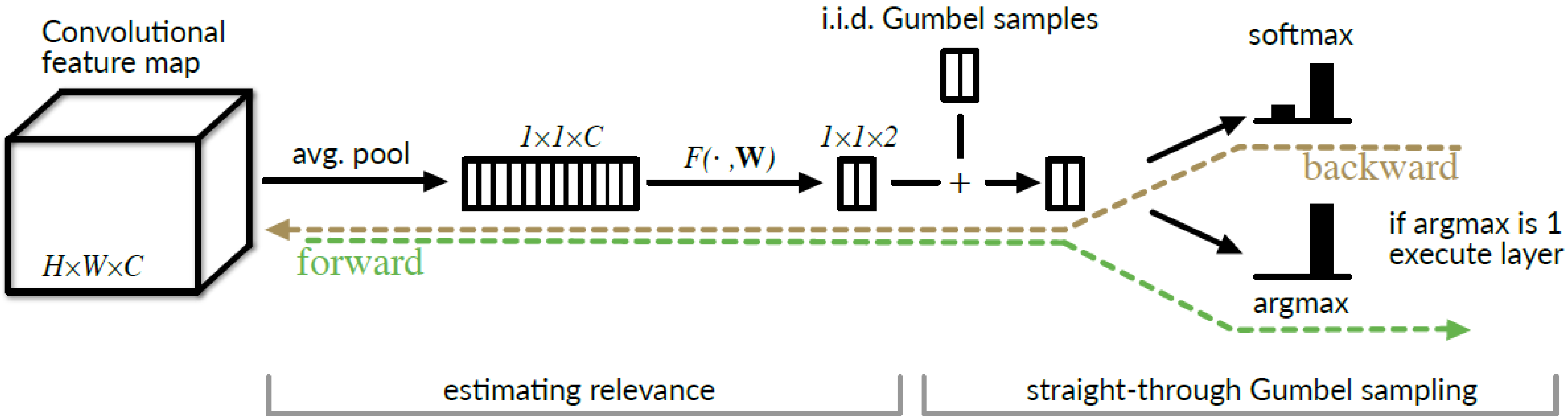
Outline

- Introduction to Neural Network Sparsity
- Background
  - Dynamic inference
  - Dynamic parameter
- Methods
  - Kernel Enhancement
  - Online Channel Selection
  - Hardware Architecture
- Main Results
- Conclusion

Dynamic Inference

- Dynamic layer skipping

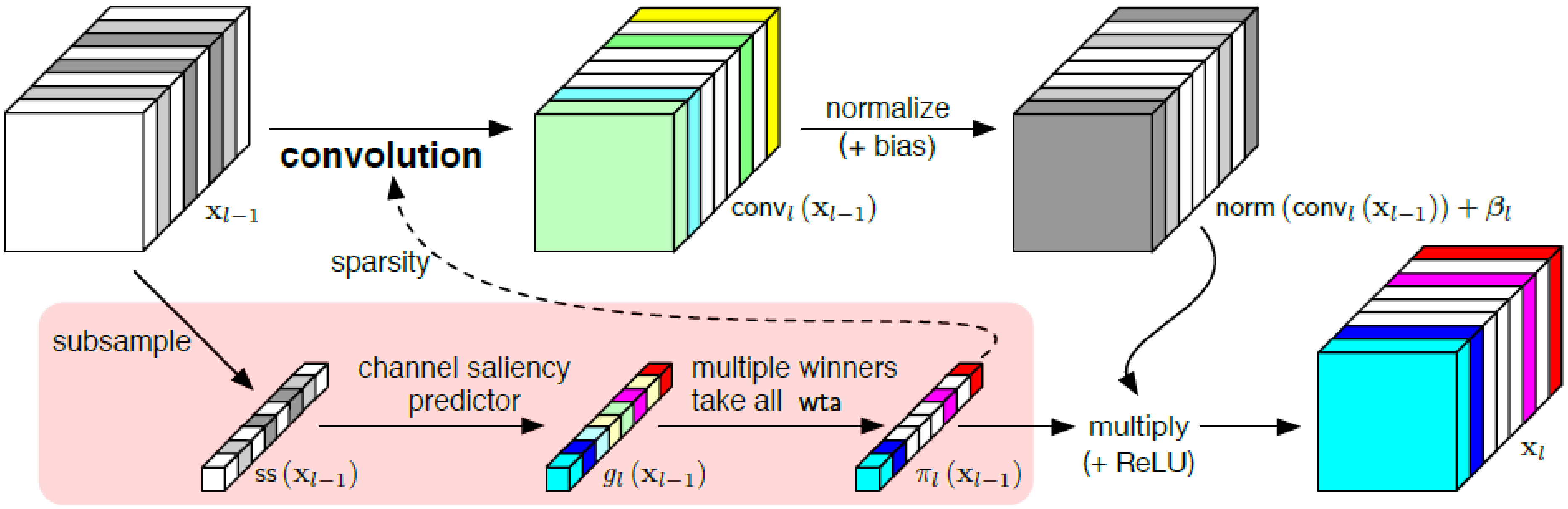




Veit et al., ECCV, 2018

Dynamic Inference

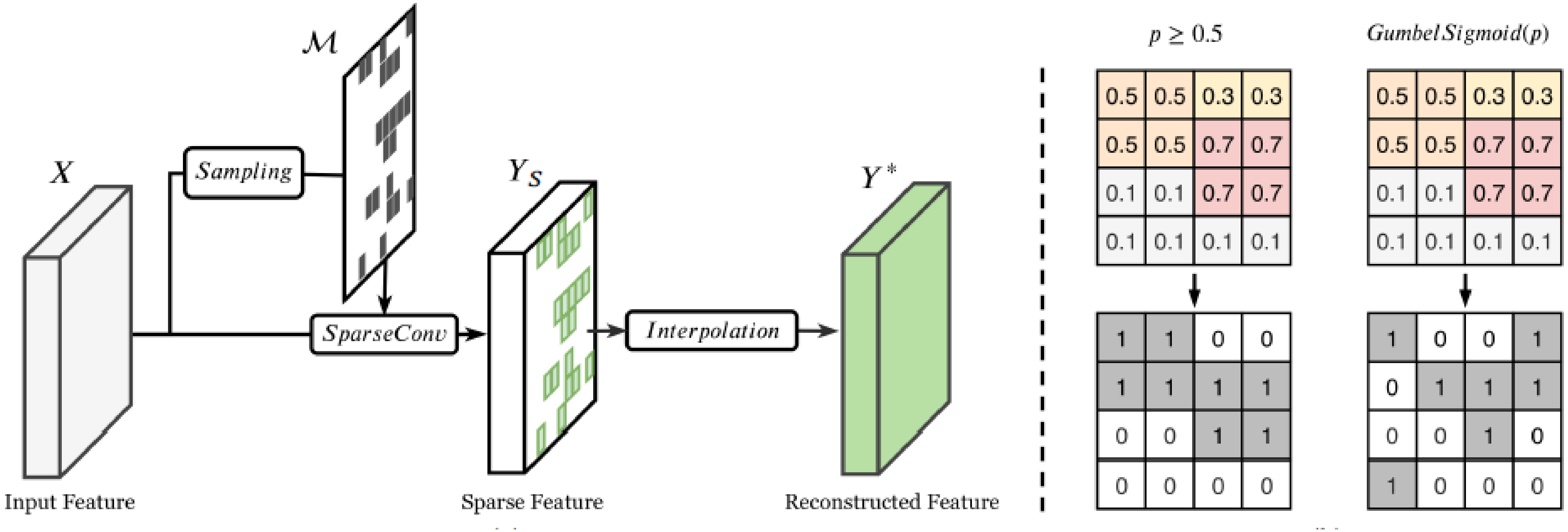
- Dynamic channel pruning



Gao et al., ICLR, 2019

Dynamic inference

- Dynamic spatial pruning

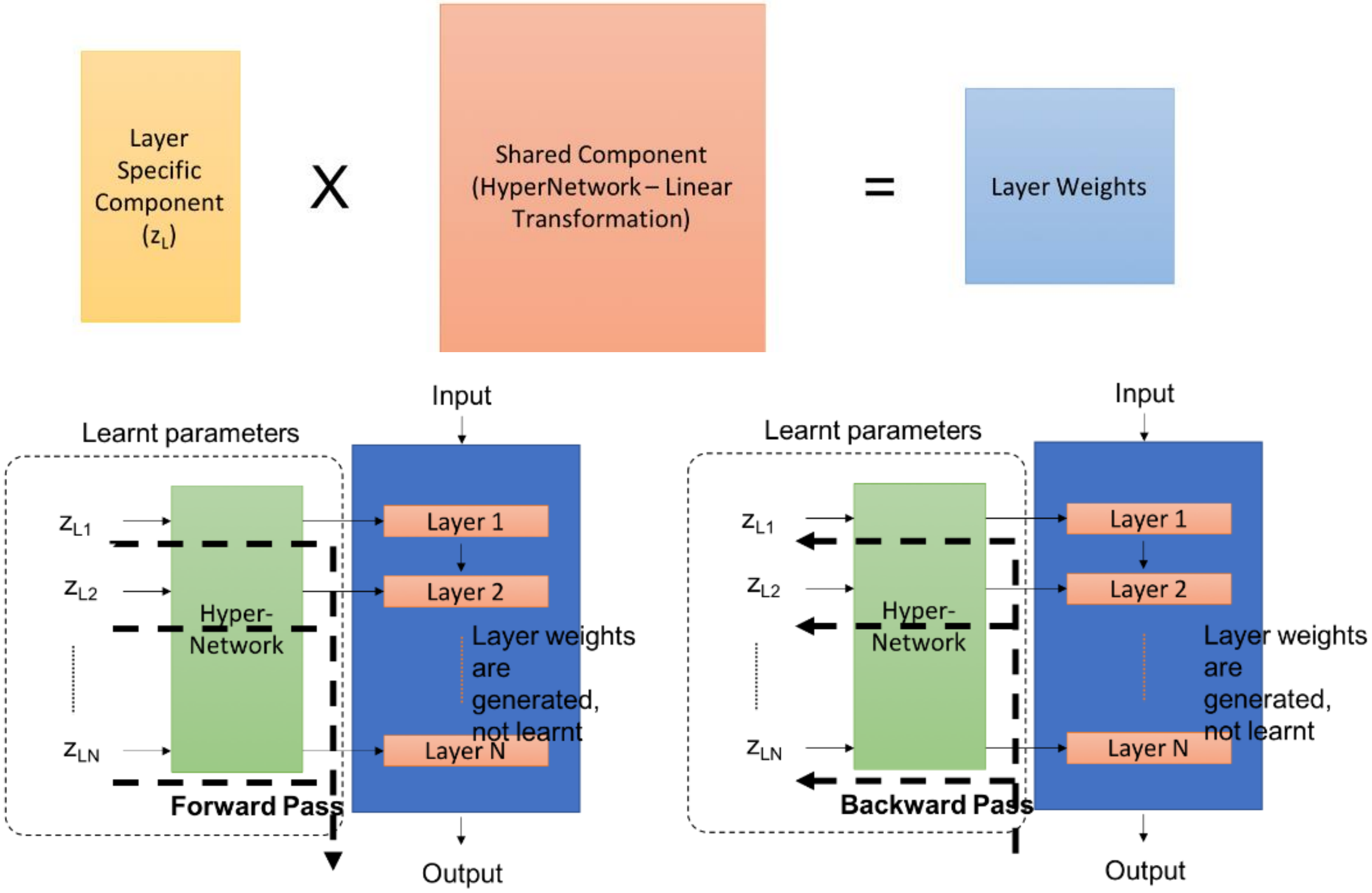


Xie et al., ECCV, 2020

Outline

- Introduction to Neural Network Sparsity
- Background
  - Dynamic inference
  - Dynamic parameter
- Methods
  - Kernel Enhancement
  - Online Channel Selection
  - Hardware Architecture
- Main Results
- Conclusion

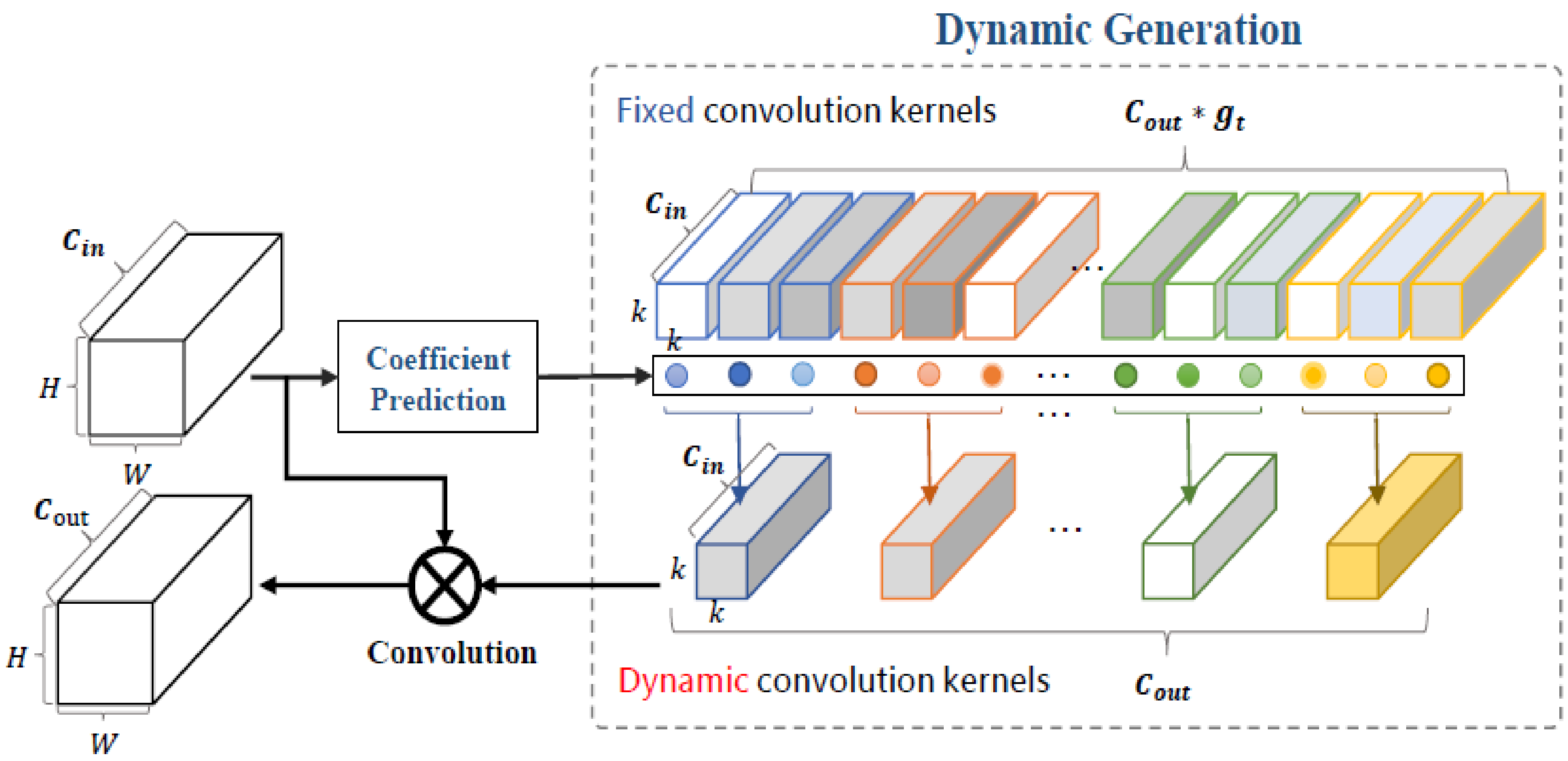
Hyper-network



Ha et al., ICLR, 2016

- We do not know weights for each layer. The weights are generated through linear transformation.
- A hyper-network is the network which is responsible for producing filters for each layer.

Kernel Generator

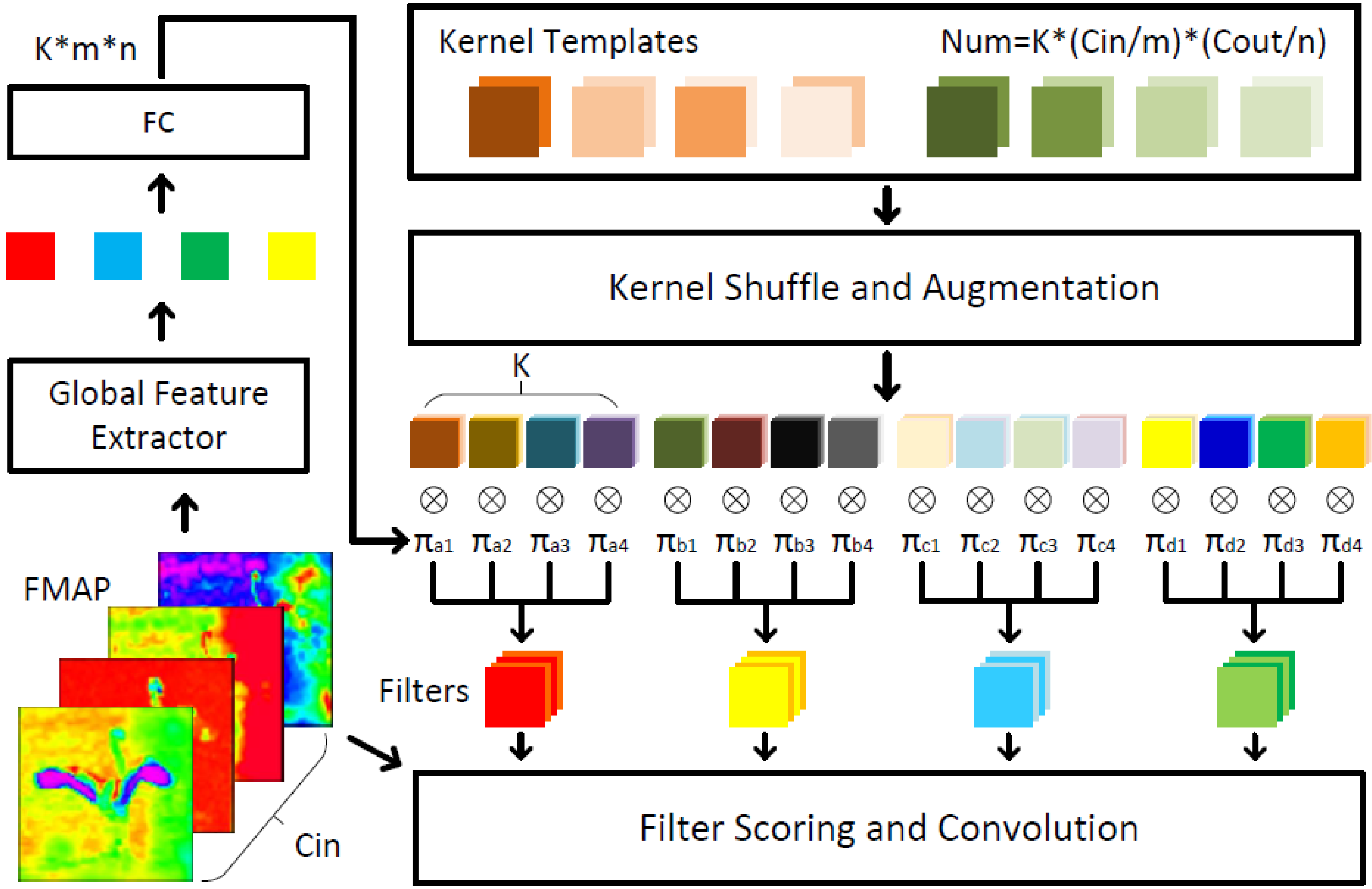


Zhang et al., CoRR, 2020

Outline

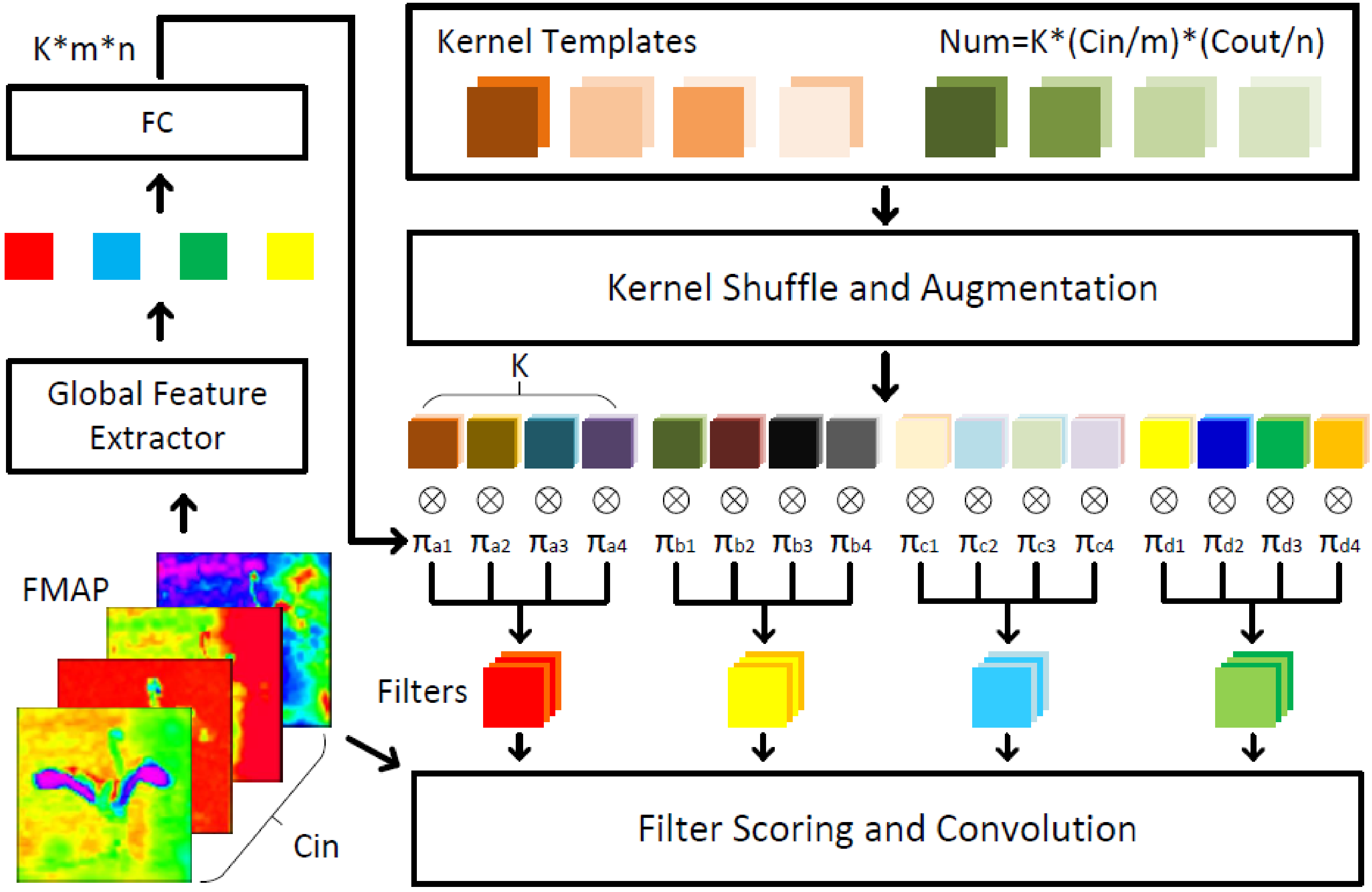
- Introduction to Neural Network Sparsity
- Background
  - Dynamic inference
  - Dynamic parameter
- Methods
  - Kernel Enhancement
  - Online Channel Selection
  - Hardware Architecture
- Main Results
- Conclusion

Overall Structure



- Dynamic parameters
  - Kernel enhancement
  - Dynamic aggregation
- Dynamic inference
  - Dynamic filter selection

Overall Structure



$$\pi = \mathbf{E}(x),$$

$$\mathrm{W_{aj}} = \sum_{i=1}^{K} \pi_{\mathrm{ji}} * \mathrm{T_{ai}},\ a = 0, ...q/K,\ j = 0, ...mn.$$

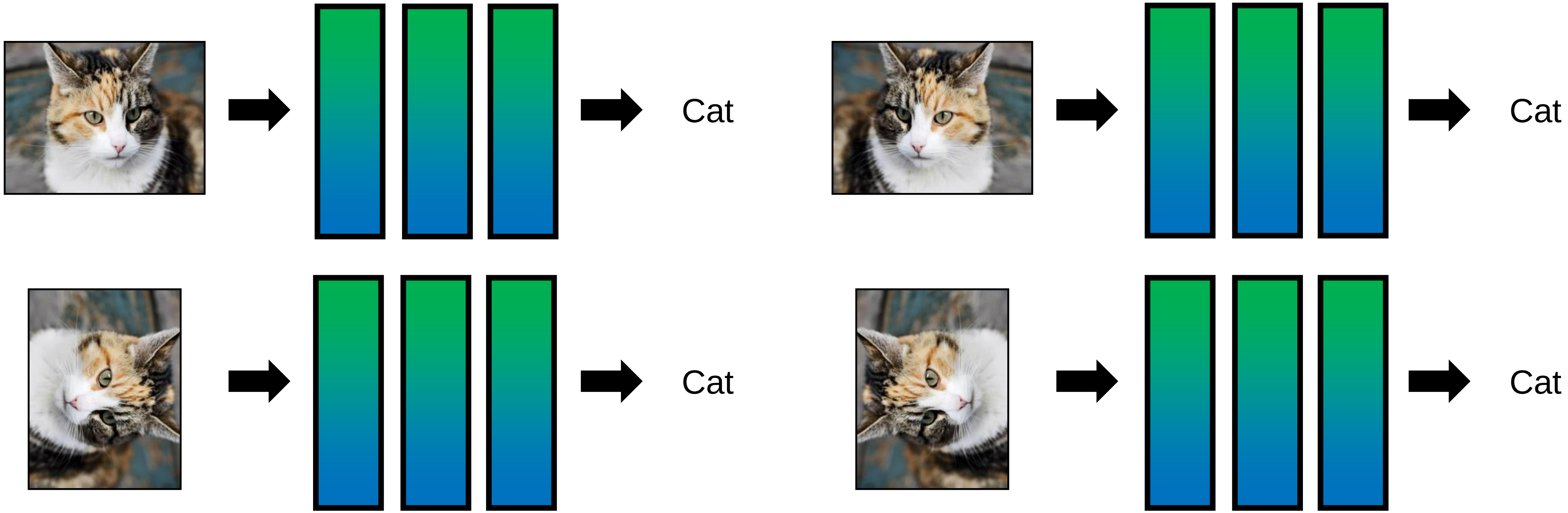
- Extract global feature
- Augment kernel templates
- Aggregate the templates by weight factor to generate filters
- Score the filter to carry channel selection

Outline

- Introduction to Neural Network Sparsity
- Background
  - Dynamic inference
  - Dynamic parameter
- Methods
  - Kernel Enhancement
  - Online Channel Selection
  - Hardware Architecture
- Main Results
- Conclusion

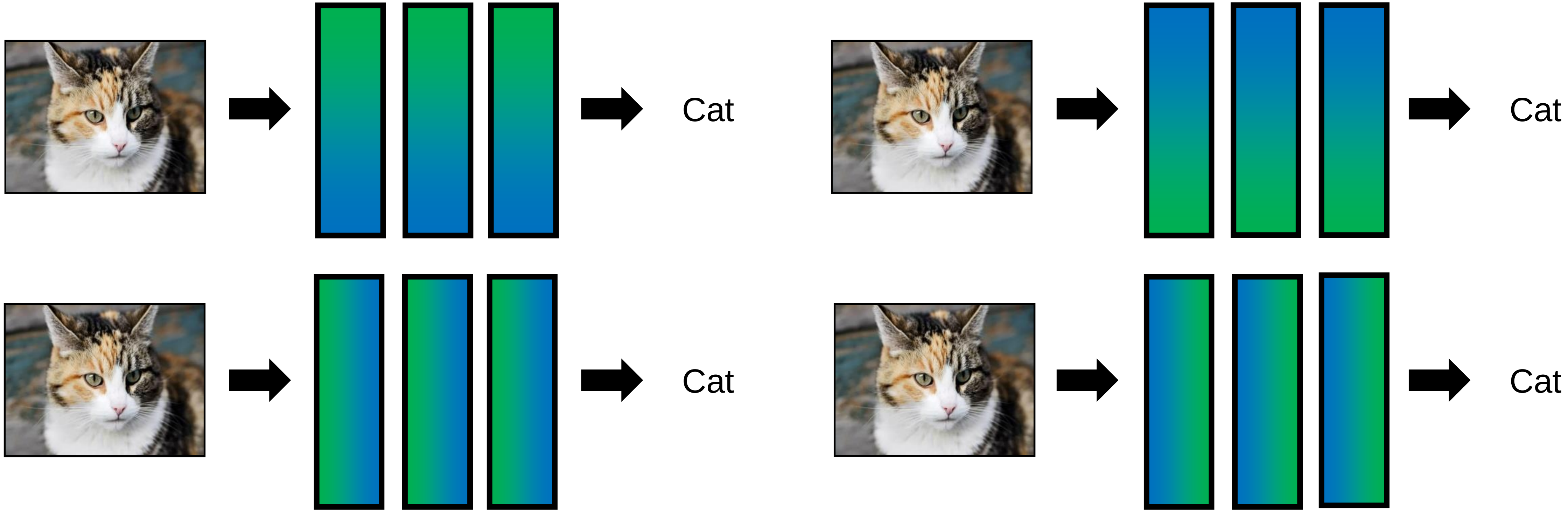
Kernel Enhancement

- Kernel Augmentation
  - **Fact 1**: CNN can handle input augmentation



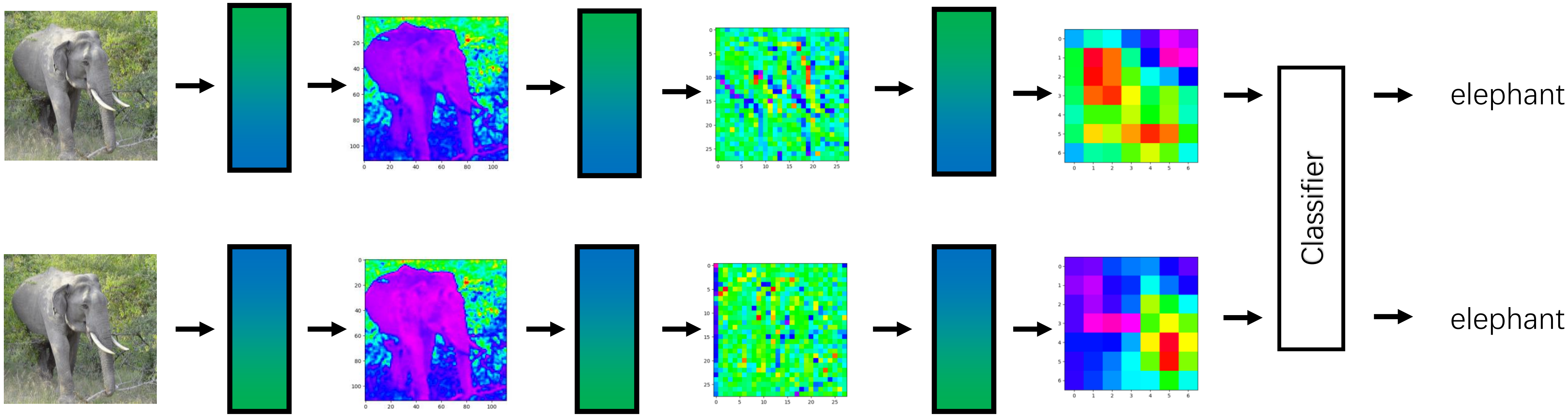
Kernel Enhancement

- Kernel Augmentation
  - CNN with augment inputs ⬌ augment CNNs with origin inputs



Kernel Enhancement

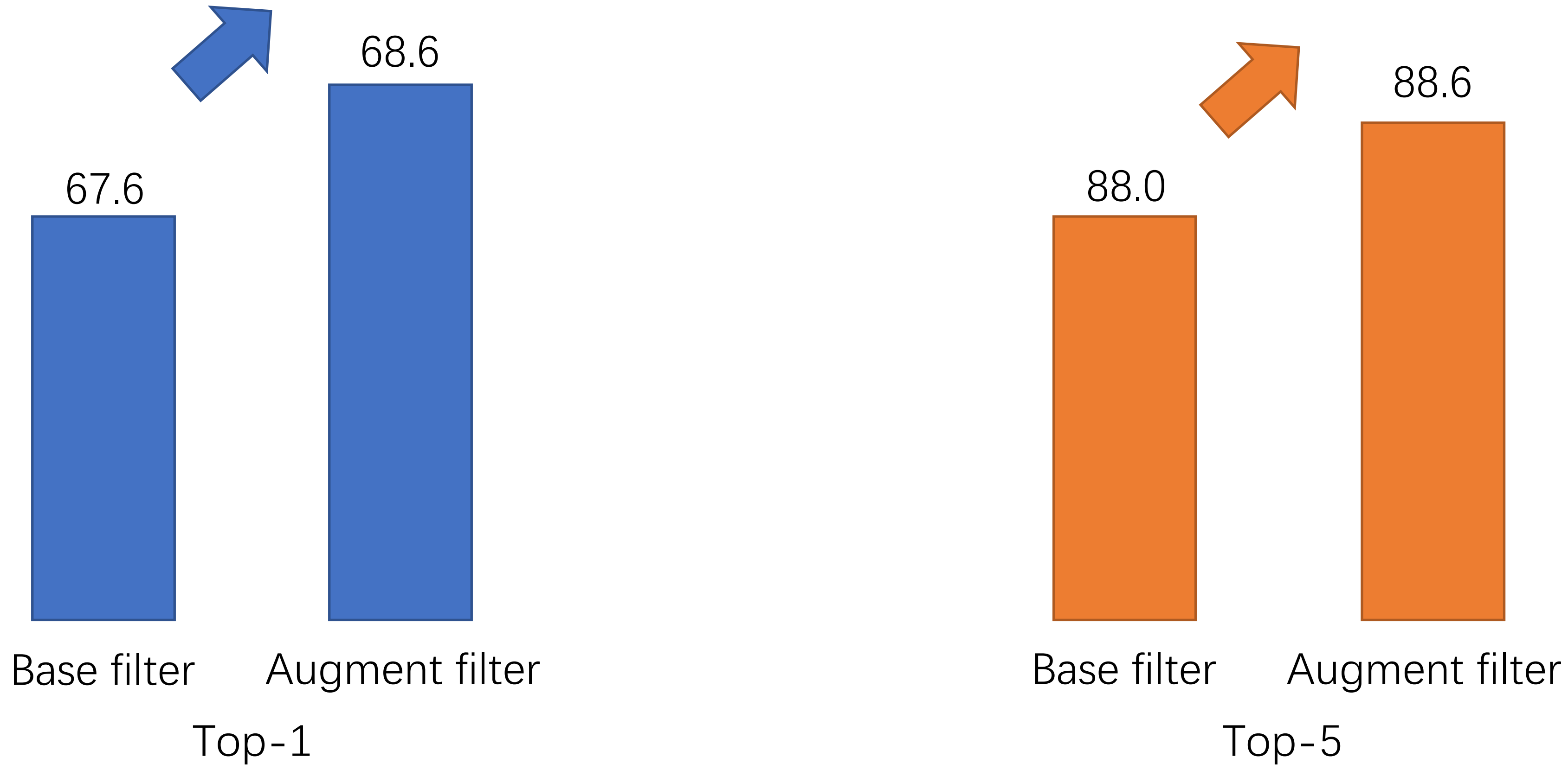
- Kernel Augmentation
  - **Fact 2**: filters work as feature extractor (the origin and the flipped CNN)



Different information extracted, both useful

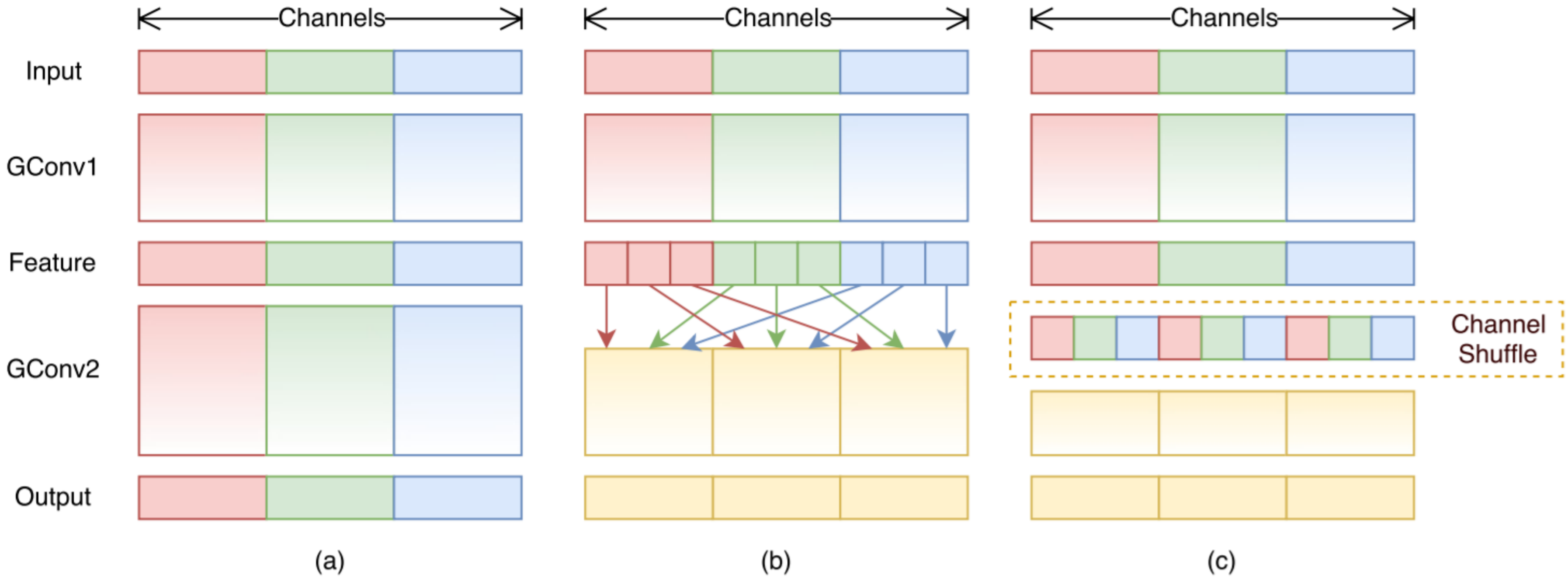
Kernel Enhancement

- Kernel Augmentation



Kernel Enhancement

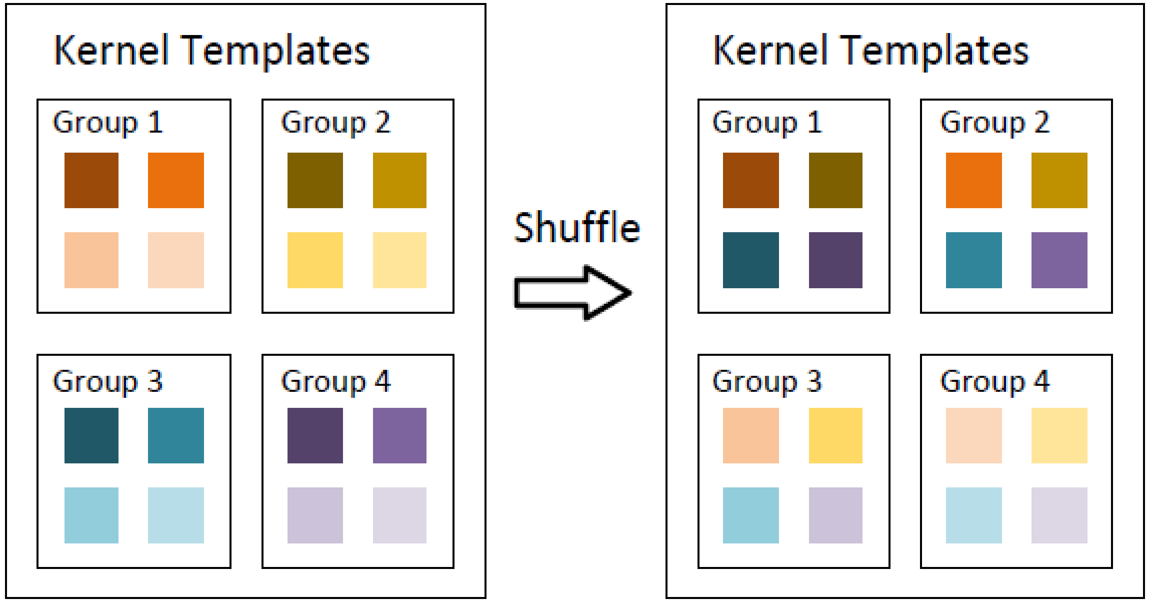
- Kernel Shuffle



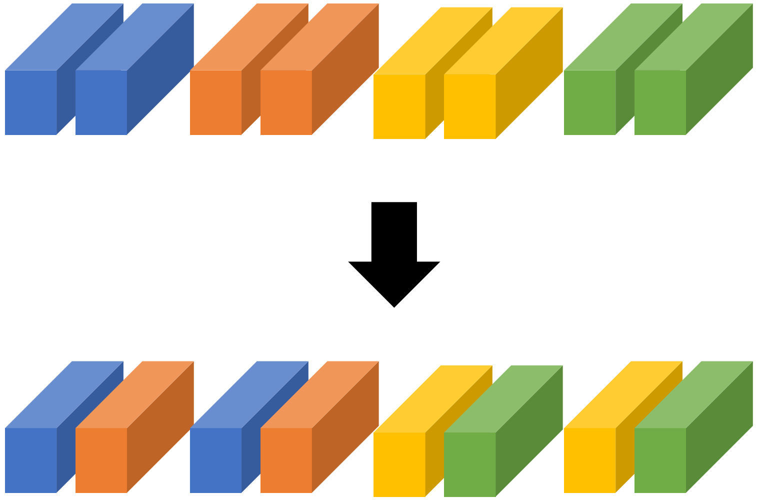
Zhang et al., CVPR, 2018

Kernel Enhancement

- Kernel Shuffle



Shuffle in the group



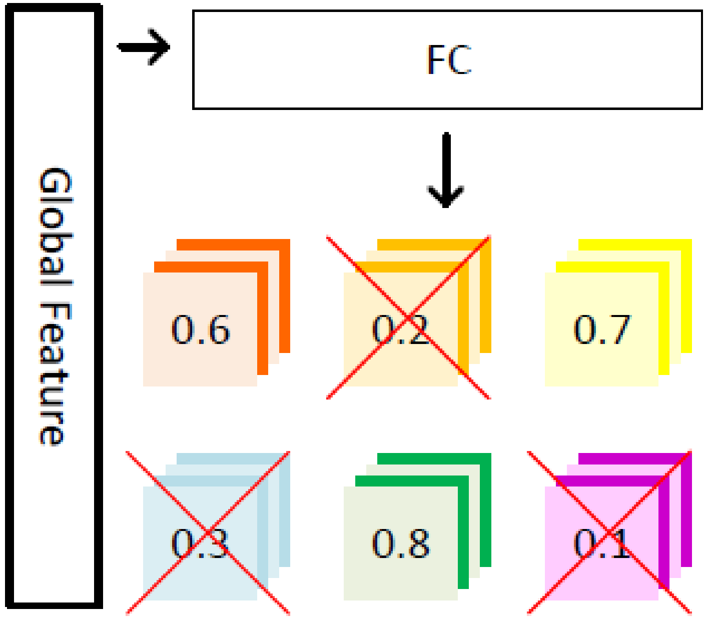
Shuffle in the channel

Outline

- Introduction to Neural Network Sparsity
- Background
  - Dynamic inference
  - Dynamic parameter
- Methods
  - Kernel Enhancement
  - Online Channel Selection
  - Hardware Architecture
- Main Results
- Conclusion

Online Channel selection

- Predict the importance scores of each channel



- Process

$$scores = \mathbf{S}(gf), \quad \mathrm{M} = Sign(scores - \delta),$$
$$out = \phi(x) * M * scores.$$

- The truncation on the channel is not differentiable

How to train it?

Online Channel selection

- The Sign function
  - Hard Sign

$$Sign(x) = \begin{cases} 1, & x > 0 \\ 0, & x < 0 \end{cases}$$

  - Soft Sign

$$Sign(x) = \begin{cases} 1, & with\ probabilty\ x \\ 0, & with\ probability\ 1 - x \end{cases}$$

  - Backward propagation

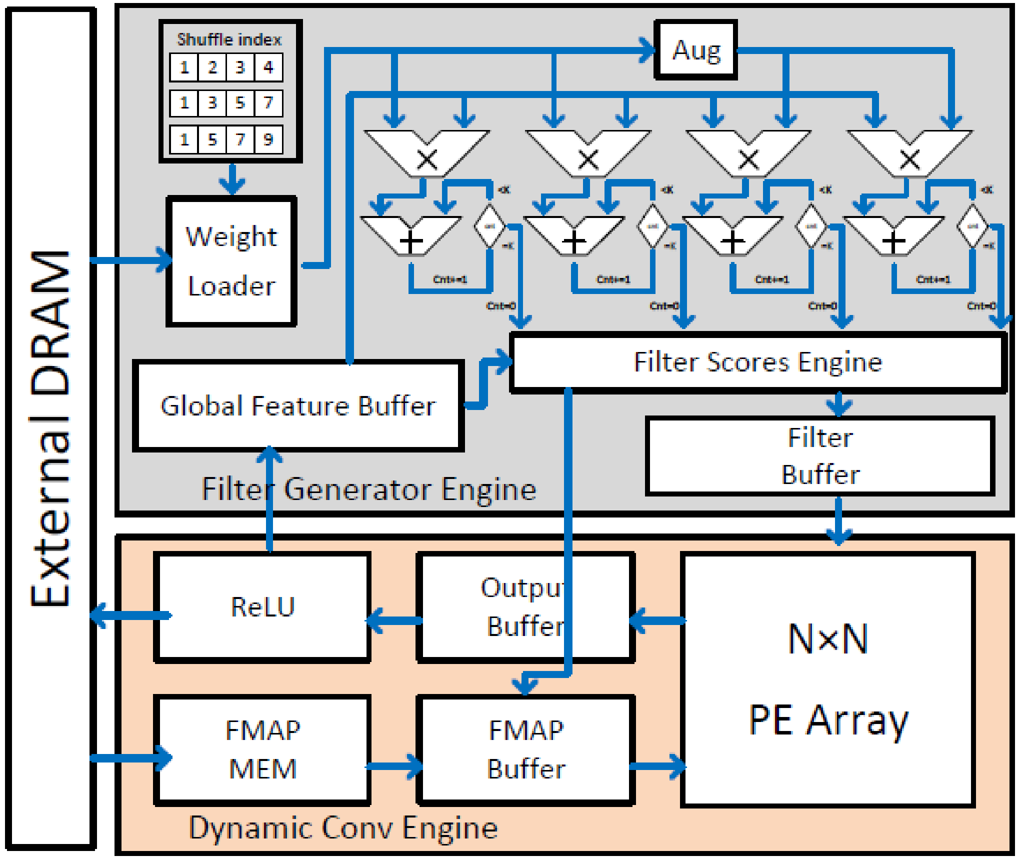
$$\frac{\partial M}{\partial scores} = 1 - tanh^2(scores - \delta).$$

Outline

- Introduction to Neural Network Sparsity
- Background
  - Dynamic inference
  - Dynamic parameter
- Methods
  - Kernel Enhancement
  - Online Channel Selection
  - Hardware Architecture
- Main Results
- Conclusion

Hardware architecture

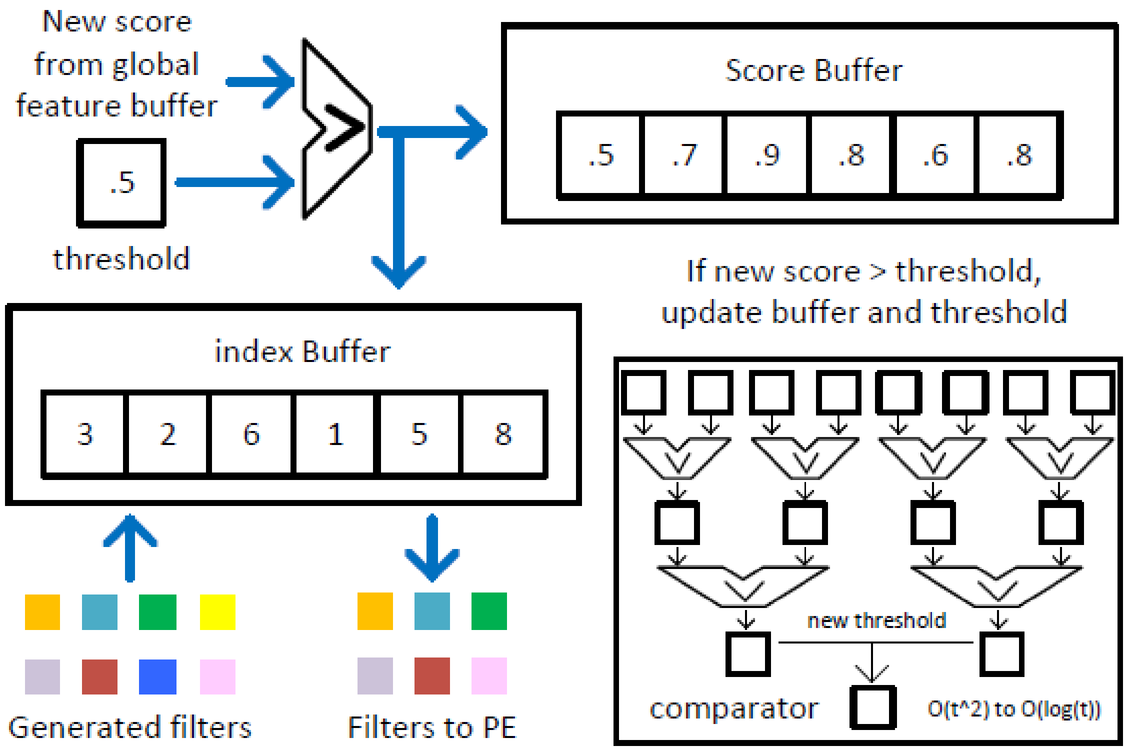
- Overall structure



- Generate the aggregation factor and predict the filter score.
- Import the templates and online generate the filters in parallel.

Hardware architecture

- Channel selecting module

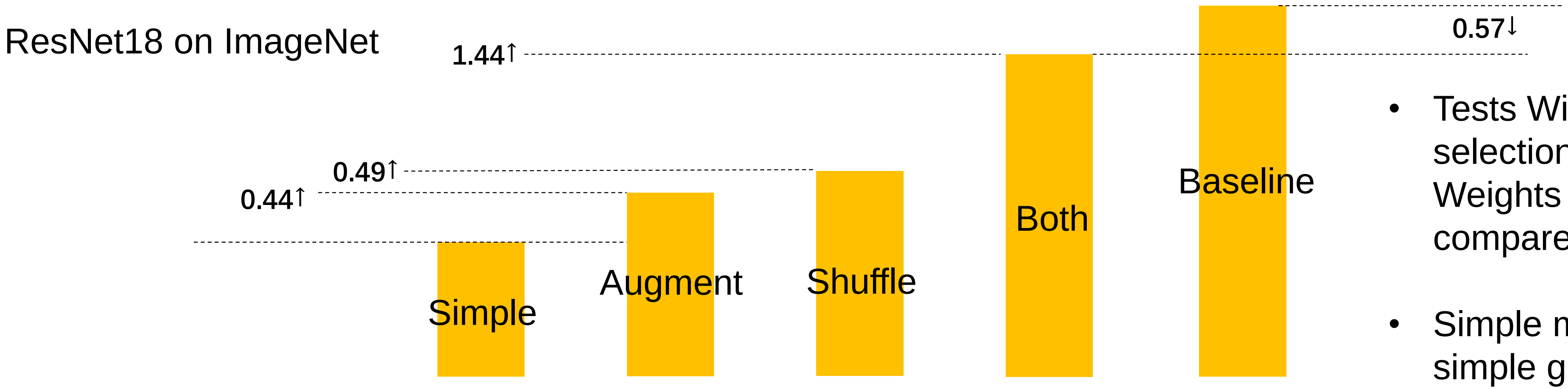


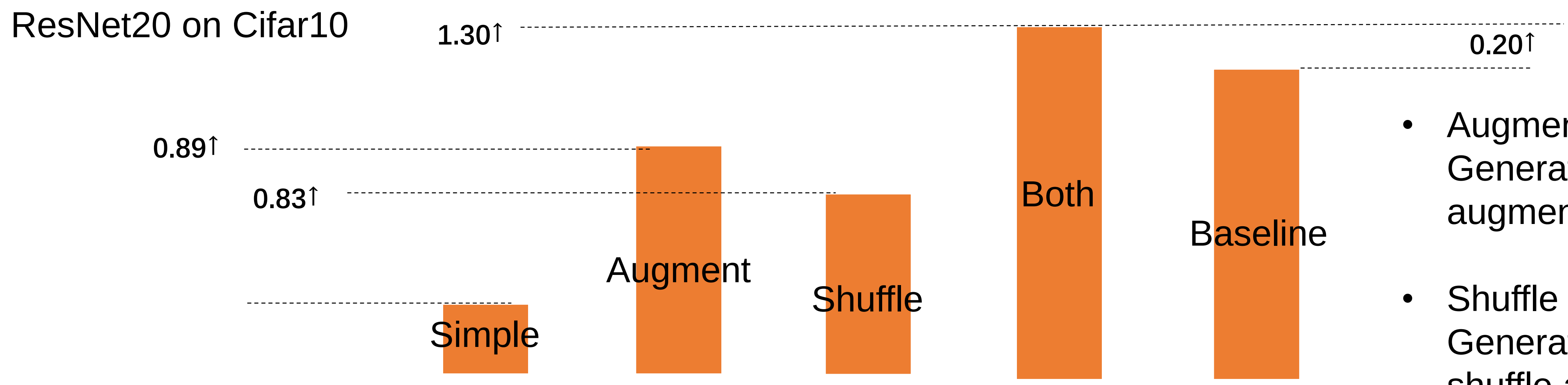
- Use a comparator tree to accelerate the score comparison.
- Online maintain a channel index buffer to control the data load to PE array.

Outline

- Introduction to Neural Network Sparsity
- Background
  - Dynamic inference
  - Dynamic parameter
- Methods
  - Kernel Enhancement
  - Online Channel Selection
  - Hardware Architecture
- Main Results
- Conclusion

Kernel Enhancement

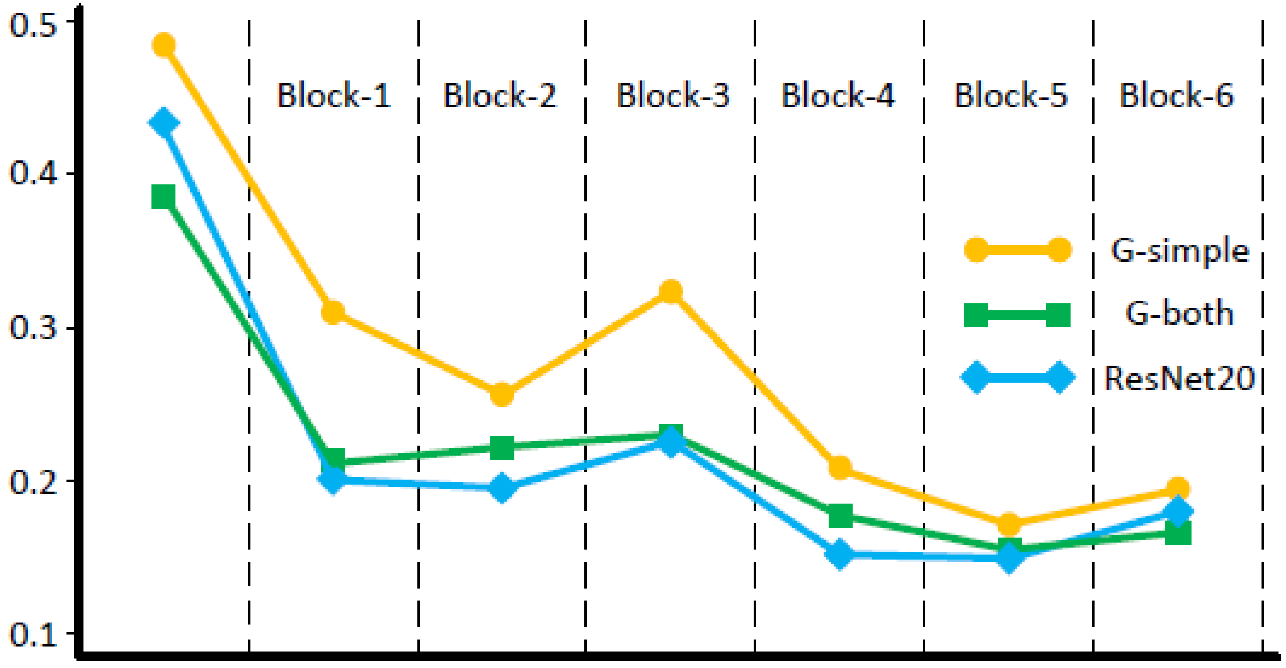




- Tests Without channel selection module. Weights are saved ~50% compared with baseline
- Simple means only simple generator applied
- Augment denotes Generator with kernel augmentation applied
- Shuffle denotes Generator with kernel shuffle applied

Kernel Enhancement

- Pearson product-moment correlation coefficient factor of feature map

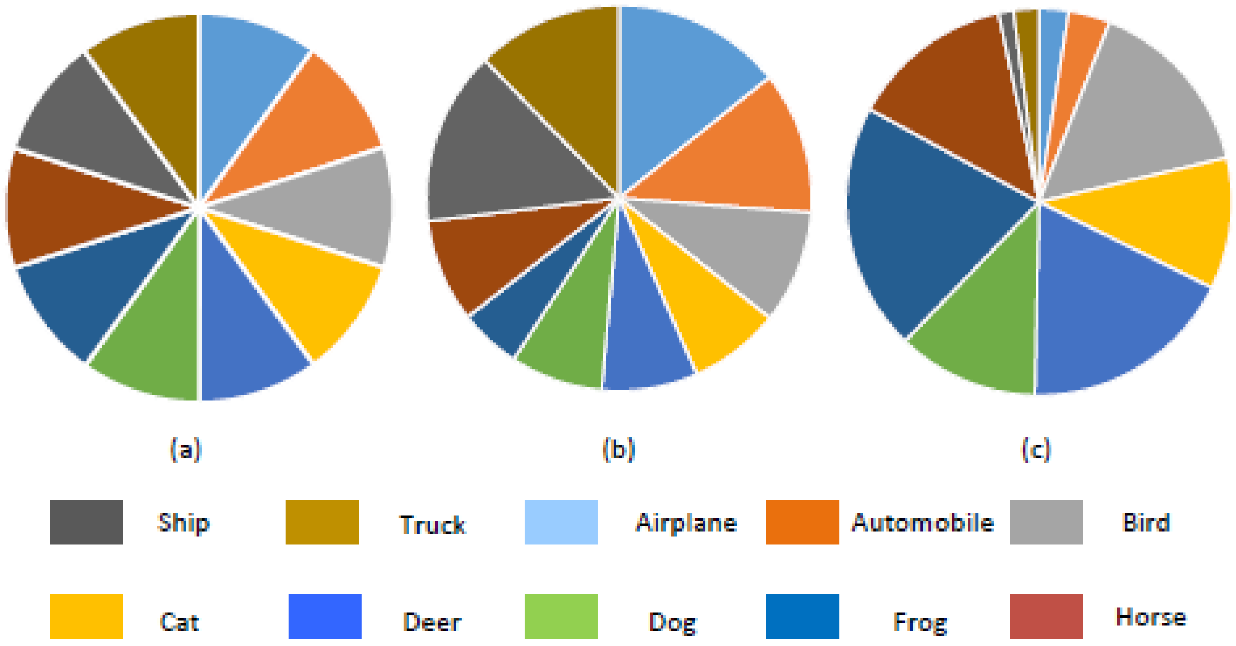


$$\rho_{X,Y} = \mathrm{corr}(X,Y) = \frac{\mathrm{cov}(X,Y)}{\sigma_X \sigma_Y} = \frac{E[(X-\mu_X)(Y-\mu_Y)]}{\sigma_X \sigma_Y}$$

- Absolute value is shown here
- Kernel enhancement can help the generator produce filters that are less similar

Channel Selection

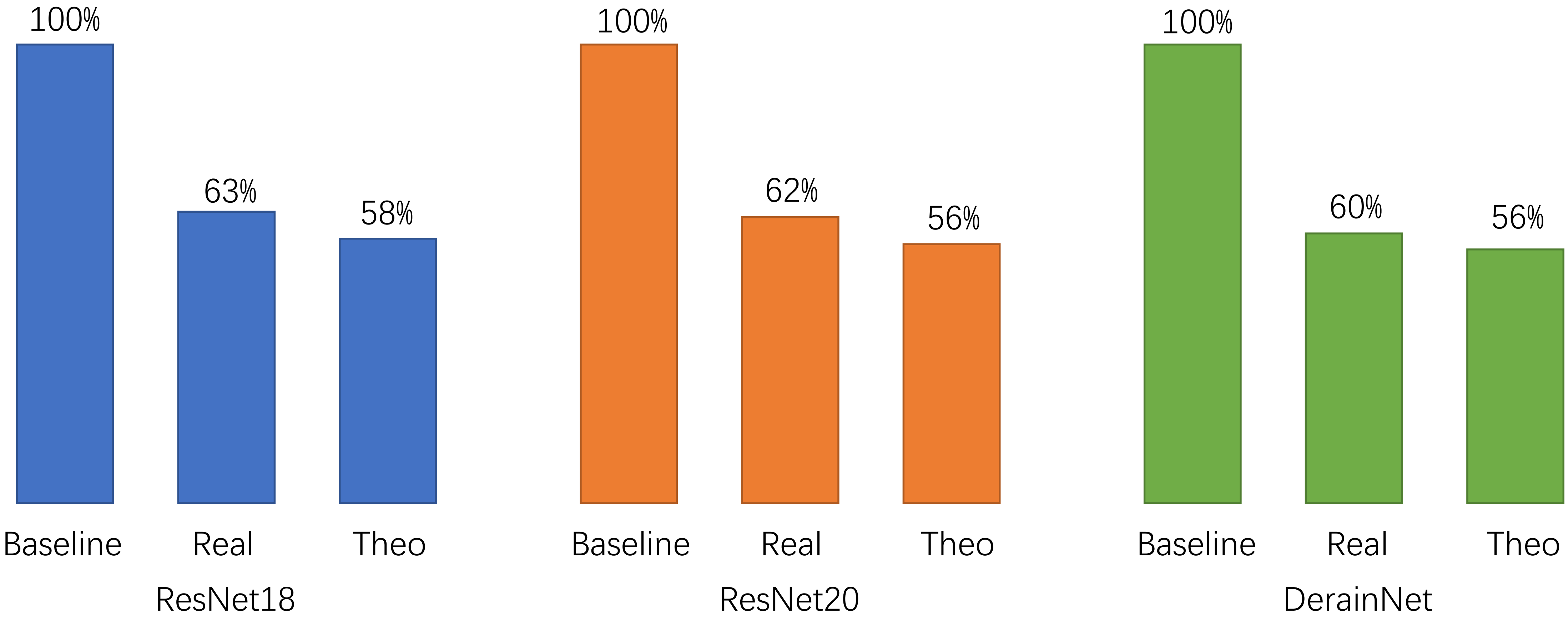
- Utilization frequency of some representative filters by each classes in Cifar10



- Input of similar classes tend to select same channels and filters
- No filters are really pruned, so the representation ability is preserved

Overall results

- Overall speedup on Ultra96-v2



Overall results

- Accuracy comparison

(a) ResNet18 on ImageNet

Method	Top-1 Acc	Top-5 Acc	Params	Flops
MIL [15]	3.65%↓	2.3%↓	0%↓	44.7%↓
FBS [6]	2.54%↓	1.46%↓	0%↓	49.5%↓
CGNN [16]	1.62%↓	1.03%↓	0%↓	38.3↓
AAS [3]	1.41%↓	0.74%↓	0%↓	48.5%↓
Ours	**1.31%↓**	**0.59%↓**	42%↓	46.4%↓

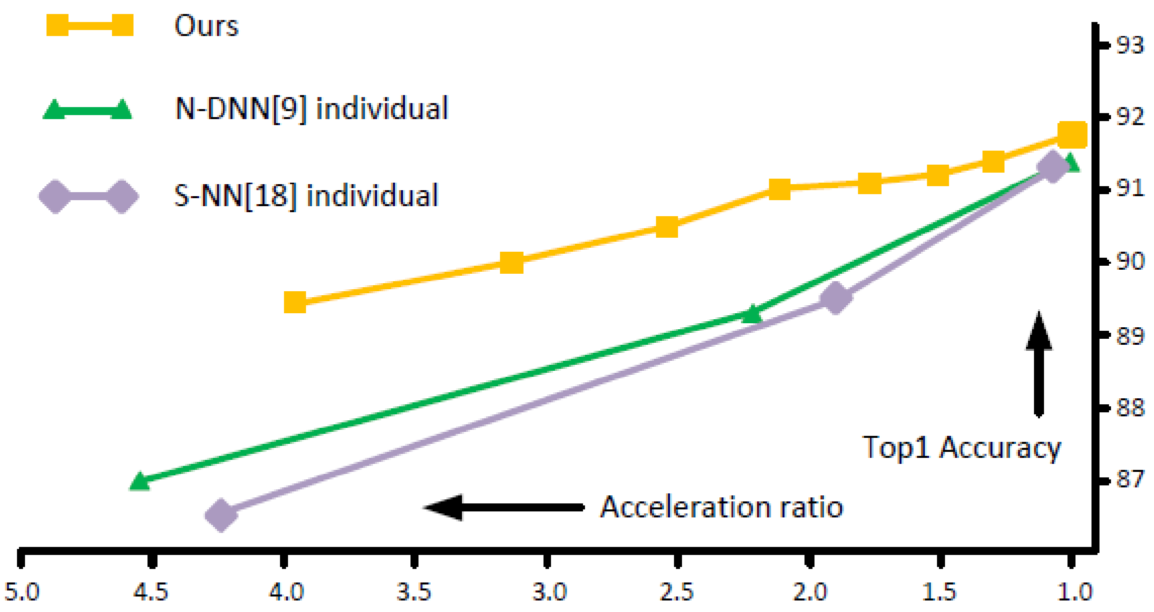
(b) ResNet20 on Cifar10

Method	Top-1 Acc Drop	Params Drop	FLOPs Drop
AIG [17]	1.9%↓	0.0%↓	50.4%↓
FBS [6]	0.48%↓	0.0%↓	**52.5%↓**
Ours	**0.46%↓**	**45.3%↓**	**51.8%↓**

- Comparison with some state-of-the-art dynamic pruning methods
- Our method generally have higher accuracy at similar speedup ratio

Overall results

- The design space and accuracy comparison



ResNet-20 on Cifar-10

- Comparison with two structured pruning methods
- At similar acceleration ratio, our method has better accuracy

Outline

- Introduction to Neural Network Sparsity
- Background
  - Dynamic inference
  - Dynamic parameter
- Methods
  - Kernel Enhancement
  - Online Channel Selection
  - Hardware Architecture
- Main Results
- Conclusion

Conclusion

- Dynamic inference with dynamic parameters can truly stimulate the potential of dynamic neural network.
- Kernel enhancement can help the generator to produce more unique filters.
- Online channel selection can help choose the most suitable filters for each input.
- In the future, new architecture can be designed for more fine-grained dynamic pruning pattern.

Thank you!