HAN LAB
Hardware, AI and Neural-nets

AutoML for 3D Deep Learning

Zhijian Liu, Haotian Tang and Song Han

Massachusetts Institute of Technology

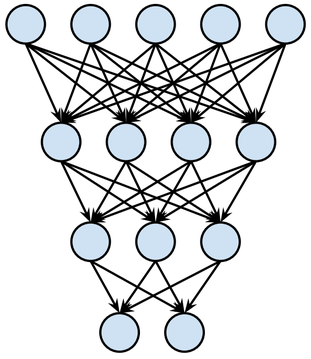
Cloud AI → **Mobile AI** → **Tiny AI**

Memory: 32GB | Memory: 4GB | Memory: 100KB

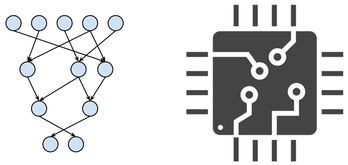
Computation: TFLOPs | Computation: GFLOPs | Computation: MFLOPs

Efficicient AI & Model Compression & TinyML

Large Neural Networks

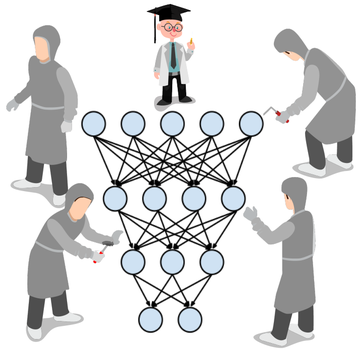


Small Neural Networks



Low-Power Hardware

AutoML Improves Productivity



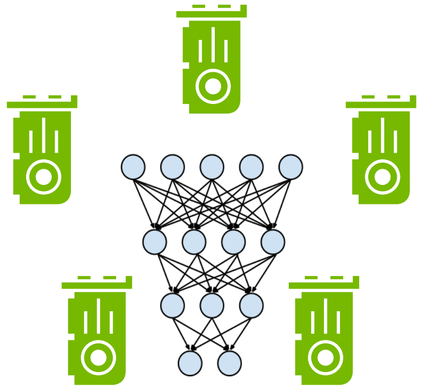
Rely on Human Expertise

Manual Architecture Design

VGGNets
Inception Models
ResNets
DenseNets
....

Less Engineer Resources

More Computational Resources

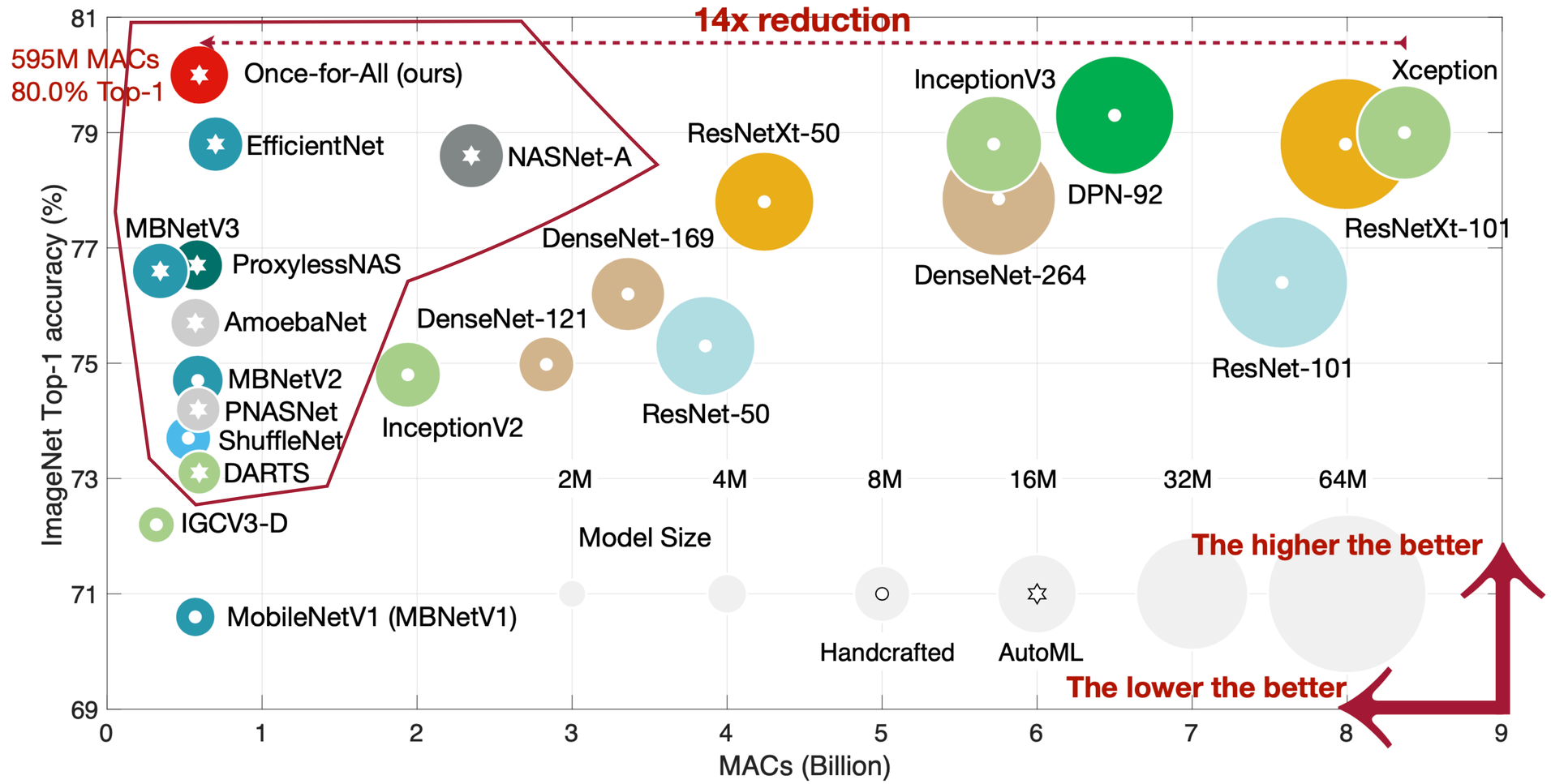


Use Machine Learning

Automatic Architecture Search

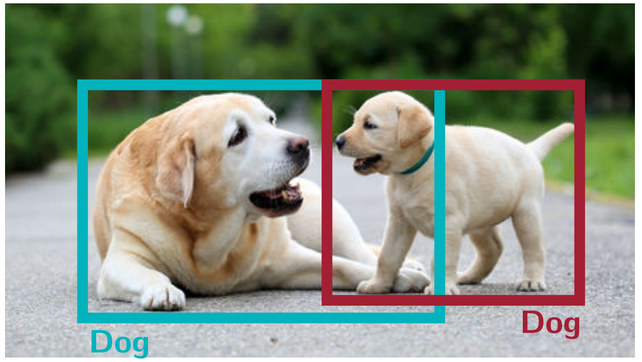
Reinforcement Learning
Neuro-evolution
Bayesian Optimization
Monte Carlo Tree Search
...

AutoML Outperforms Human Design



AutoML has Various Applications

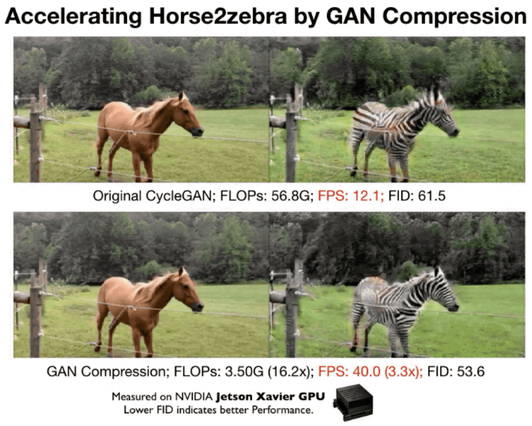
Image Classification



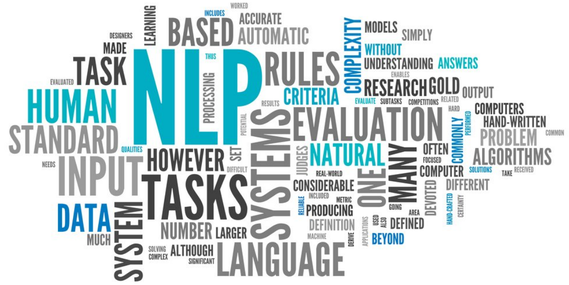
Object Detection

Video Recognition

Semantic Segmentation

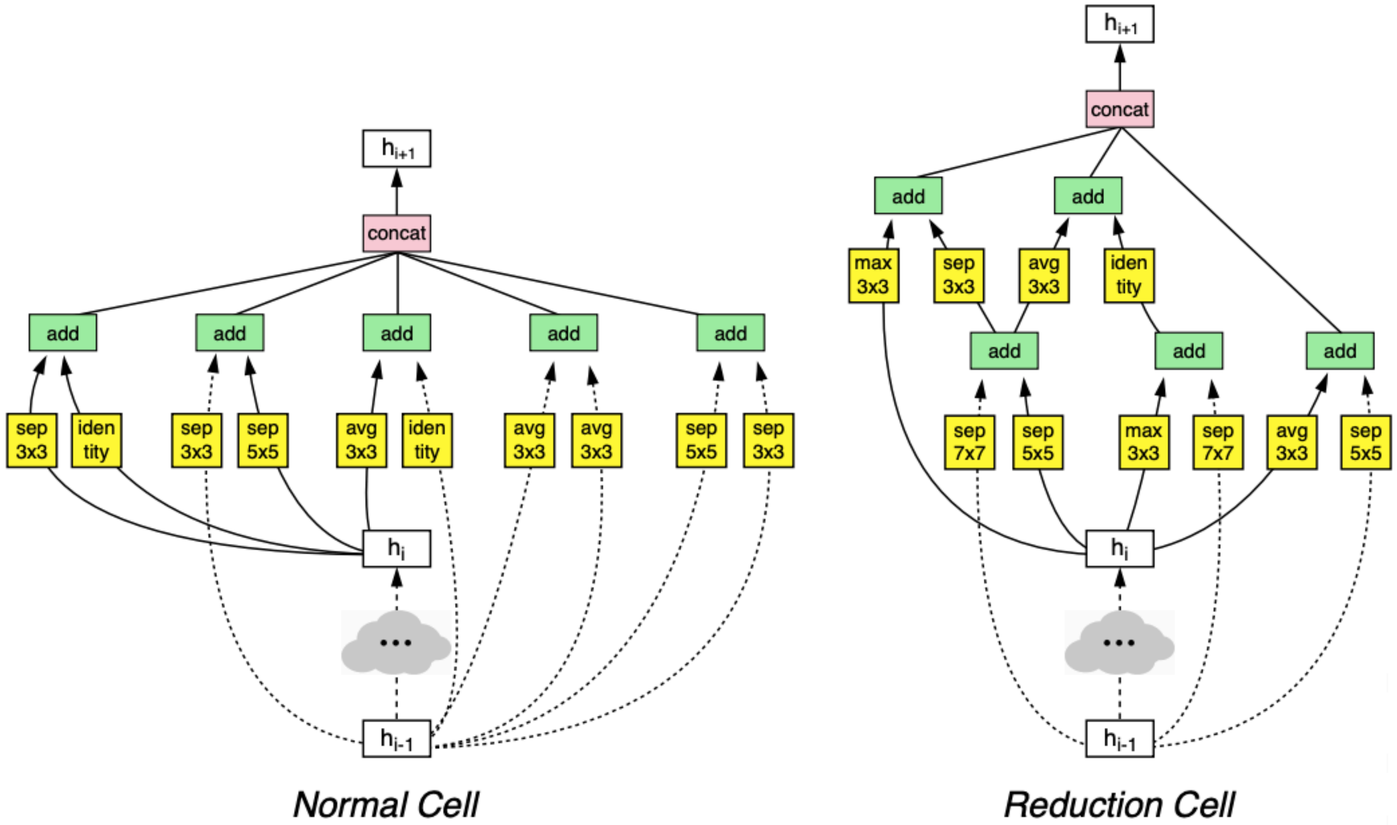


GAN Compression

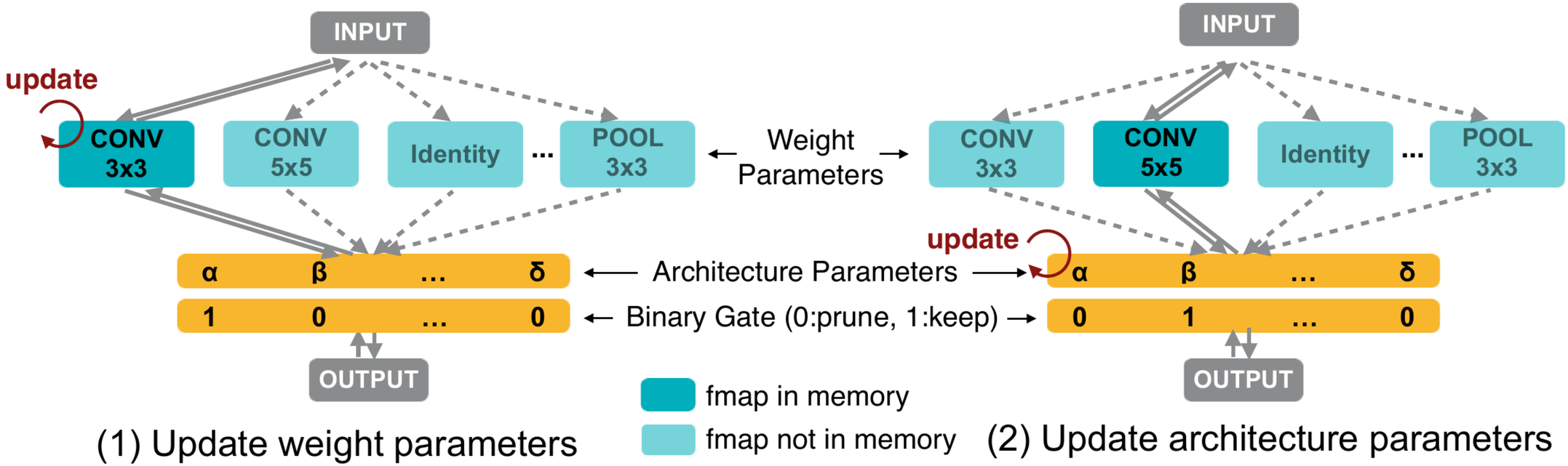


Natural Language Processing

AutoML for Image Classification



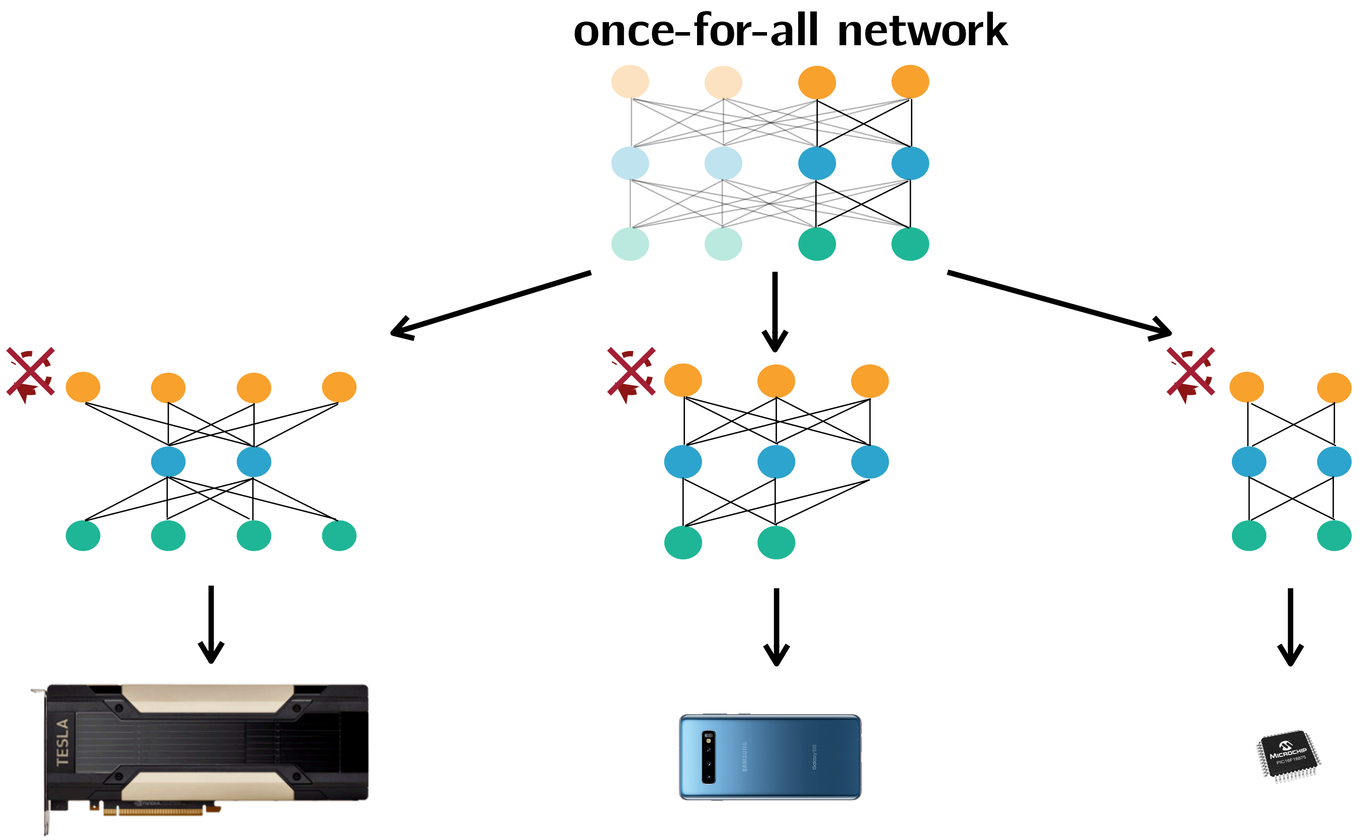
AutoML for Image Classification



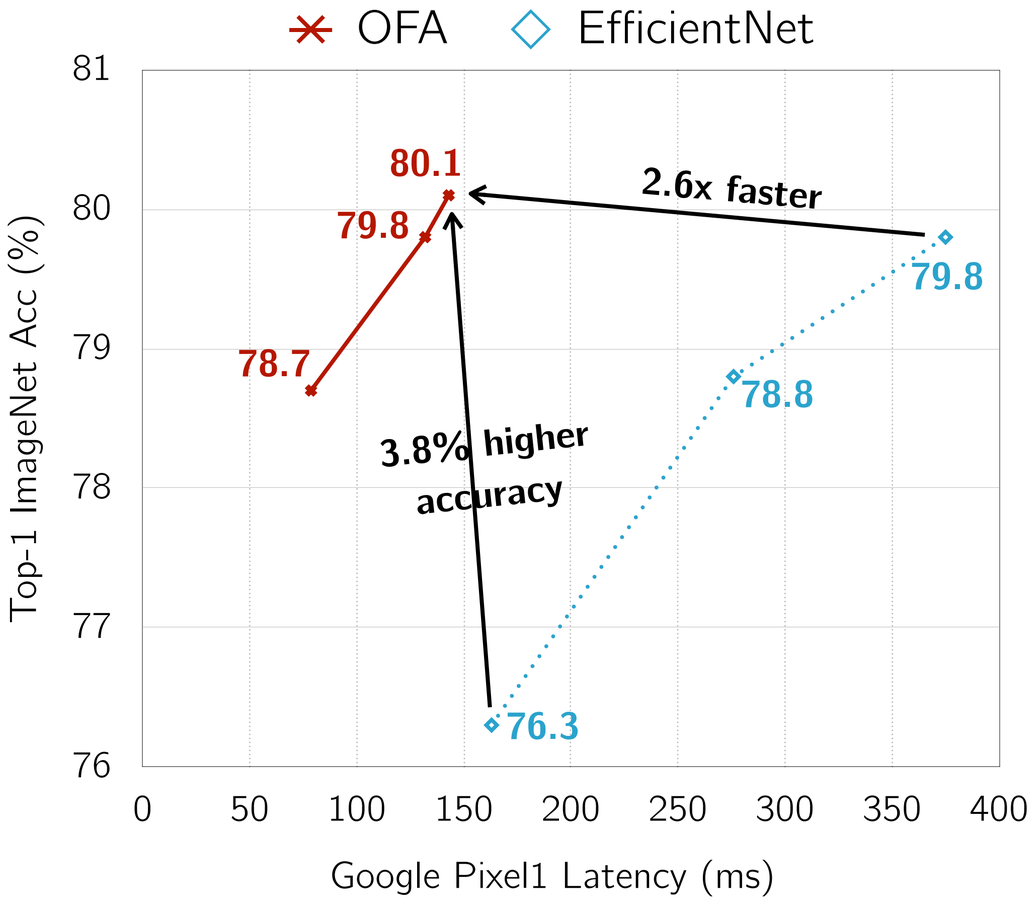
AutoML for Image Classification

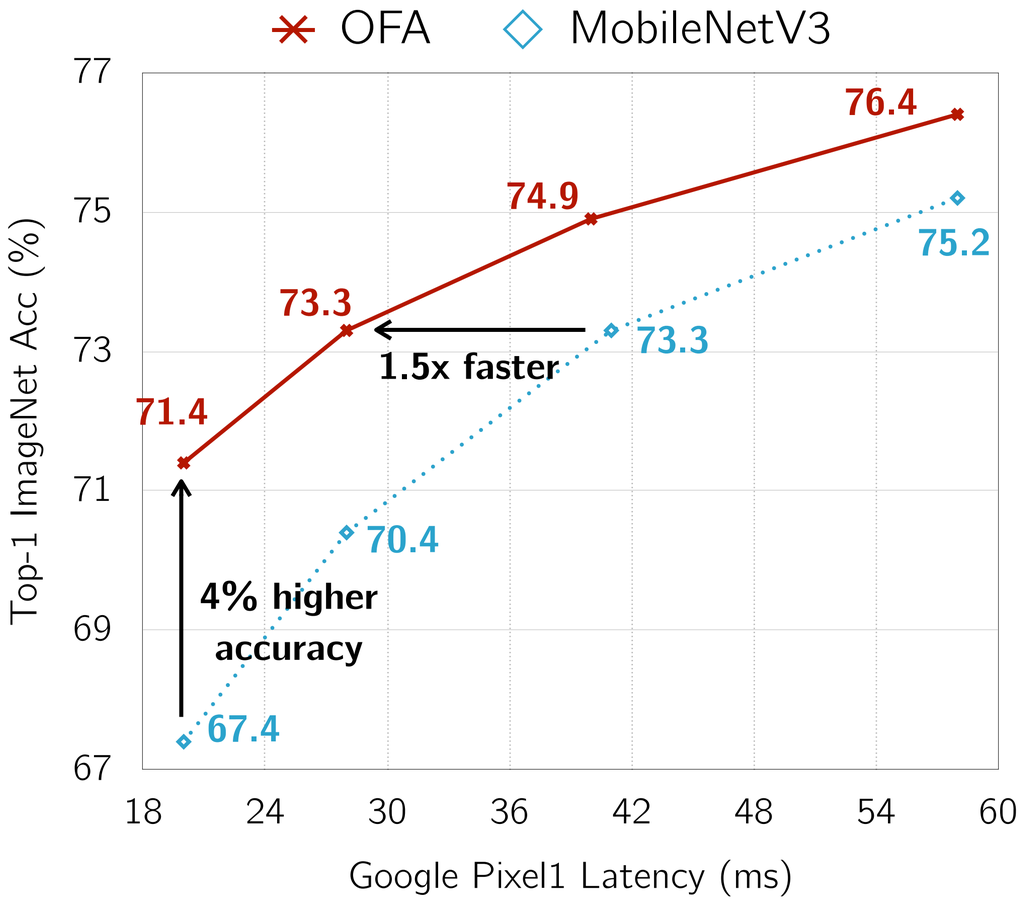
	Model	Top-1	Latency	Hardware Aware	No Proxy	No Repeat	Search Cost
Manually Designed	MobilenetV1	70.6	113ms	-	-	x	-
	MobilenetV2	72.0	75ms	-	-	x	-
NAS	NASNet-A	74.0	183ms	x	x	x	48000
	AmoebaNet-A	74.4	190ms	x	x	x	75600
	MNasNet	74.0	76ms	yes	x	x	40000
ProxylessNAS	ProxylessNAS-G	71.8	83ms	yes	yes	yes	200
	ProxylessNAS-G + LL	74.2	79ms	yes	Yes	yes	200
	ProxylessNAS-R	74.6	78ms	yes	Yes	yes	200
	ProxylessNAS-R + MIXUP	75.1	78ms	yes	yes	yes	200

AutoML for Image Classification



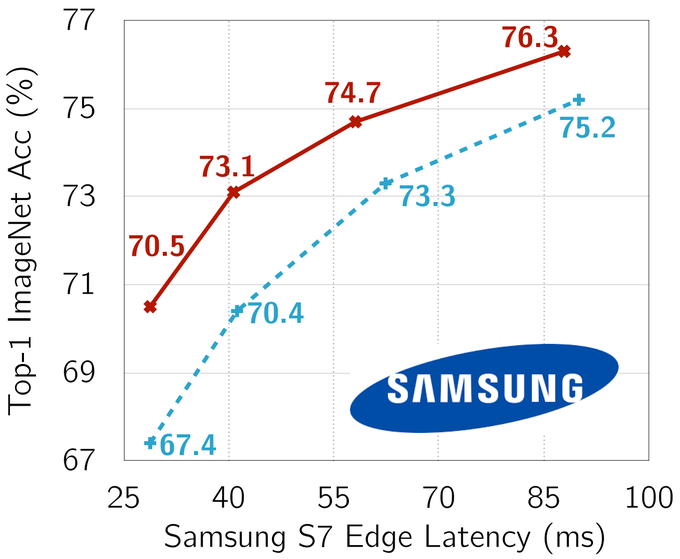
AutoML for Image Classification

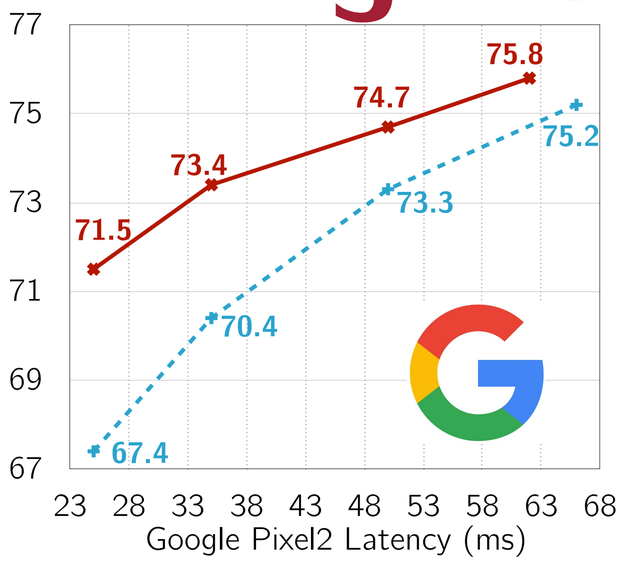


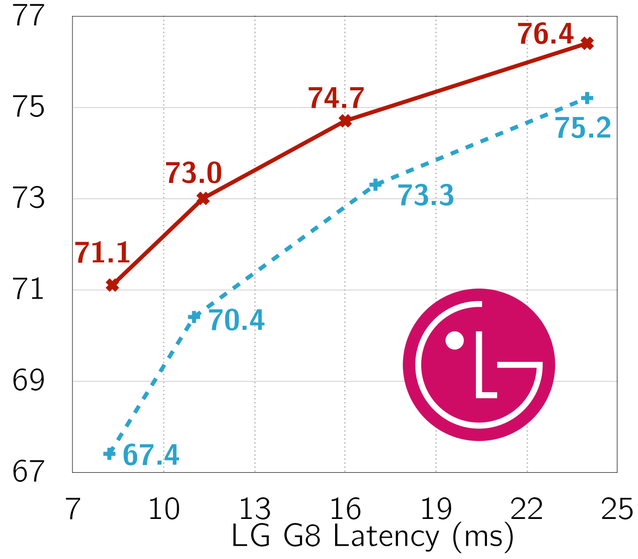


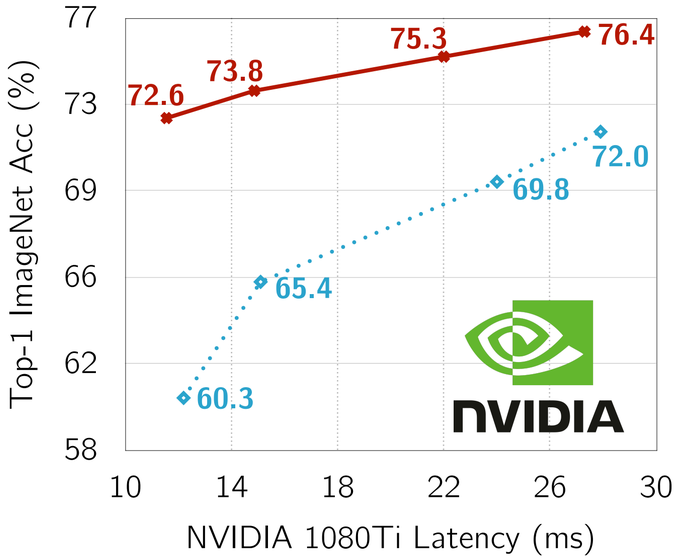
- Once-for-all is **2.6x faster** than EfficientNet and **1.5x faster** than MobileNetV3 on Google Pixel1 without loss of accuracy.

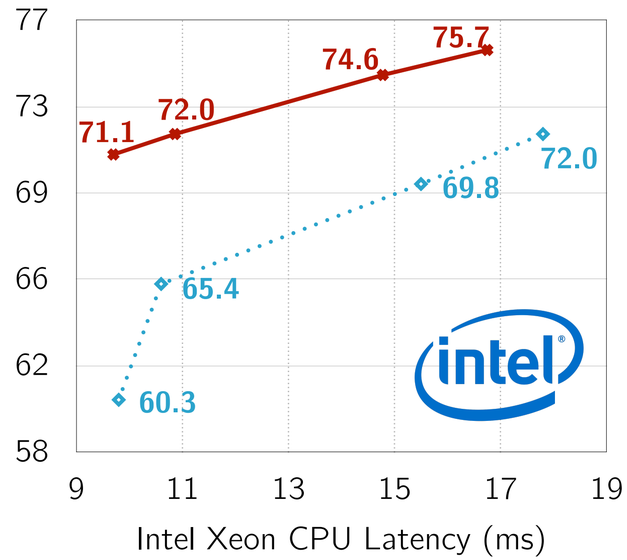
AutoML for Image Classification

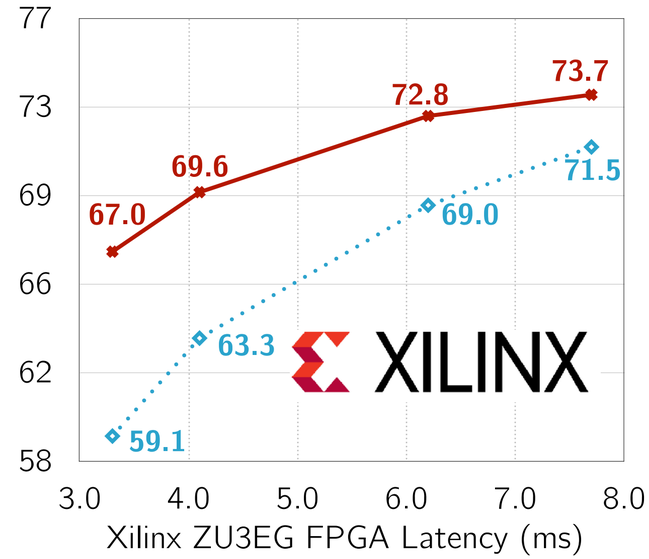




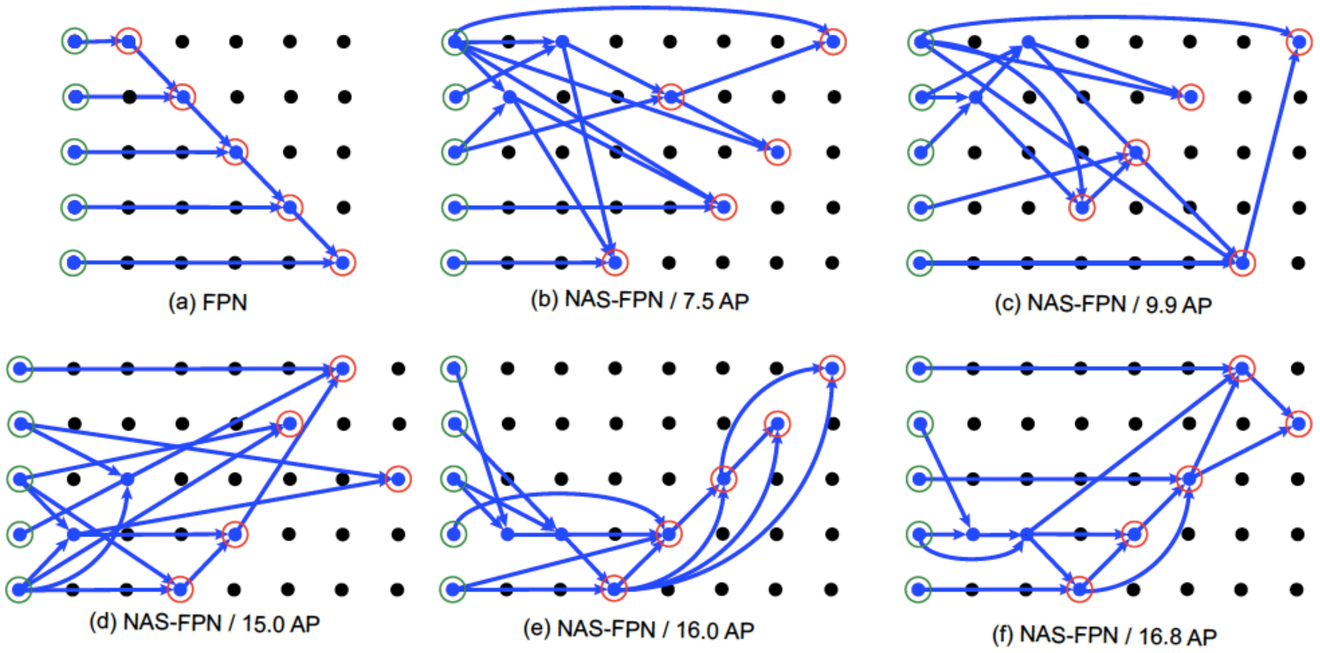




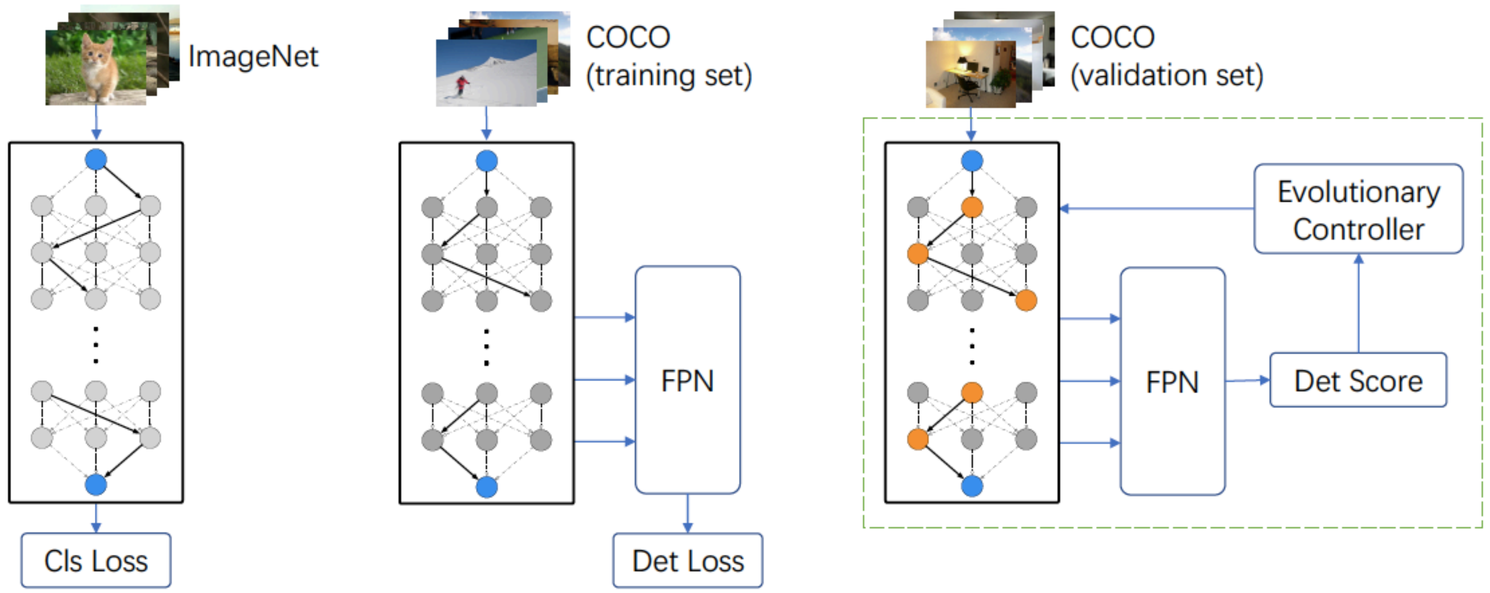




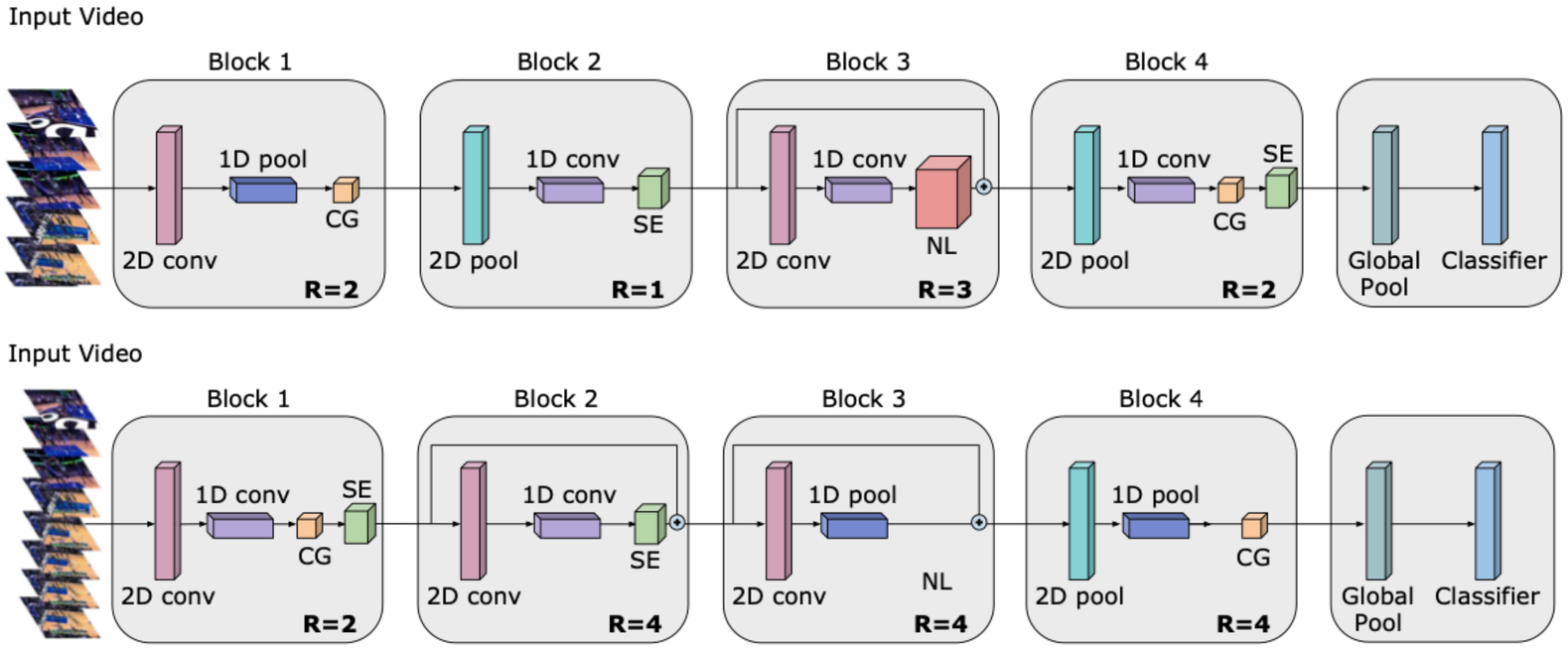
AutoML for Object Detection



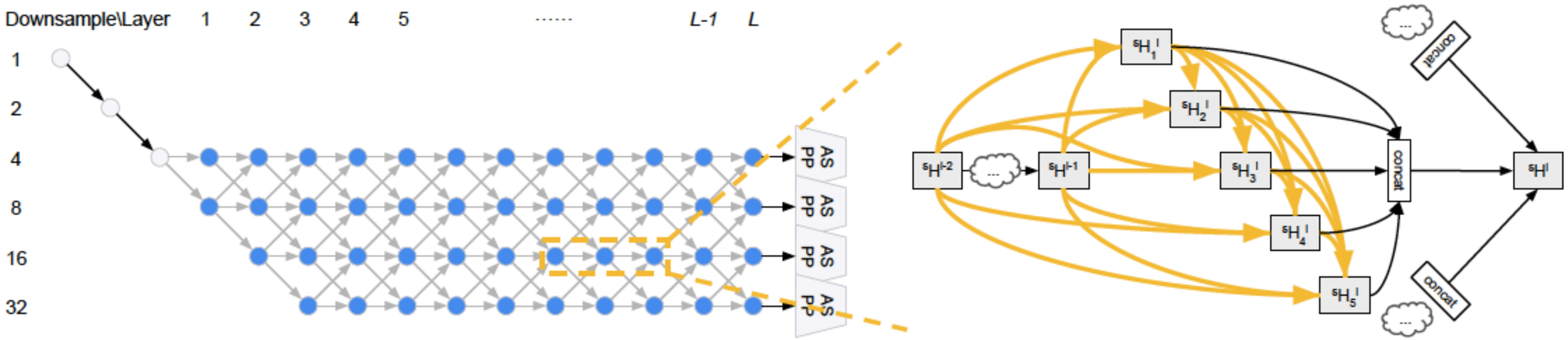
AutoML for Object Detection



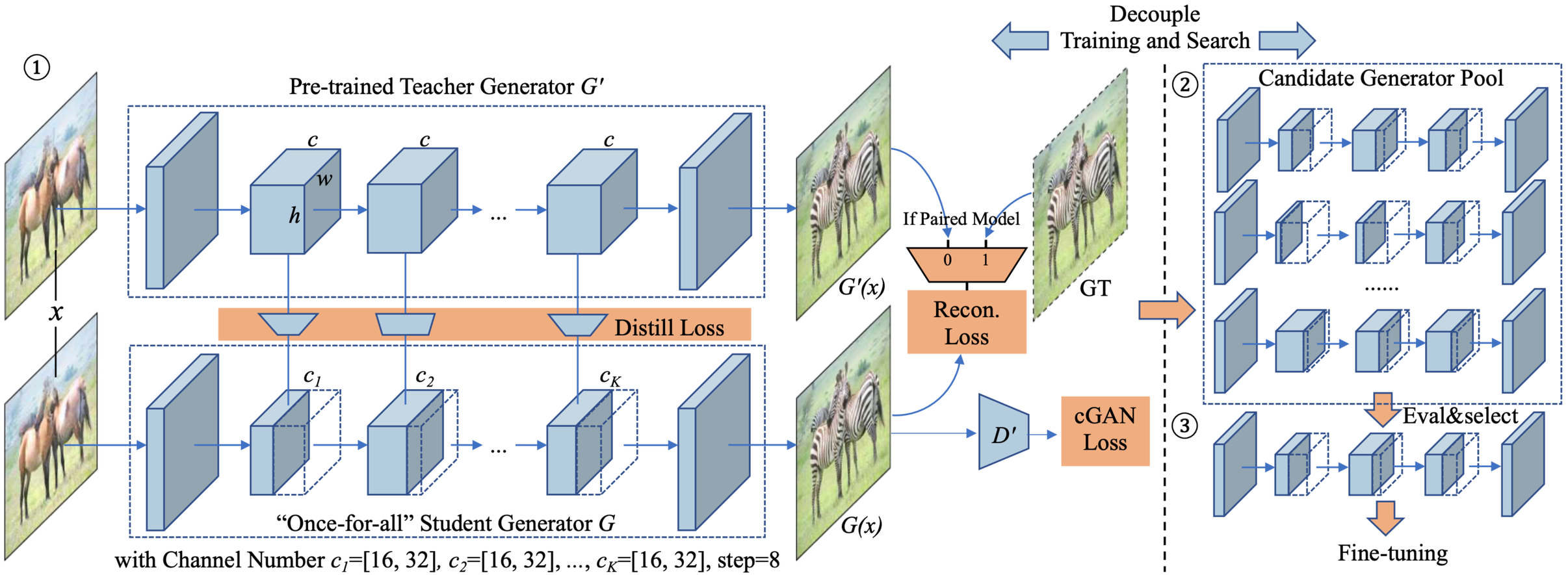
AutoML for Video Understanding



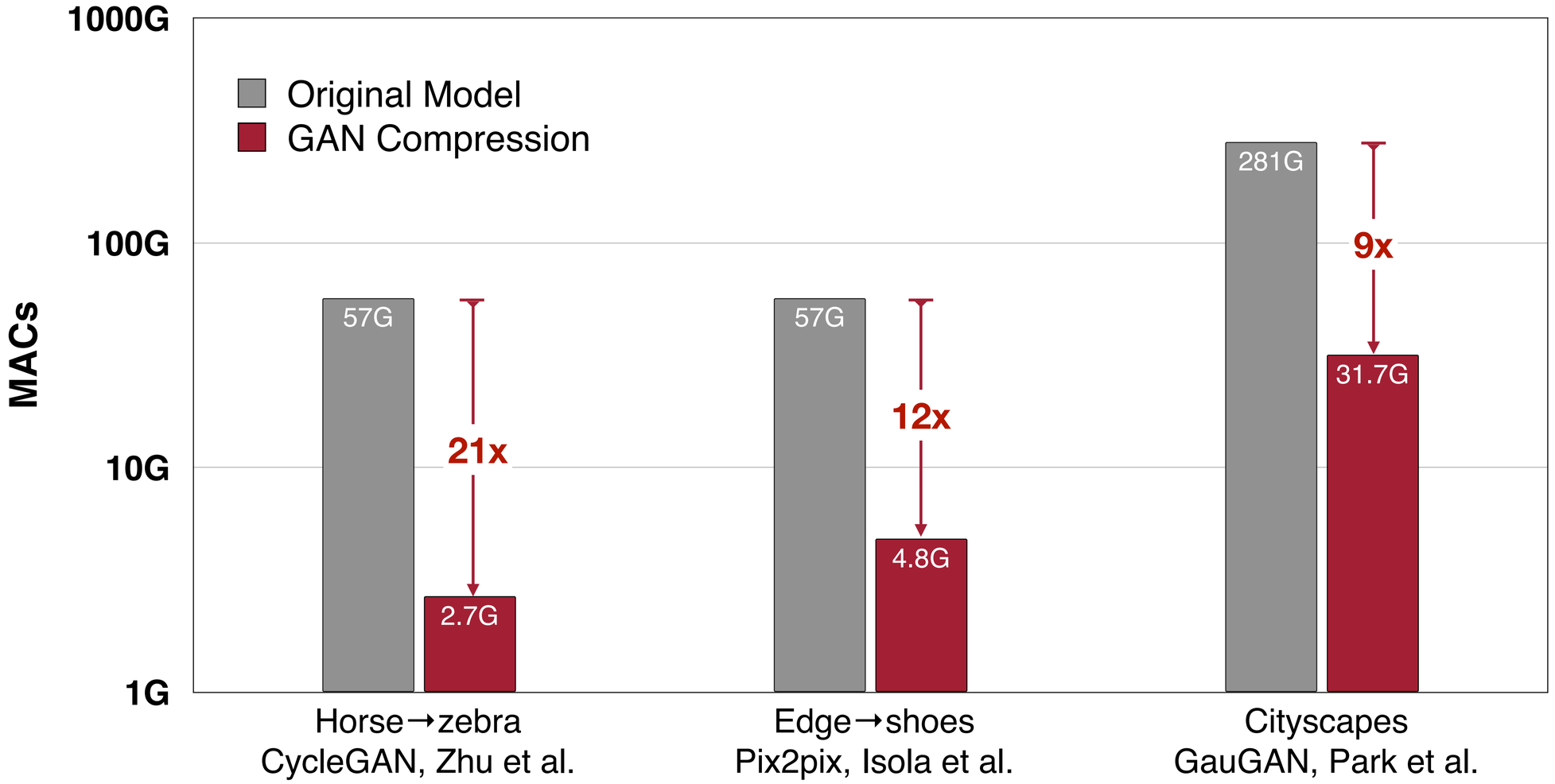
AutoML for Semantic Segmentation



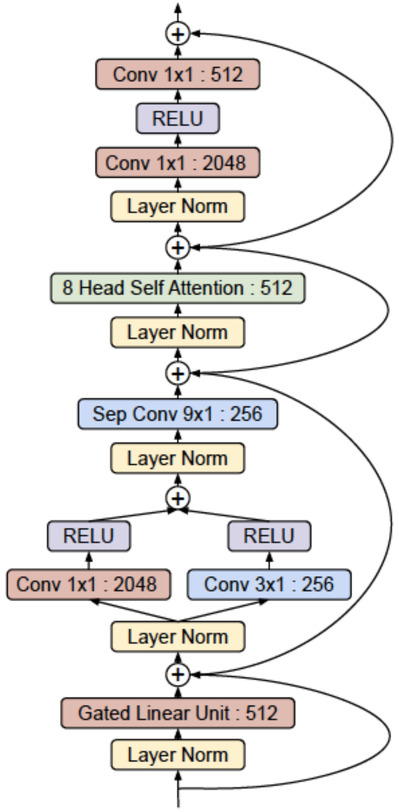
AutoML for GAN Compression

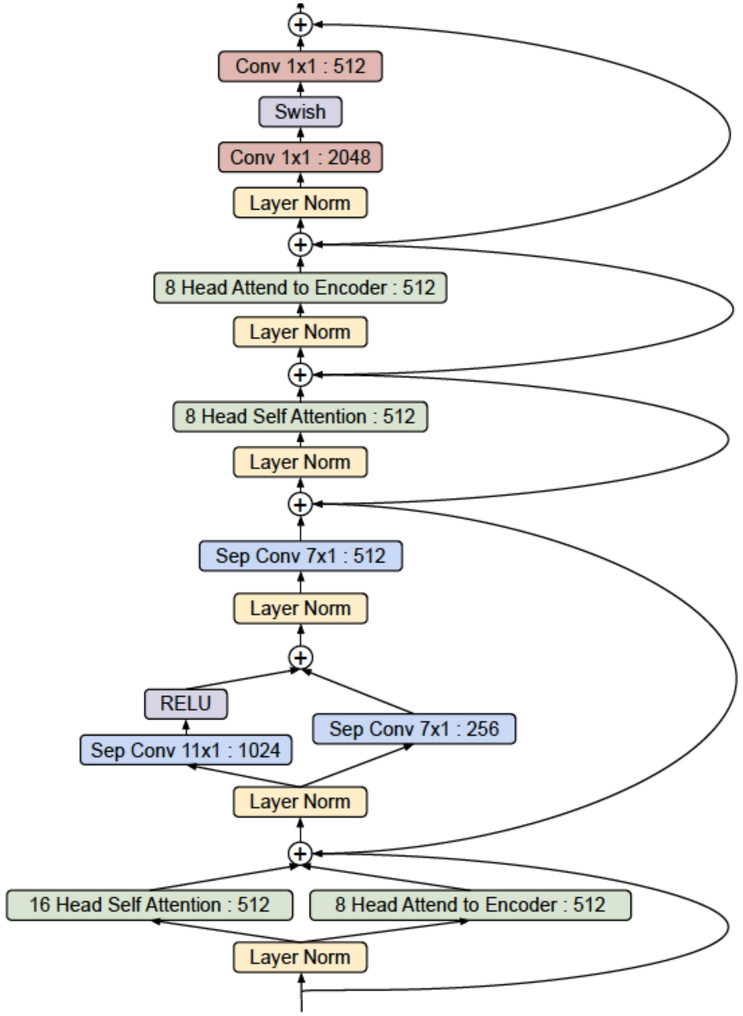


AutoML for GAN Compression

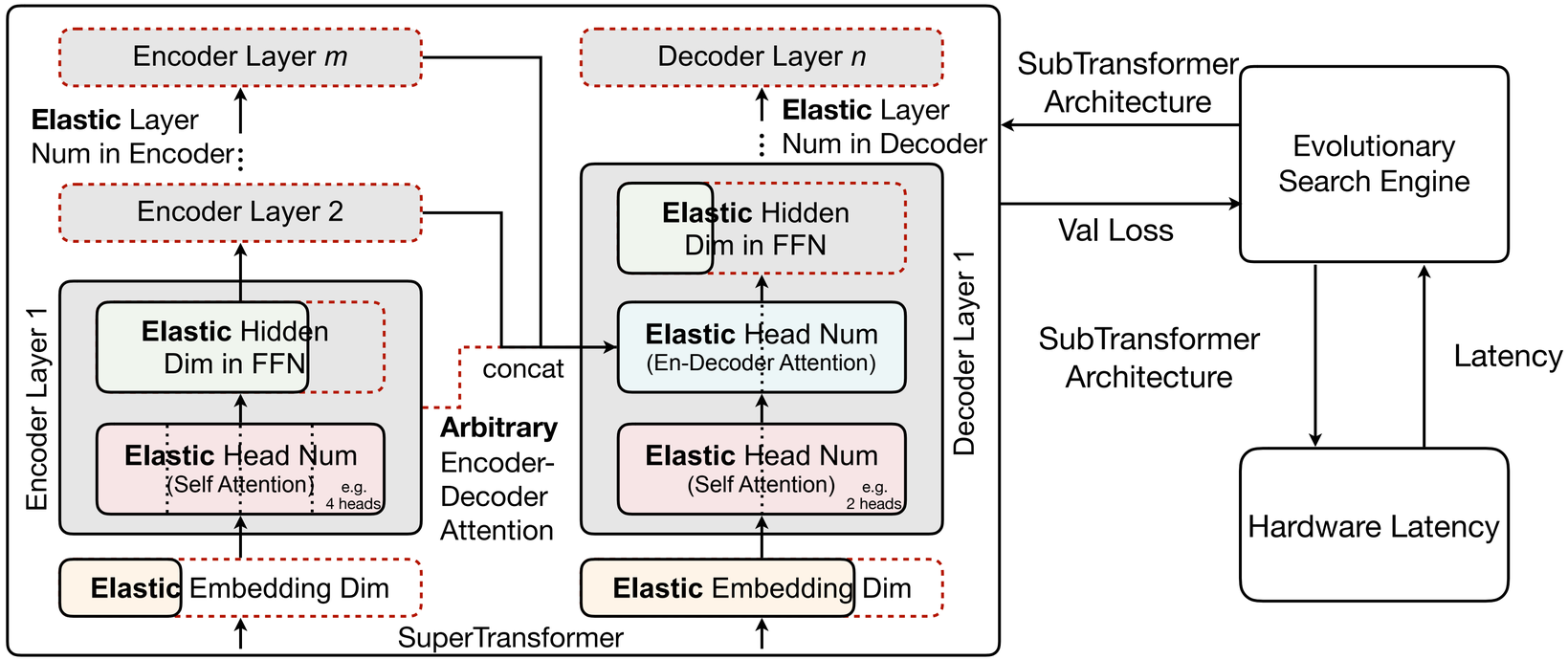


AutoML for NLP



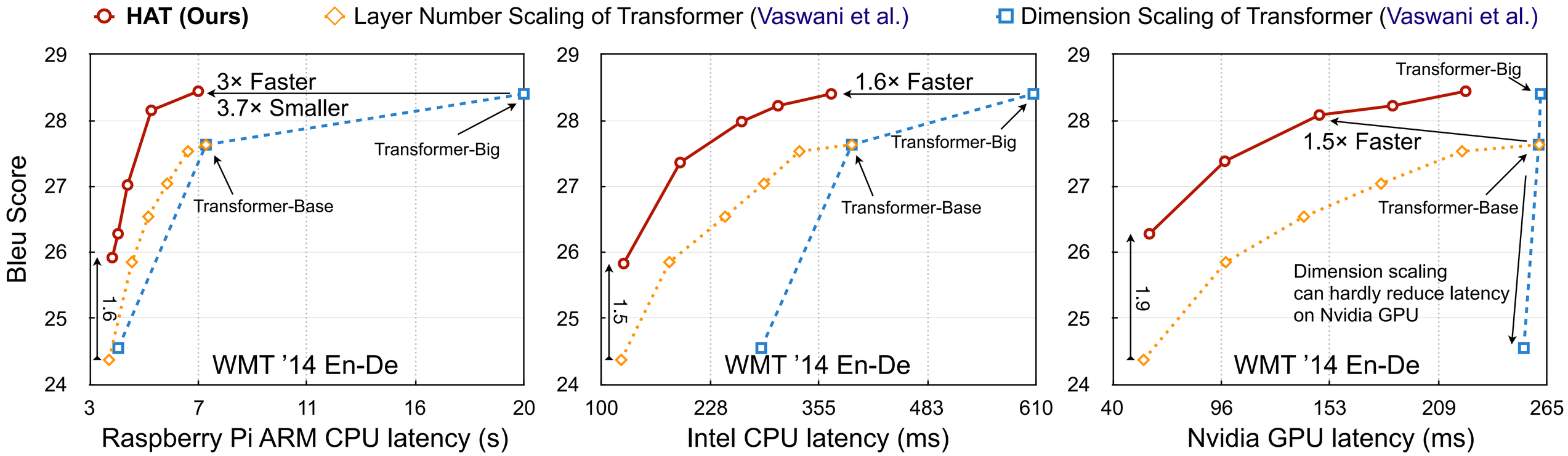


AutoML for NLP



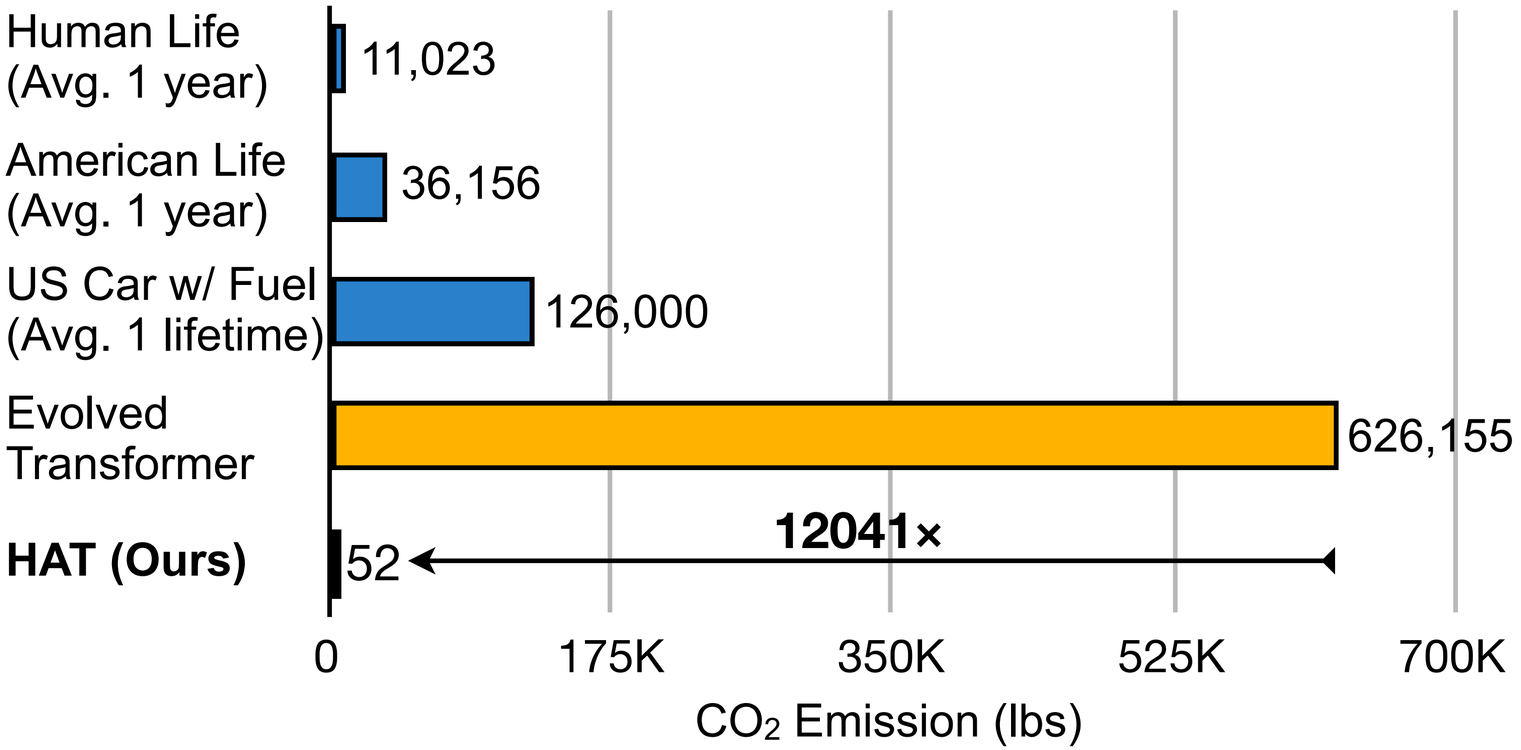
Training SuperTransformer by uniformly sampling and training all SubTransformers

AutoML for NLP



On WMT'14 En-De, HAT achieves same performance, **3.7x** smaller model size;
3x, 1.6x, 1.5x faster on Raspberry Pi, CPU, GPU, respectively than Transformer Baseline

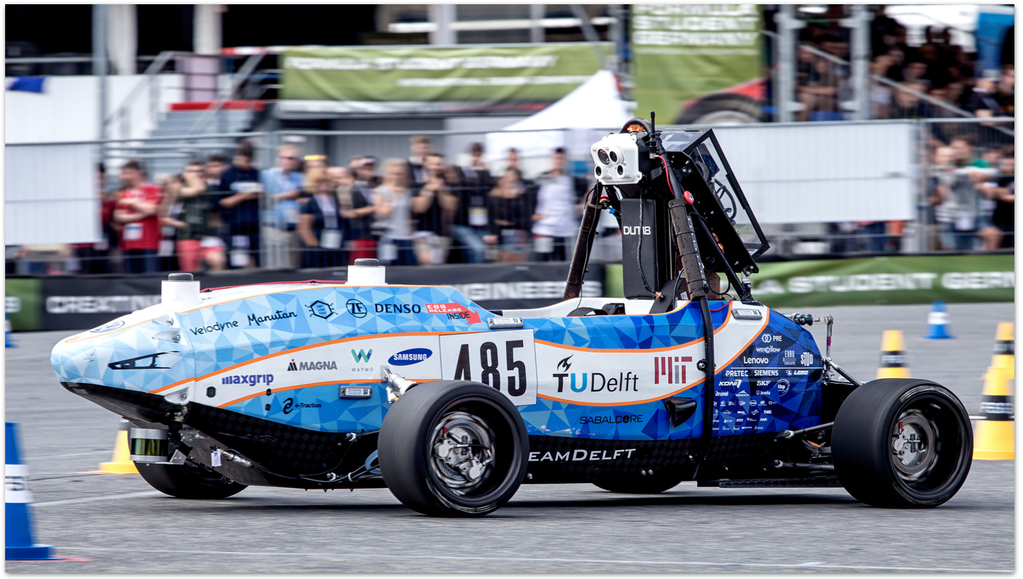
AutoML for NLP



CO_2 Emission of HAT training is only **52** pounds, while that of Evolved Transformer is **626,155** pounds

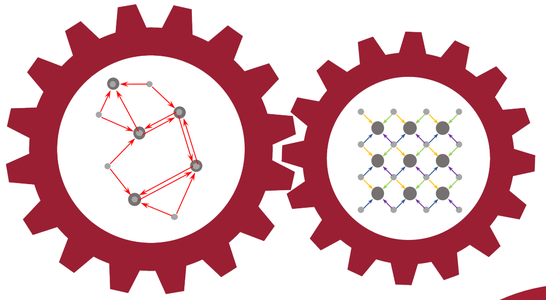
3D Deep Learning Has Broad Applications

VR/AR



Autonomous Driving

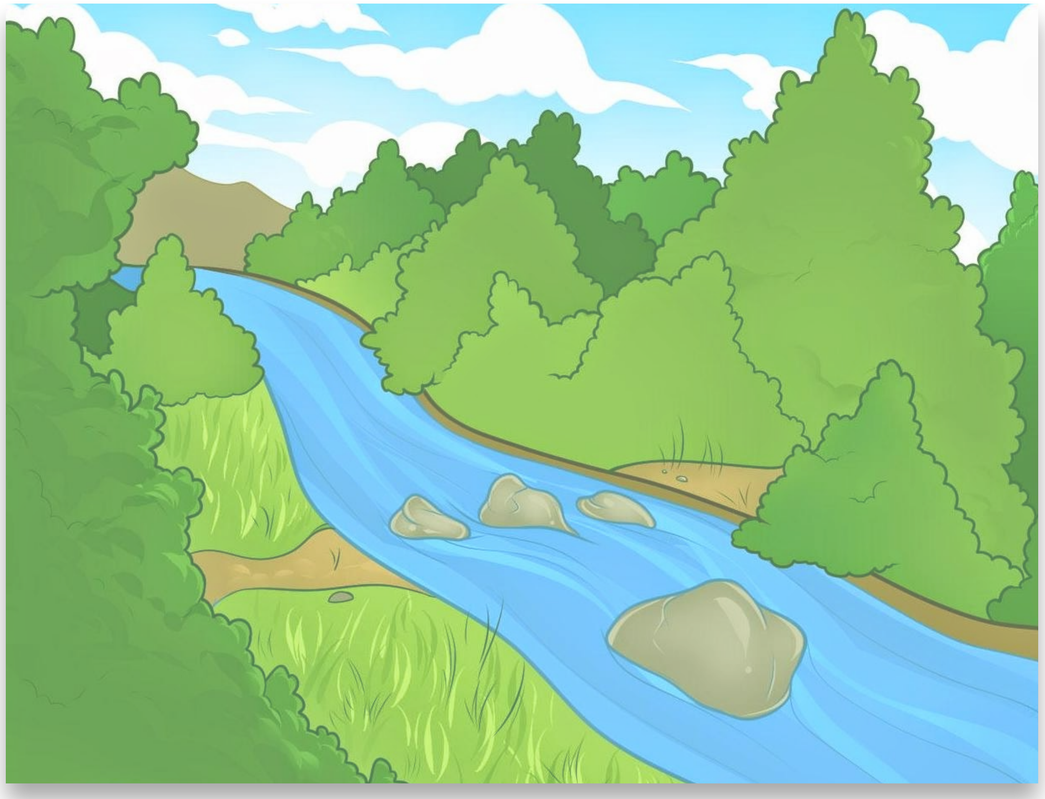
AutoML for 3D: Challenges

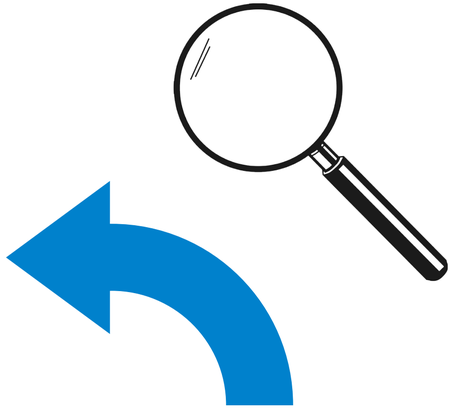


Upper-Stream

New Design Space

New Modules





Down-Stream

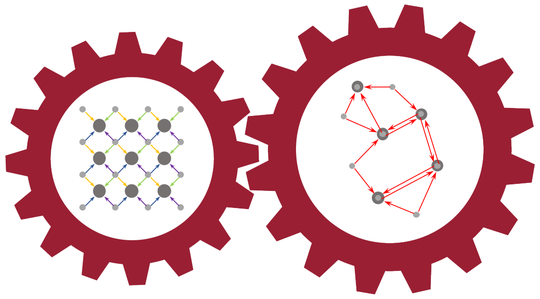
Efficient Search Algorithm

AutoML for 3D: Challenges

Upper-Stream

Design Space

New Modules

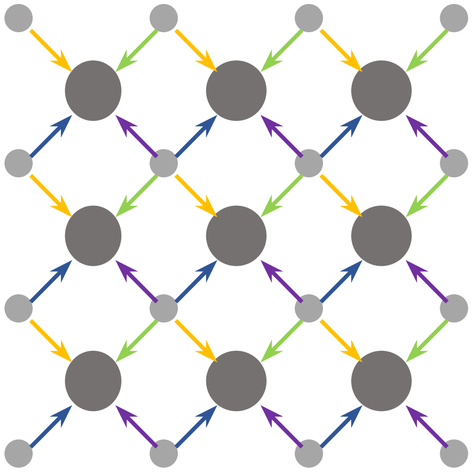


Down-Stream

Search Algorithm

We need both **down-stream design** and **upper-stream design** in AutoML for 3D.

Background: 3D Deep Learning Models

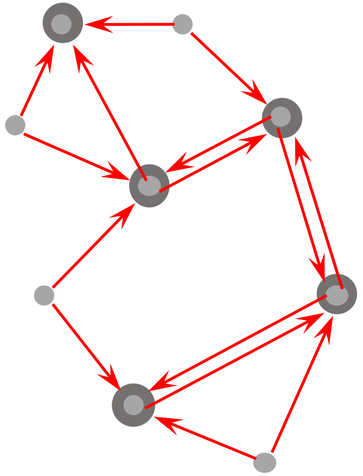


Voxel-Based Approaches

3D ShapeNets [CVPR'15]

VoxNet [IROS'15] **3D U-Net** [MICCAI'16]

Volumetric CNNs [CVPR'16]



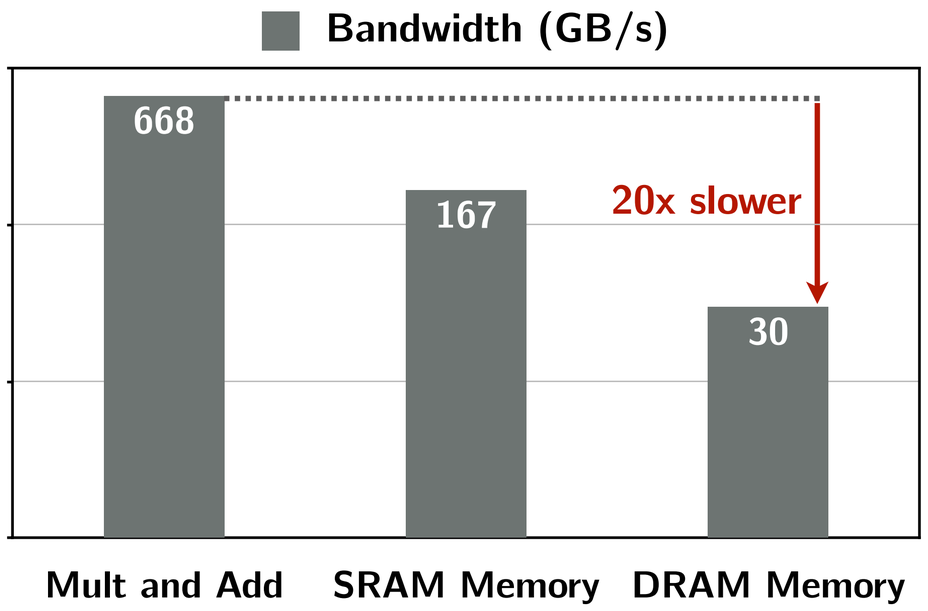
Point-Based Approaches

PointNet [CVPR'17] **PointNet++** [NIPS'17]

PointCNN [NeurIPS'18] **SpiderCNN** [ECCV'18]

DGCNN [SIGGRAPH'19]

Efficient 3D Deep Learning: Bottleneck



Sequential Memory Access

1	2	3	4	5	6	7	8

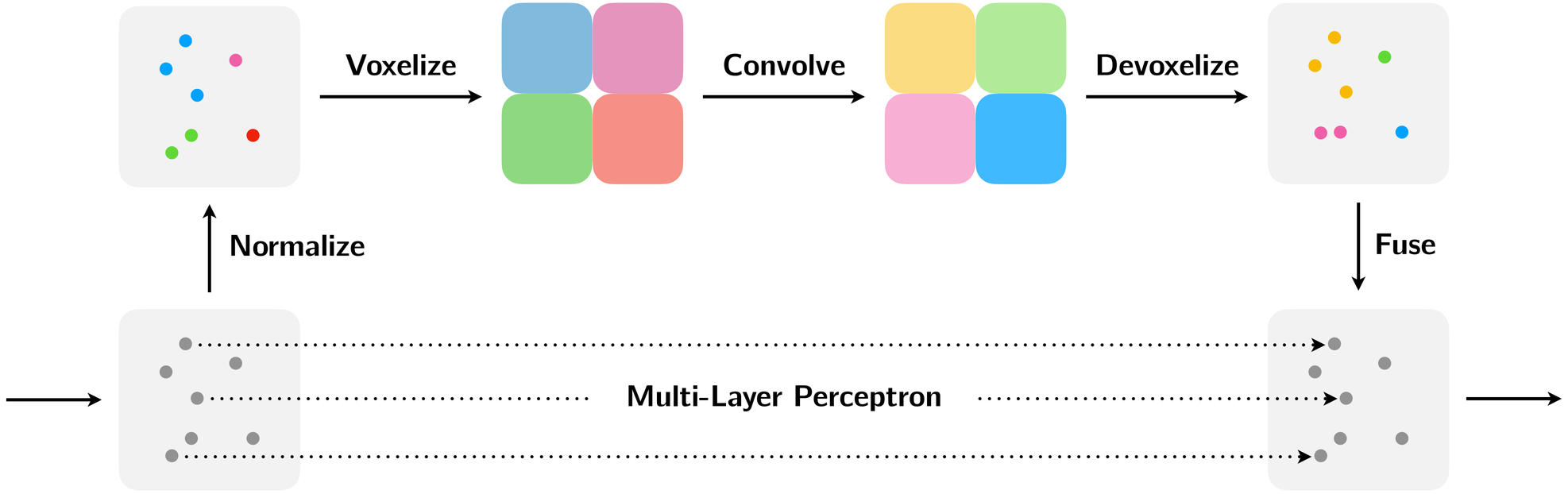
Random Memory Access

7	5	2	4	6	1	8	3

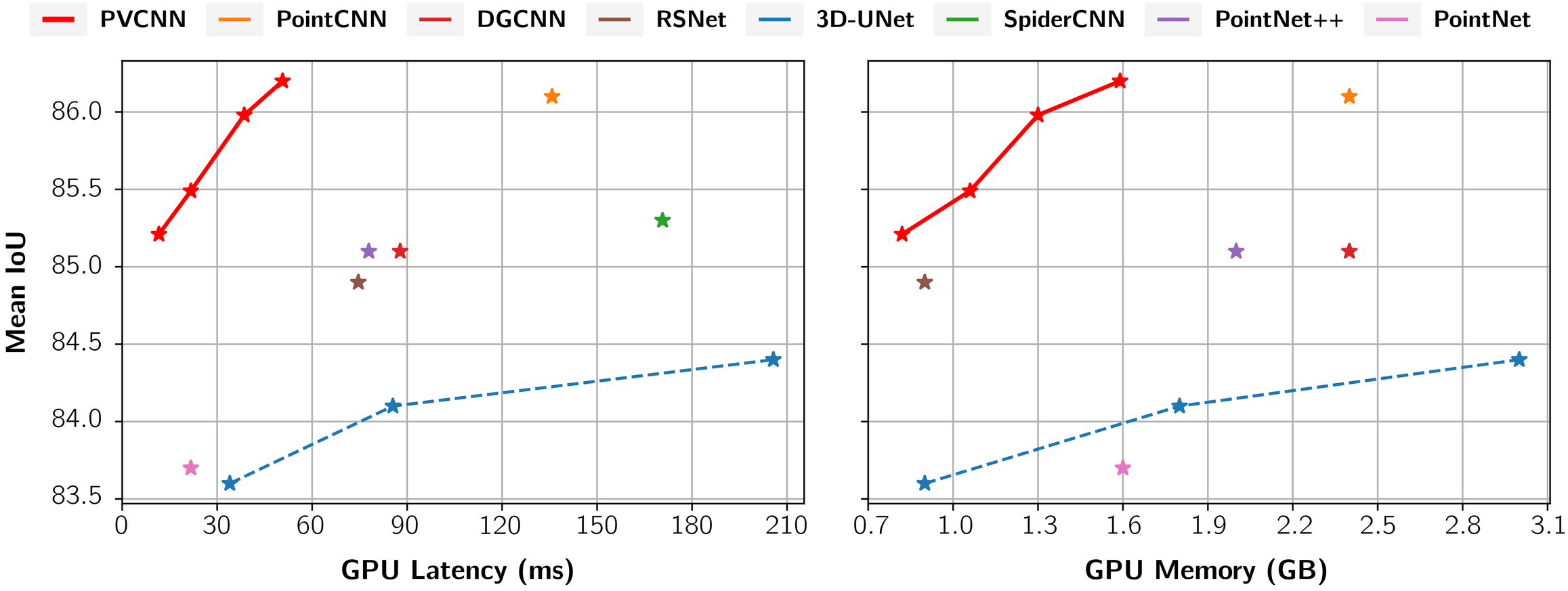
Off-chip DRAM access is much more expensive than arithmetic operation!

Random memory access is inefficient due to the potential bank conflicts!

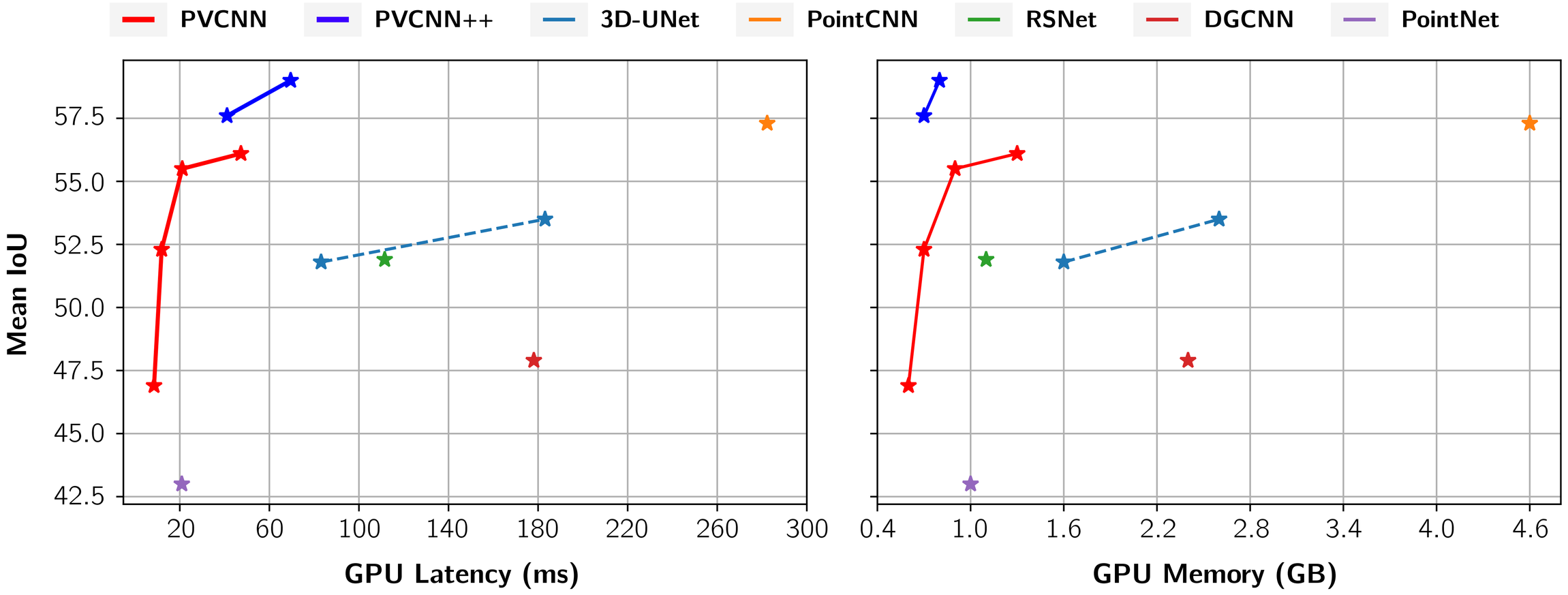
Point-Voxel Convolution (PVConv)



Results: 3D Part Segmentation (ShapeNet)



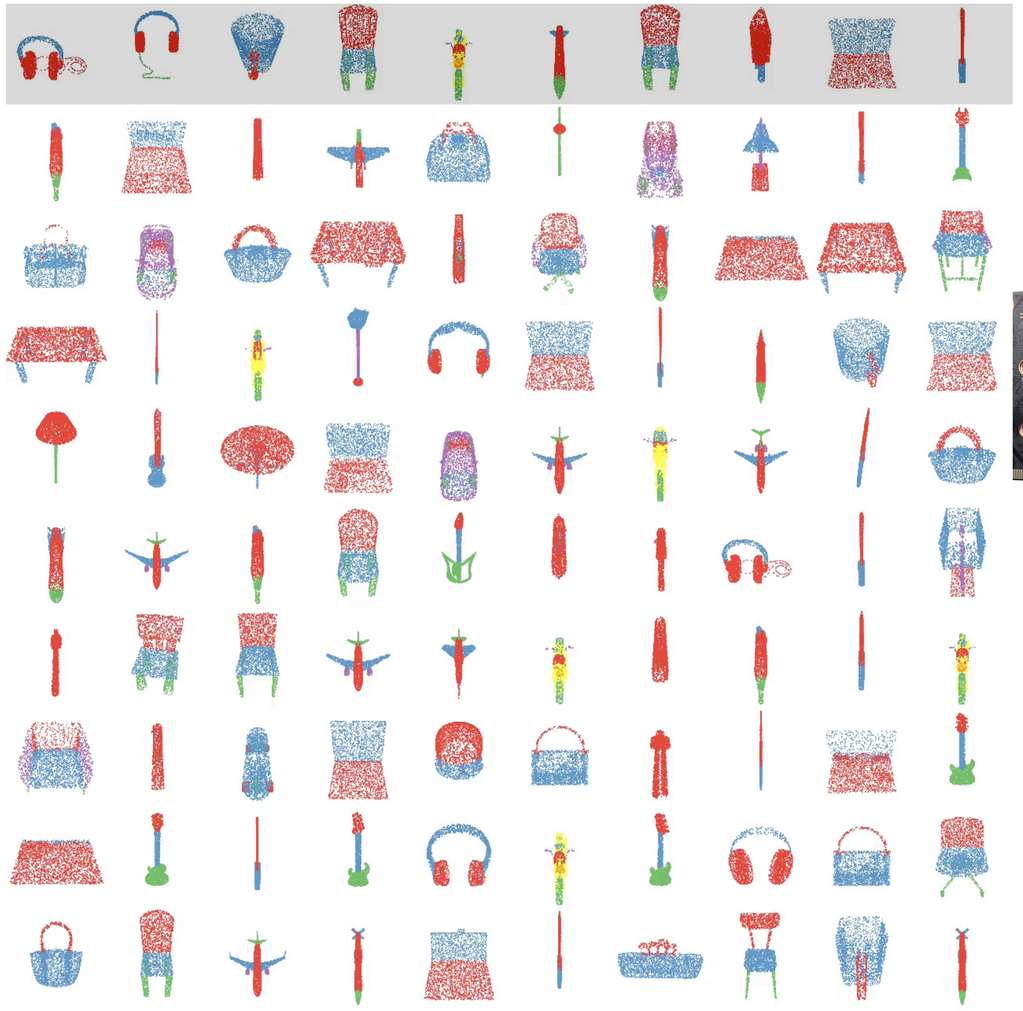
Results: 3D Semantic Segmentation (S3DIS)

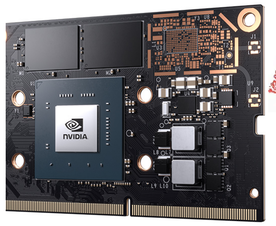


Results: 3D Semantic Segmentation on Edge Devices

PVCNN (IoU = **85.1%**)**:**

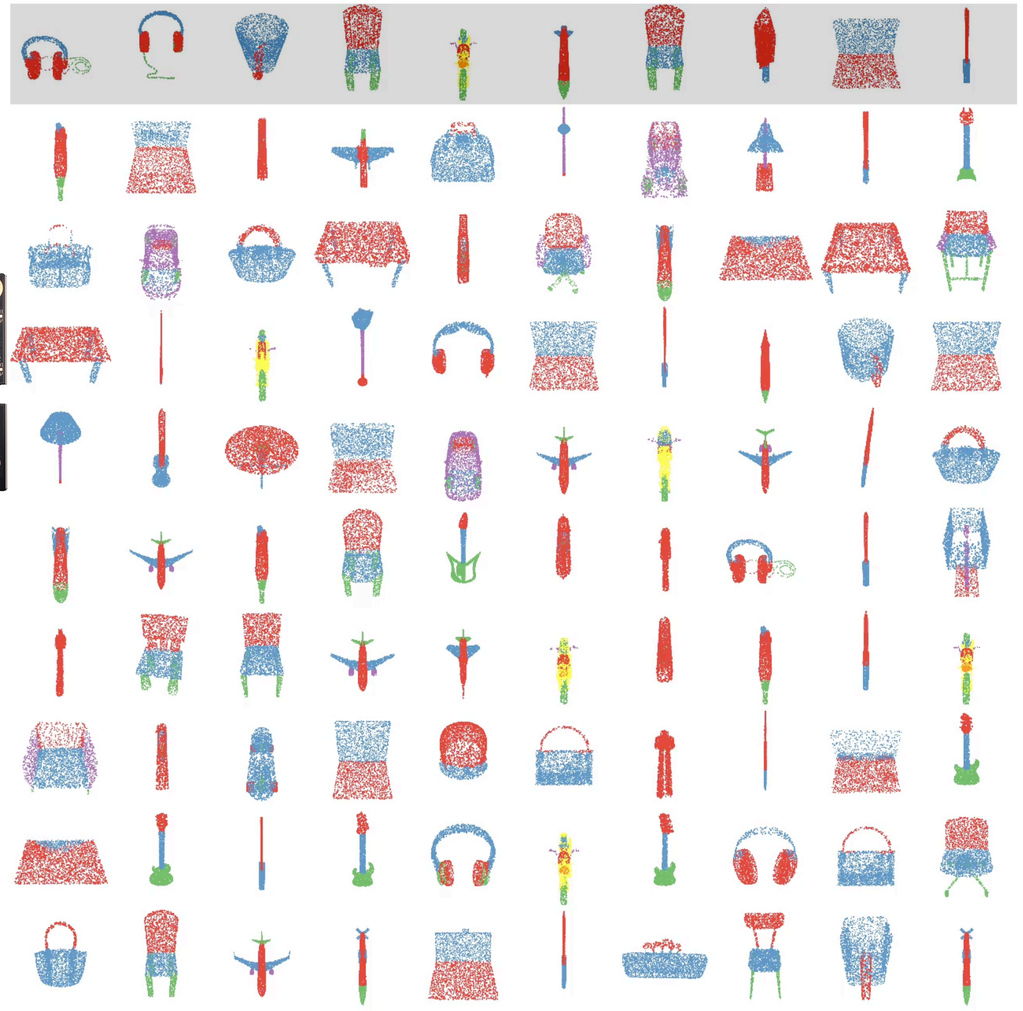
Throughput: **19.9** objects / sec





PointNet (IoU = **83.7%**)**:**

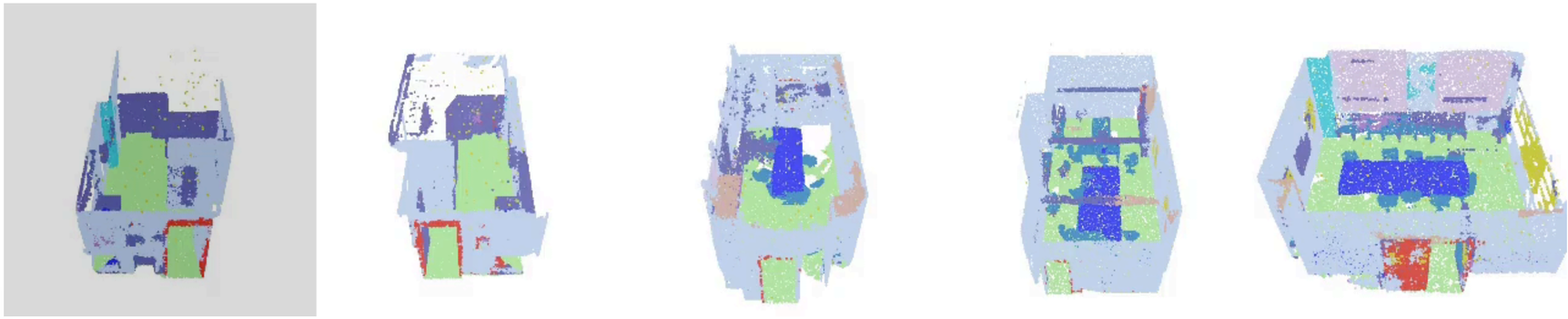
Throughput: **8.2** objects / sec



Results: 3D Semantic Segmentation on Edge Devices

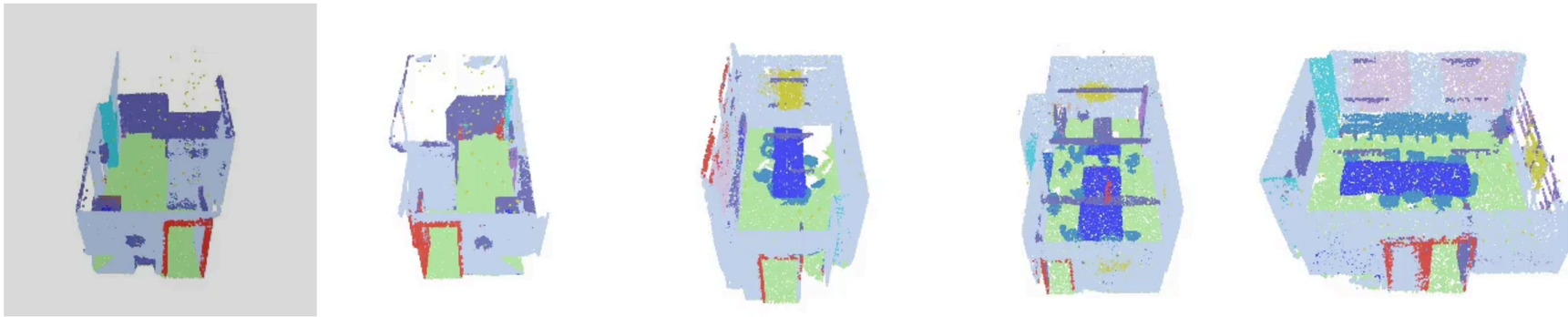
PointNet (IoU = **43.0** %)**:**

Latency: **4131.8** ms / room

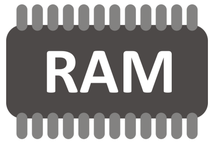


PVCNN (IoU = **52.3** %)**:**

Latency: **2748.8** ms / room

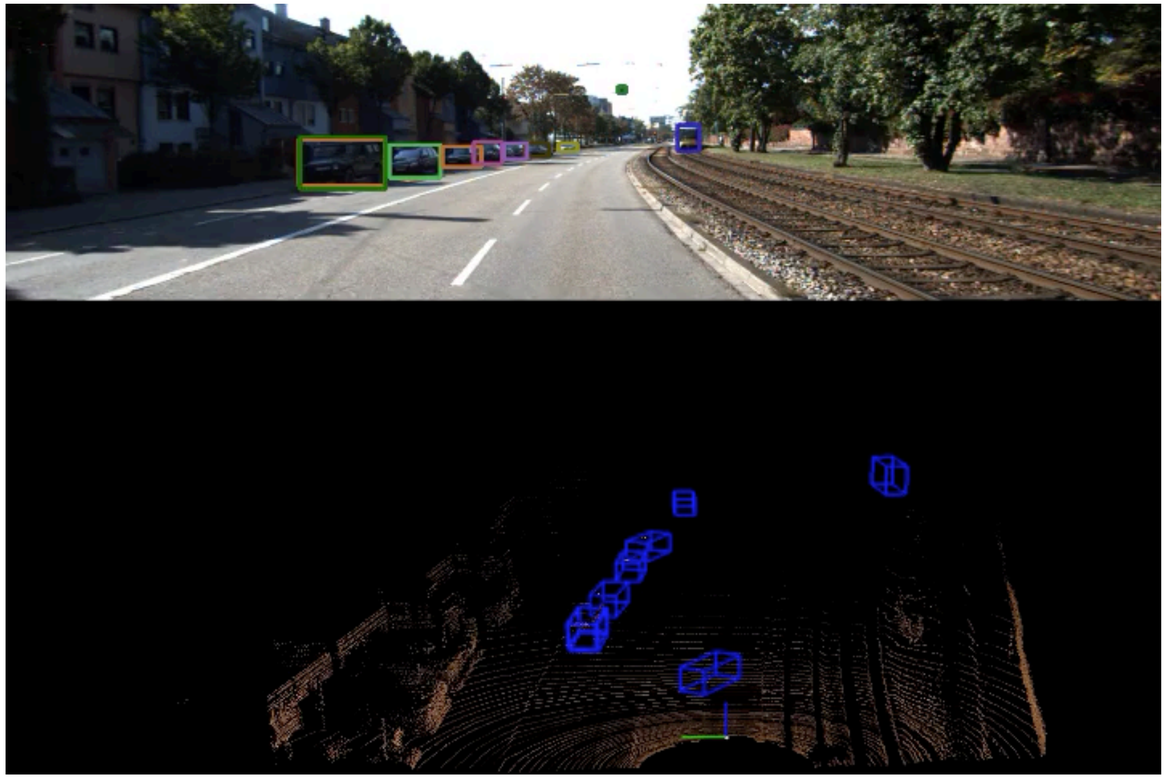


Results: 3D Object Detection (KITTI)



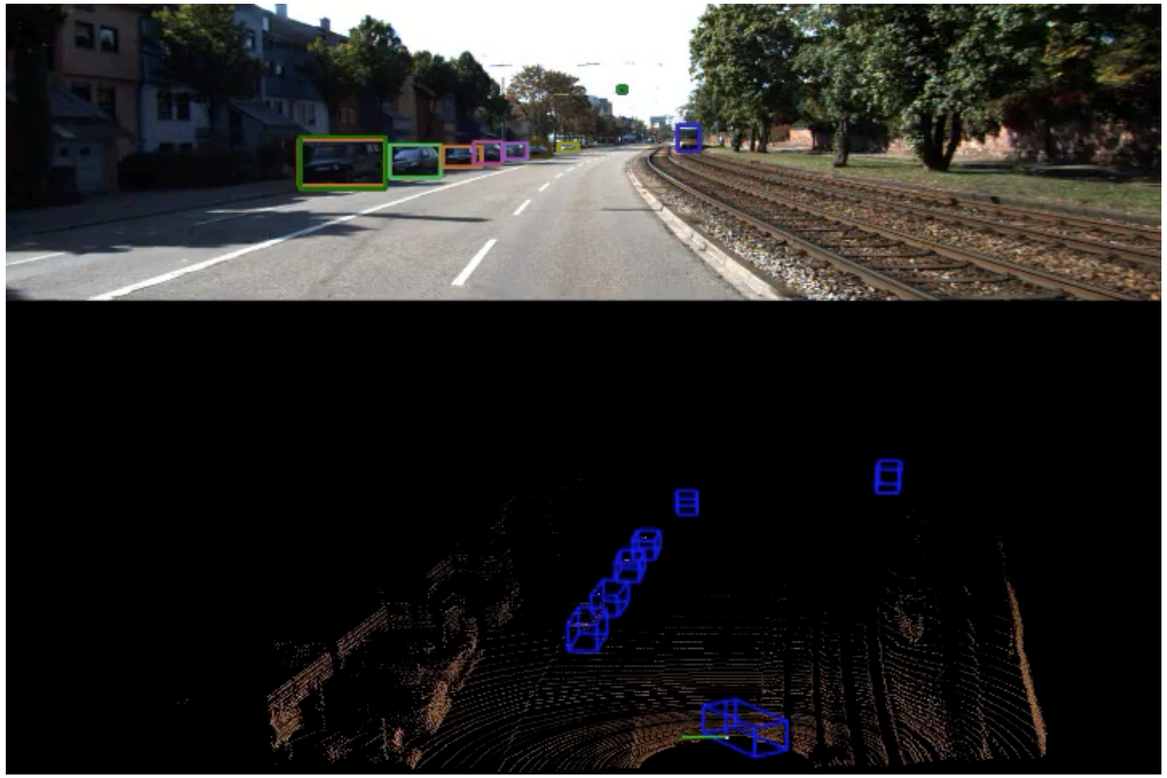
	GPU Latency	GPU Memory	Pedestrian	Cyclist	Car
F-PointNet++	105.2 ms	2.0 GB	61.6	62.4	72.8
PVCNN (efficient)	**58.9 ms** **(1.8x)**	**1.4 GB** **(1.4x)**	60.7 (-0.9)	63.6 (+1.2)	73.0 (+0.2)
PVCNN (complete)	**69.6 ms** **(1.5x)**	**1.4 GB** **(1.4x)**	**64.9** **(+3.3)**	**65.9** **(+3.5)**	**73.1** **(+0.3)**
	Faster	**Lower**	**More Accurate**		

Results: 3D Object Detection (KITTI)



F-PointNet++

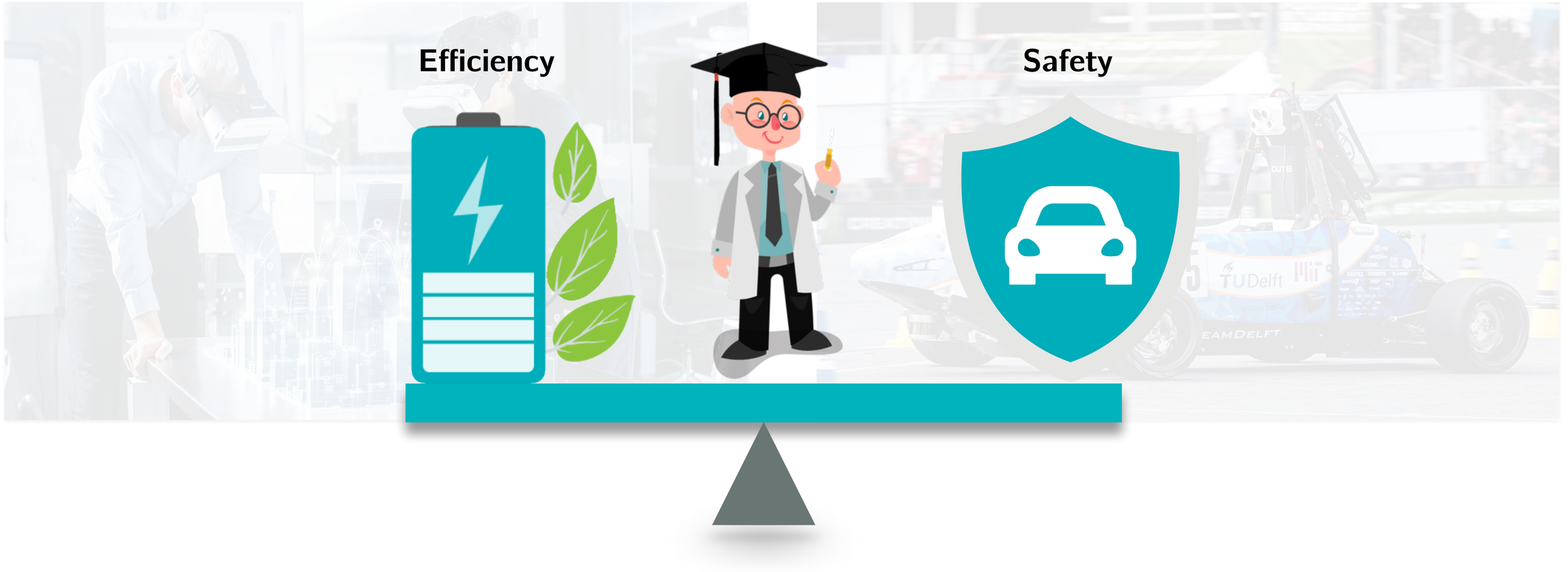
(10 batches per second)



PVCNN

(**17** batches per second, **1.8x** faster)

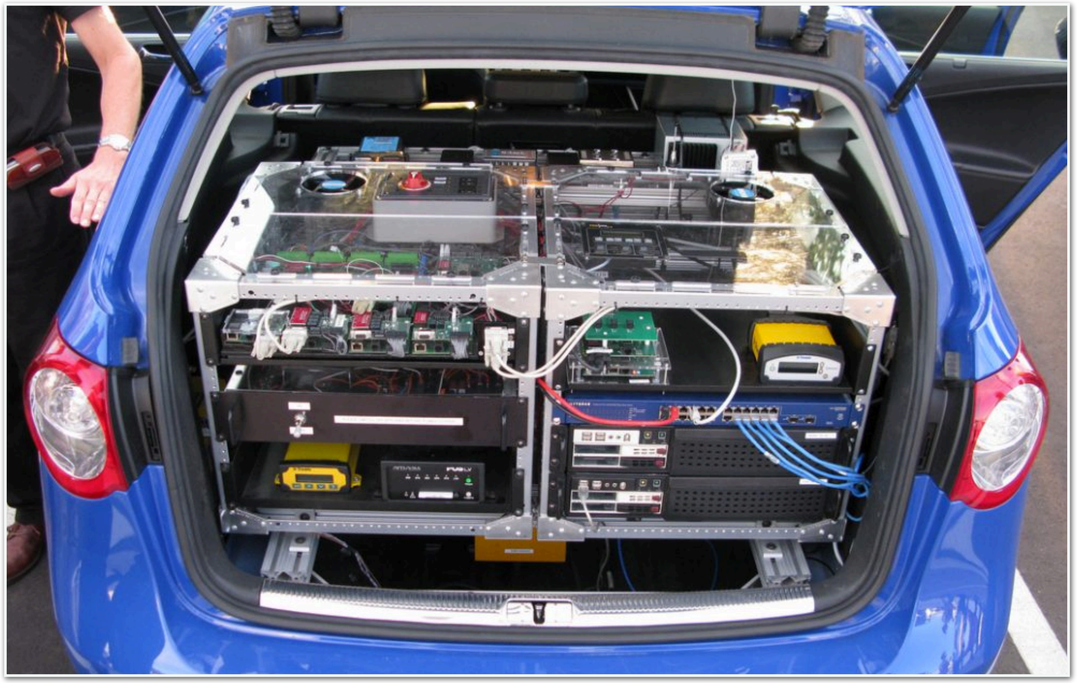
Efficiency and Safety are Equally Important



Limited Hardware Resources in Real World

VR/AR Headsets

A full backpack of computers!

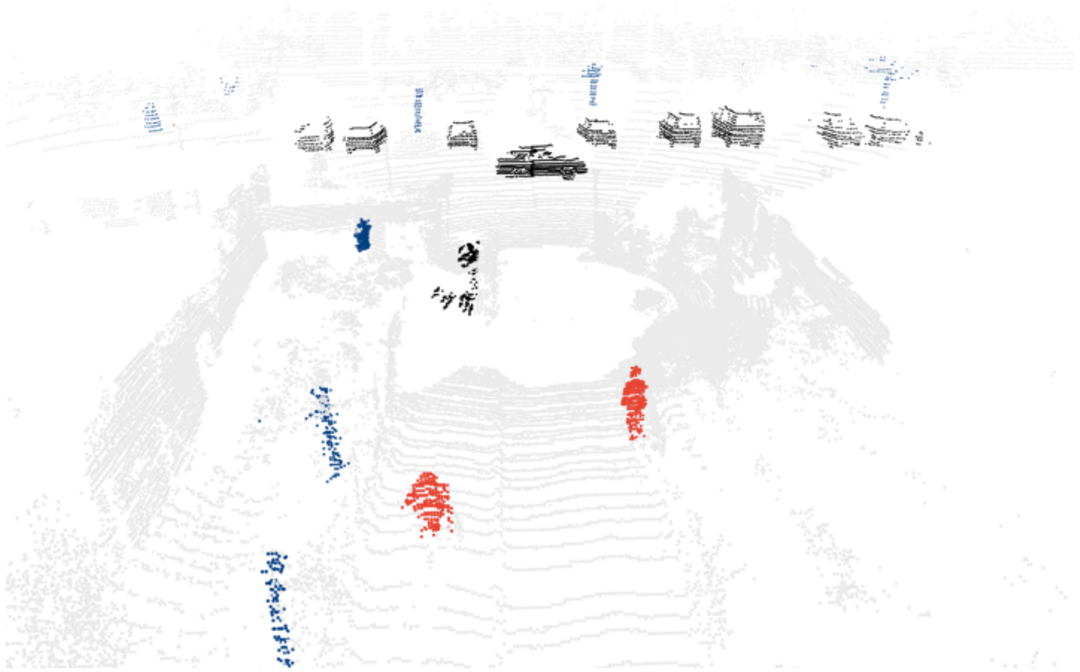


Self-Driving Cars

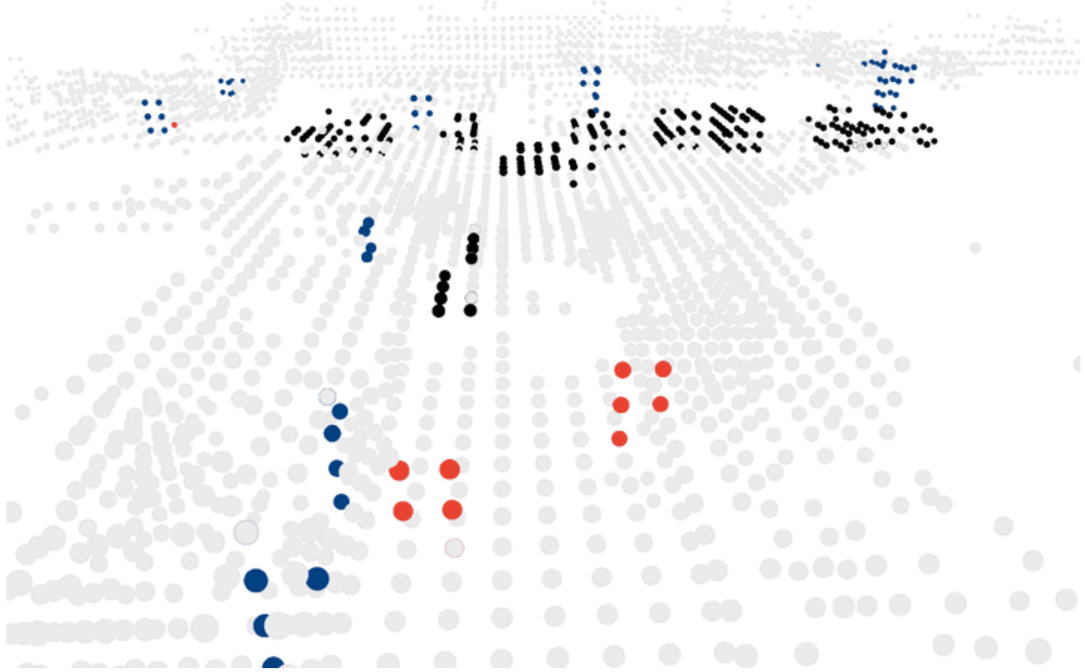
A whole trunk of computers!

We need **more efficient** algorithms that consumes **less computation**.

Low Resolution with Constrained Memory

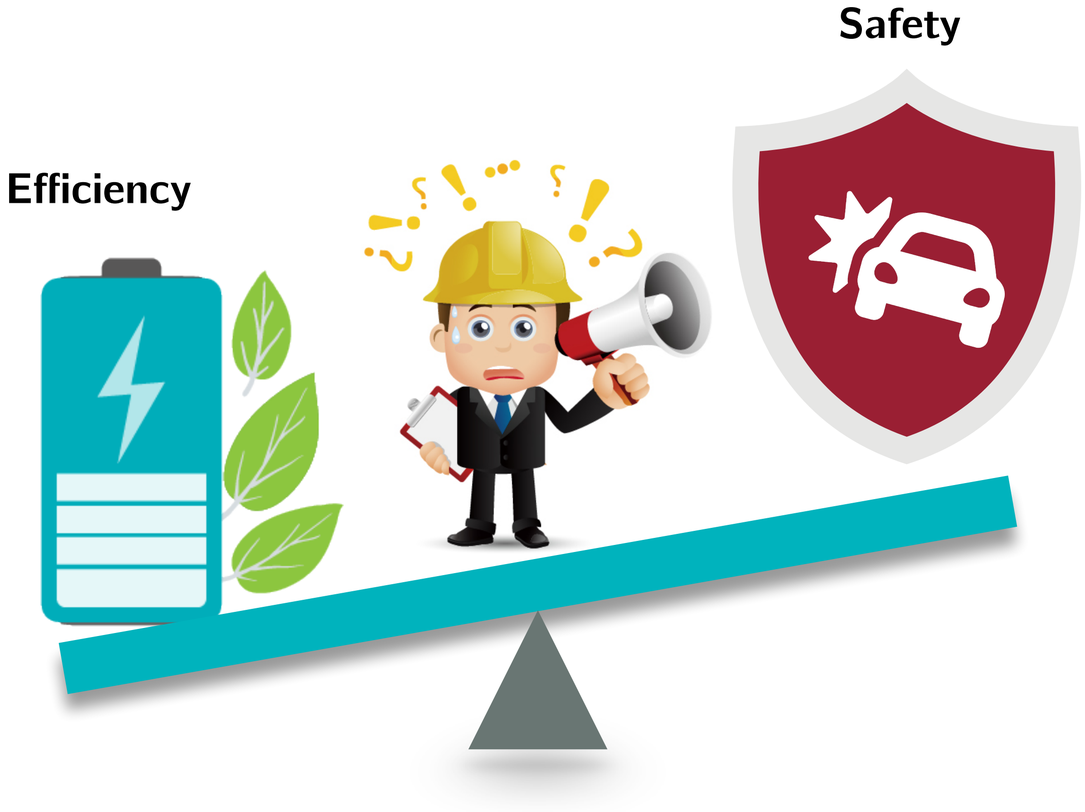


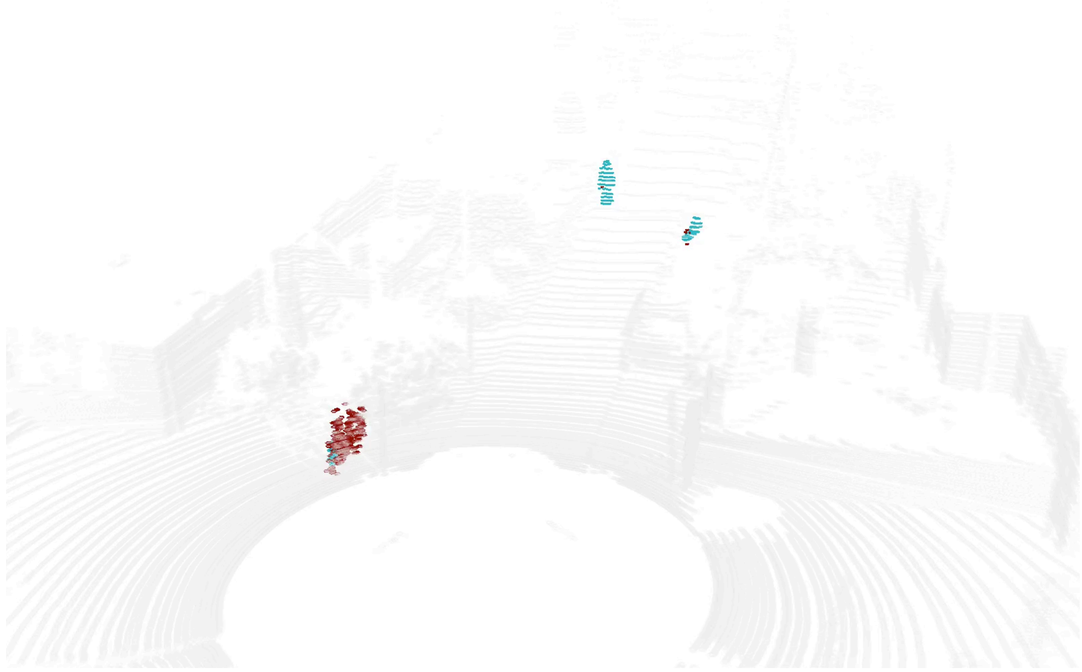
Original Scene



Downsampled Scene

Compromised Safety under Low Resolution

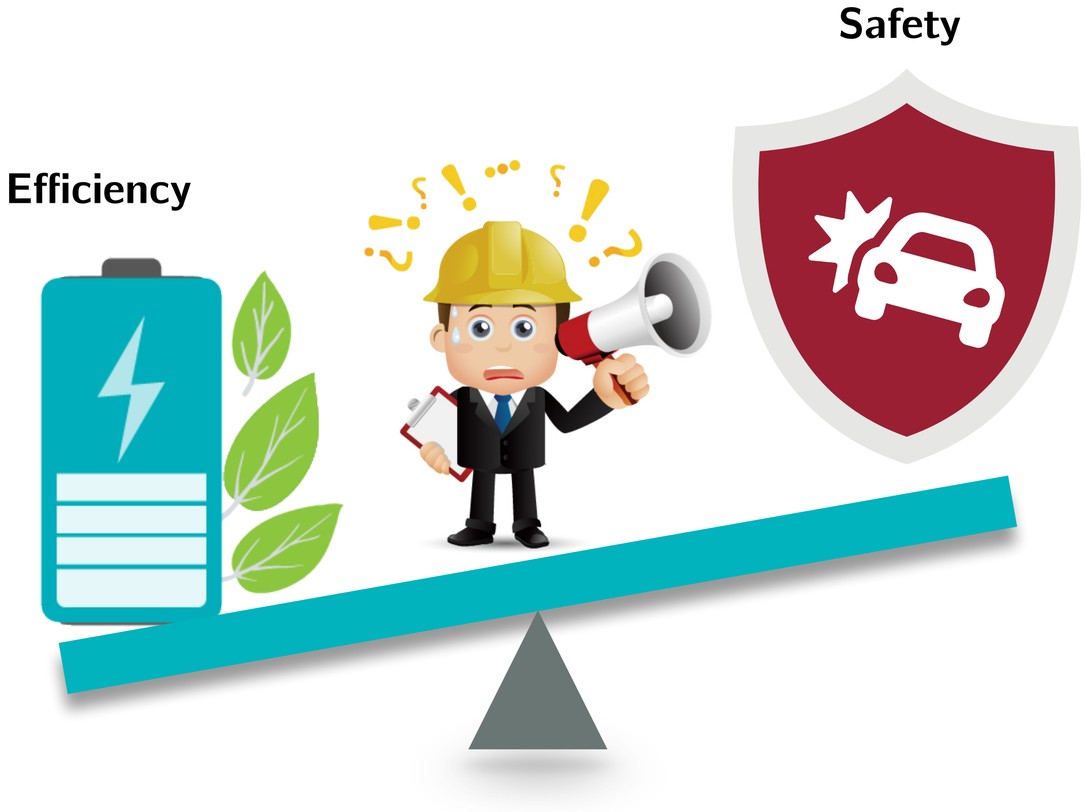


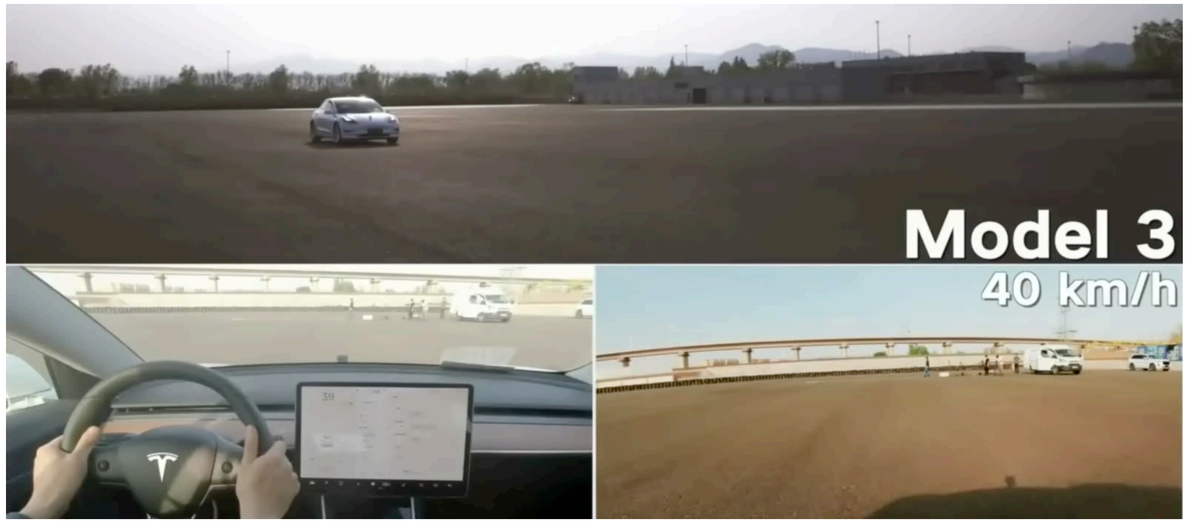


Green: Recognized **Red:** Failed

MinkowskiNet frequently fails to recognize a nearby bicycle and pedestrian.

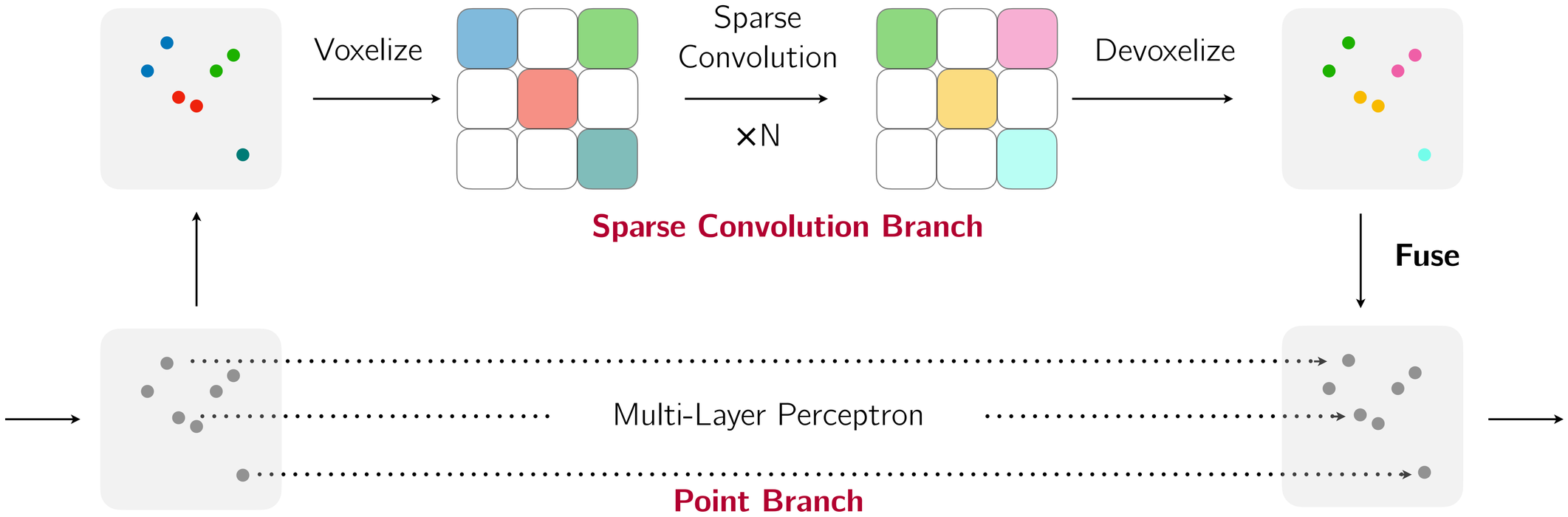
Compromised Safety under Low Resolution



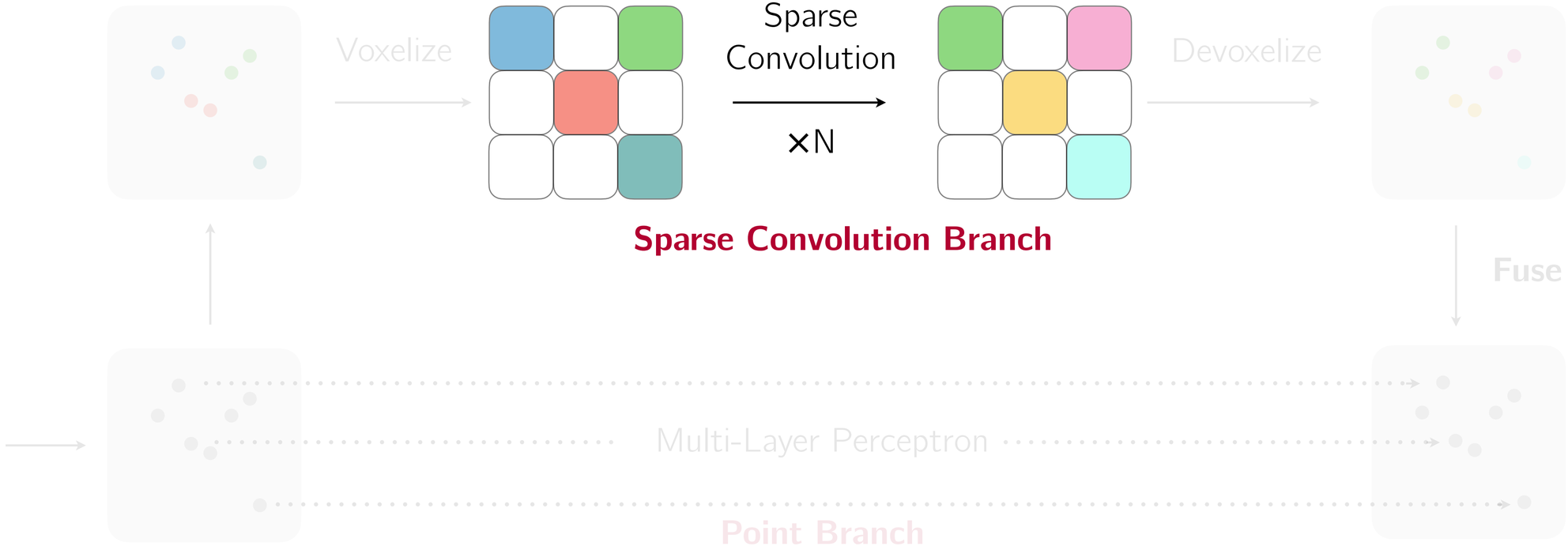


A real self-driving car **crashes** after ignoring small objects.

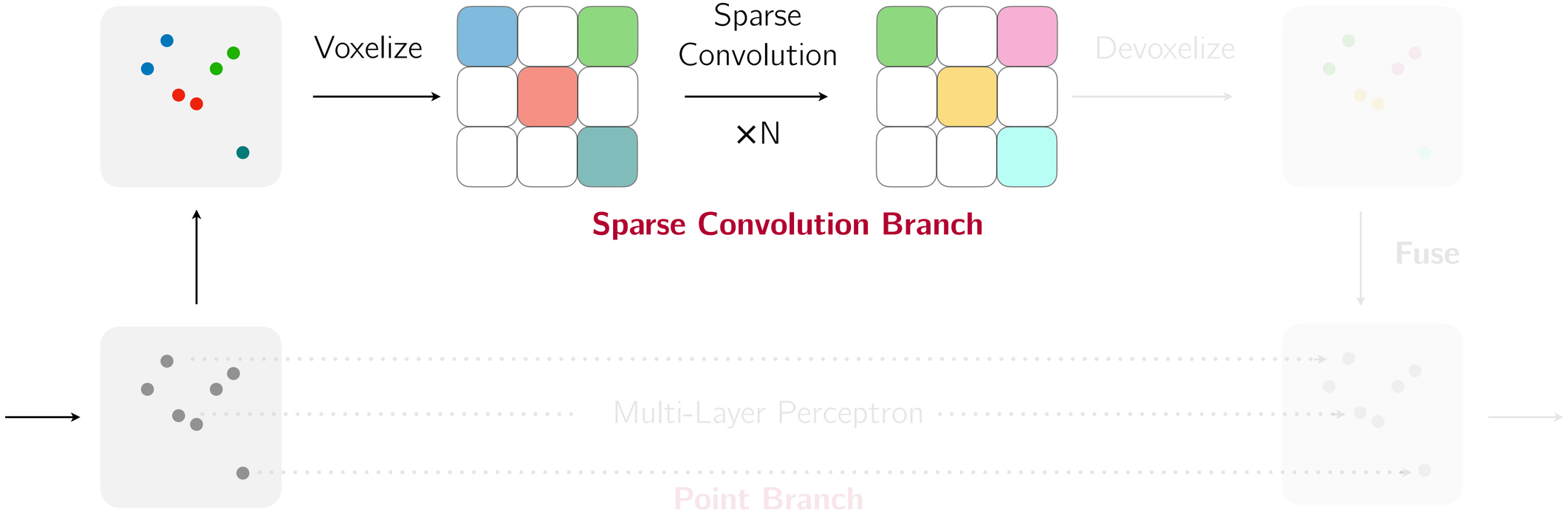
Designing Efficient 3D Modules (SPVConv)



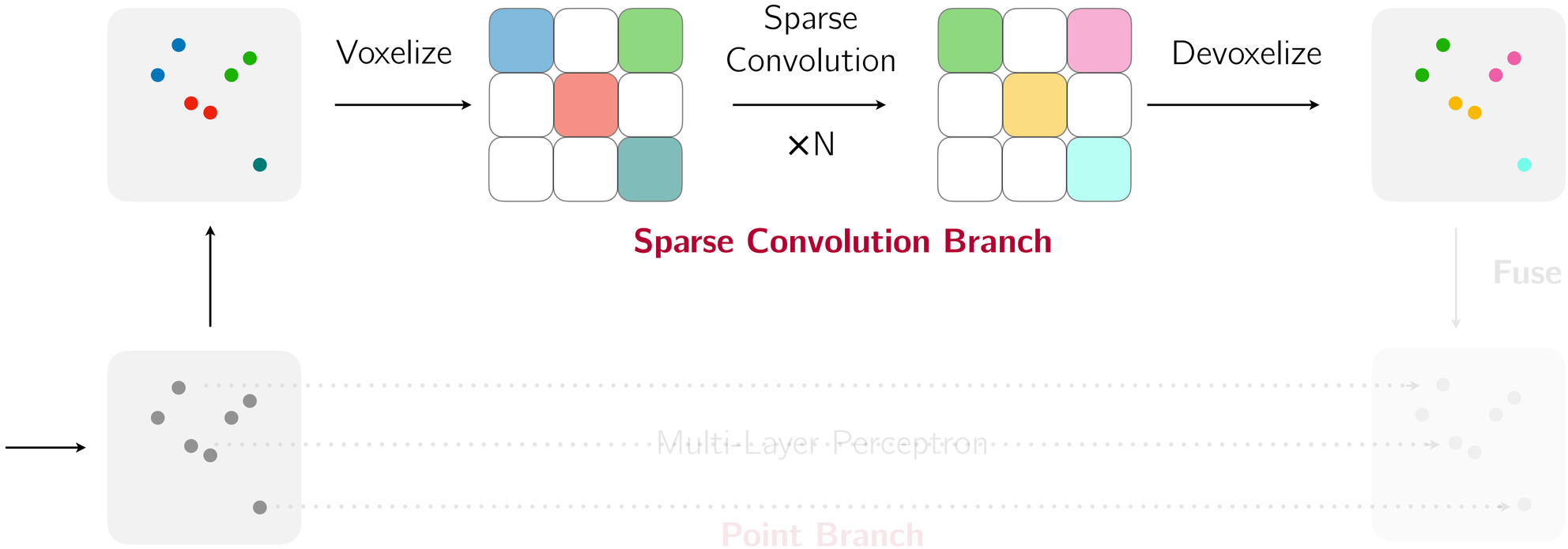
Designing Efficient 3D Modules (SPVConv)



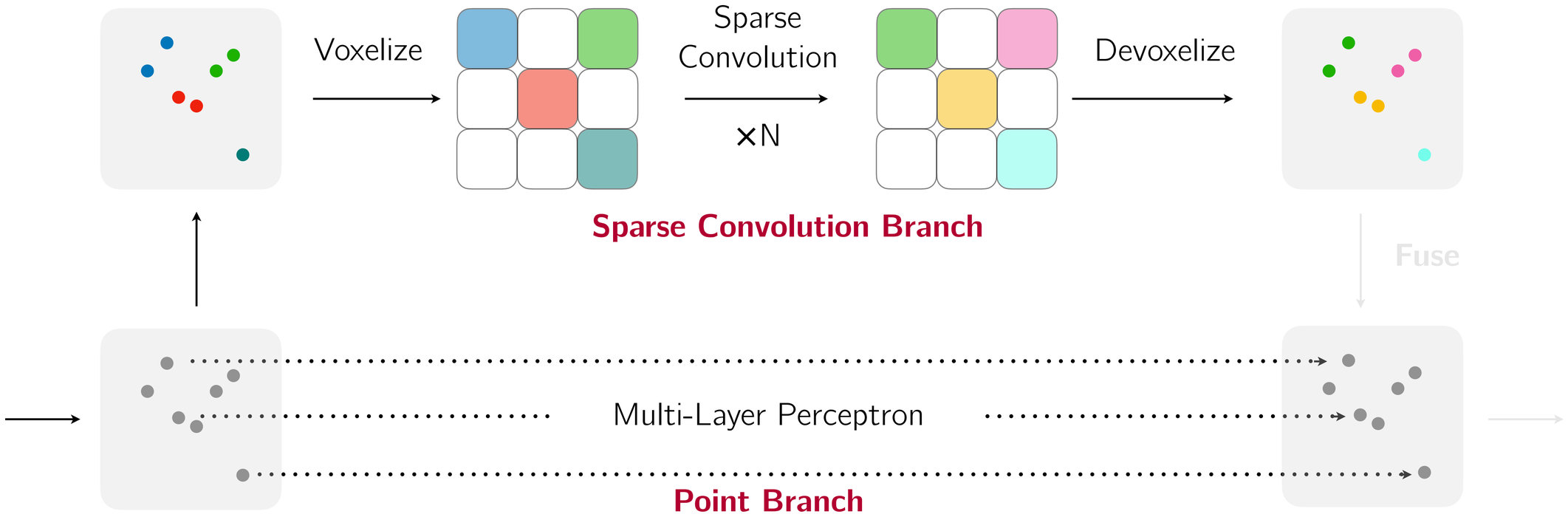
Designing Efficient 3D Modules (SPVConv)



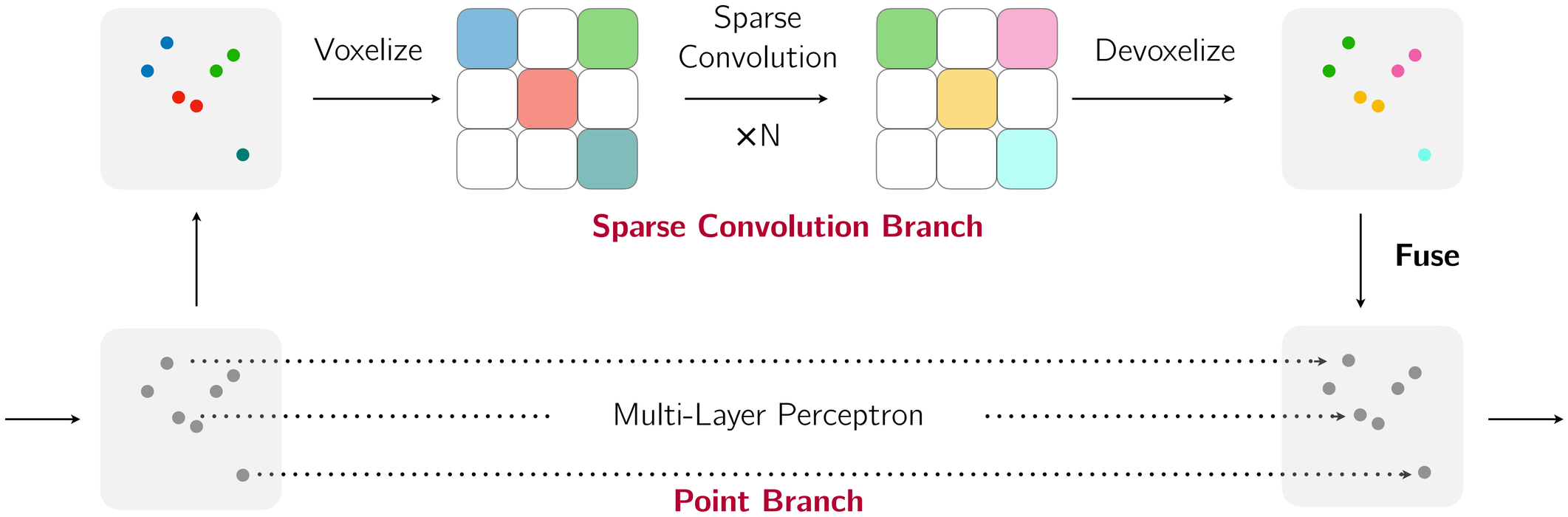
Designing Efficient 3D Modules (SPVConv)



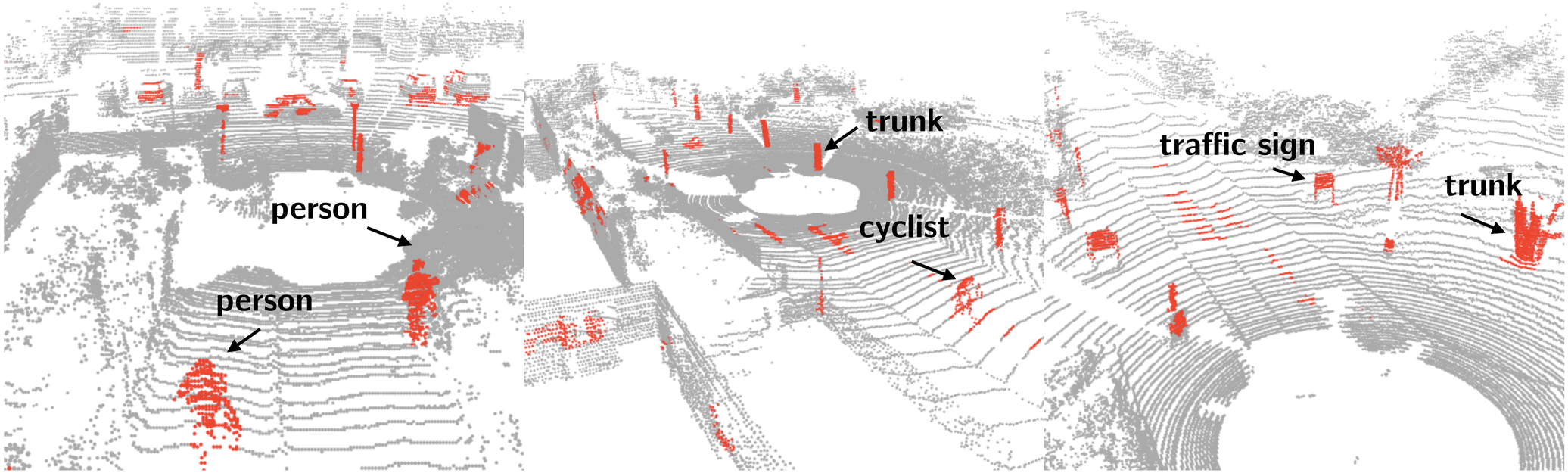
Designing Efficient 3D Modules (SPVConv)



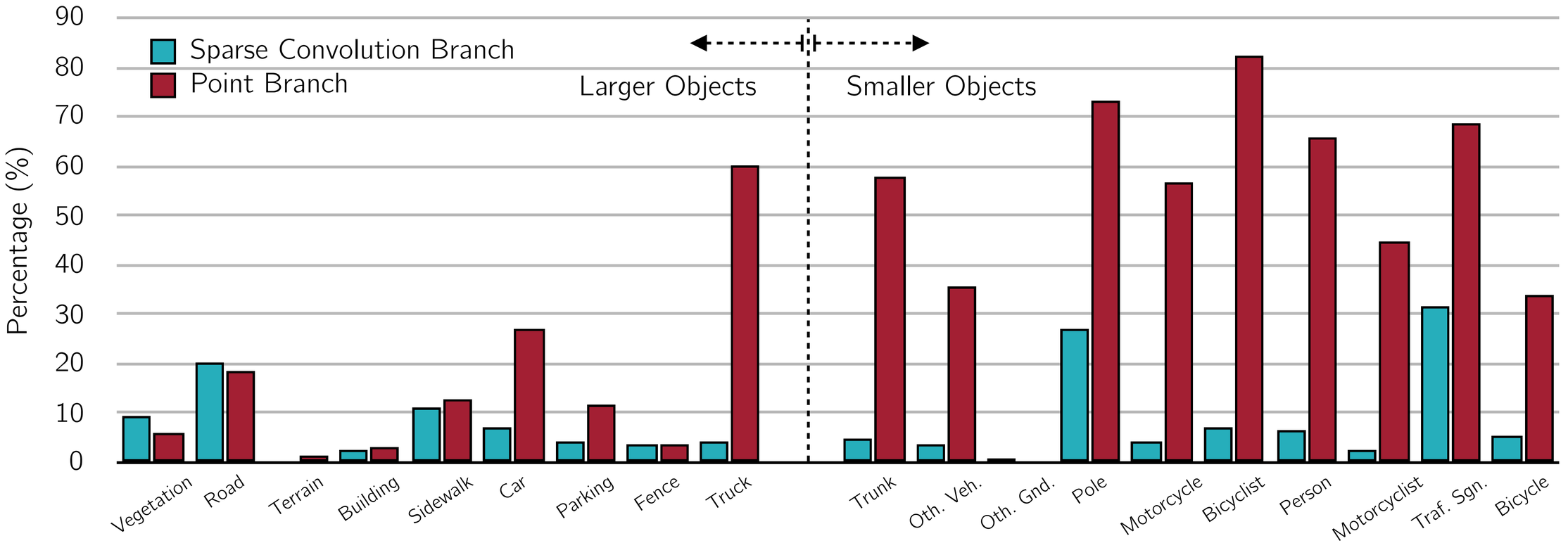
Designing Efficient 3D Modules (SPVConv)



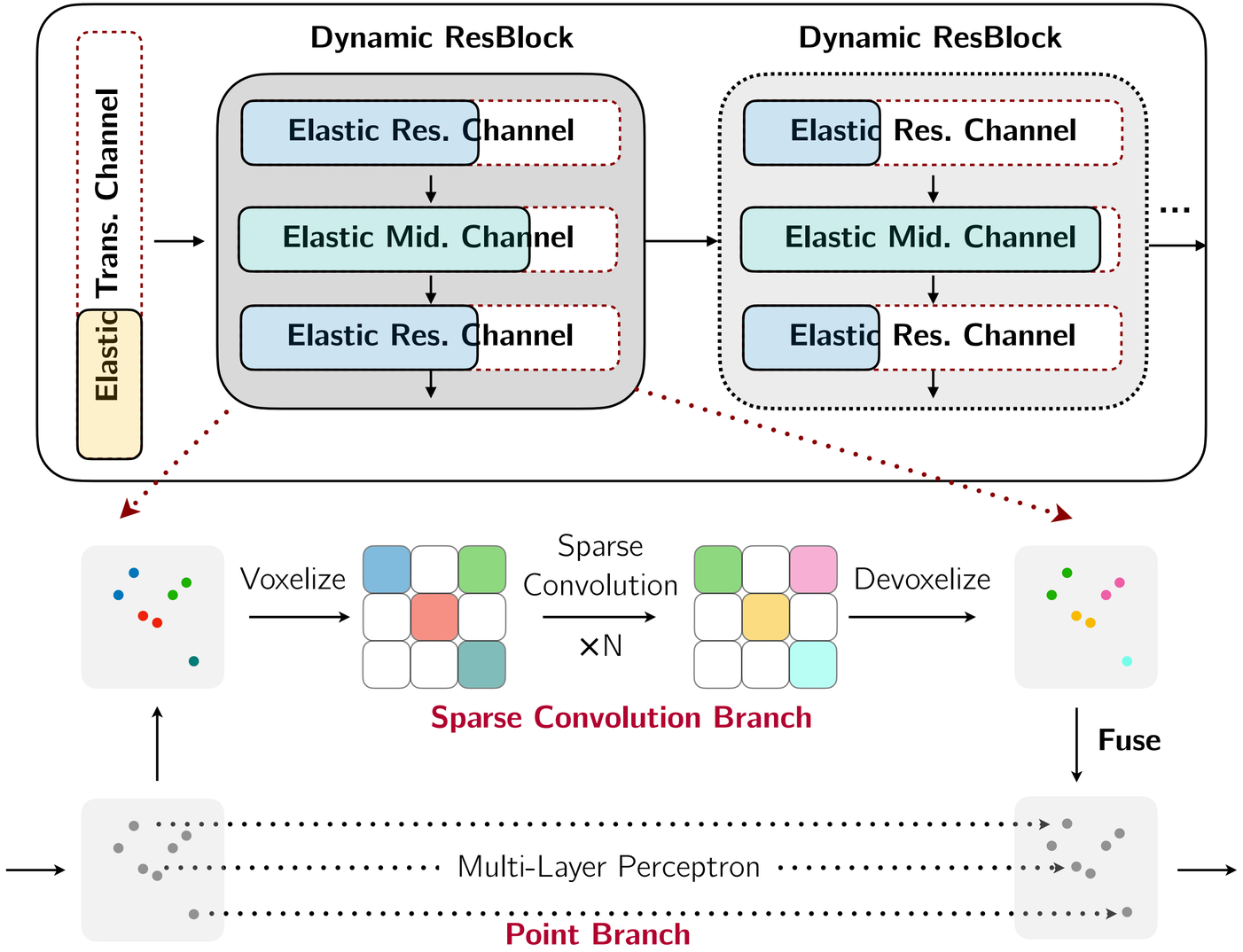
Designing Efficient 3D Modules (SPVConv)



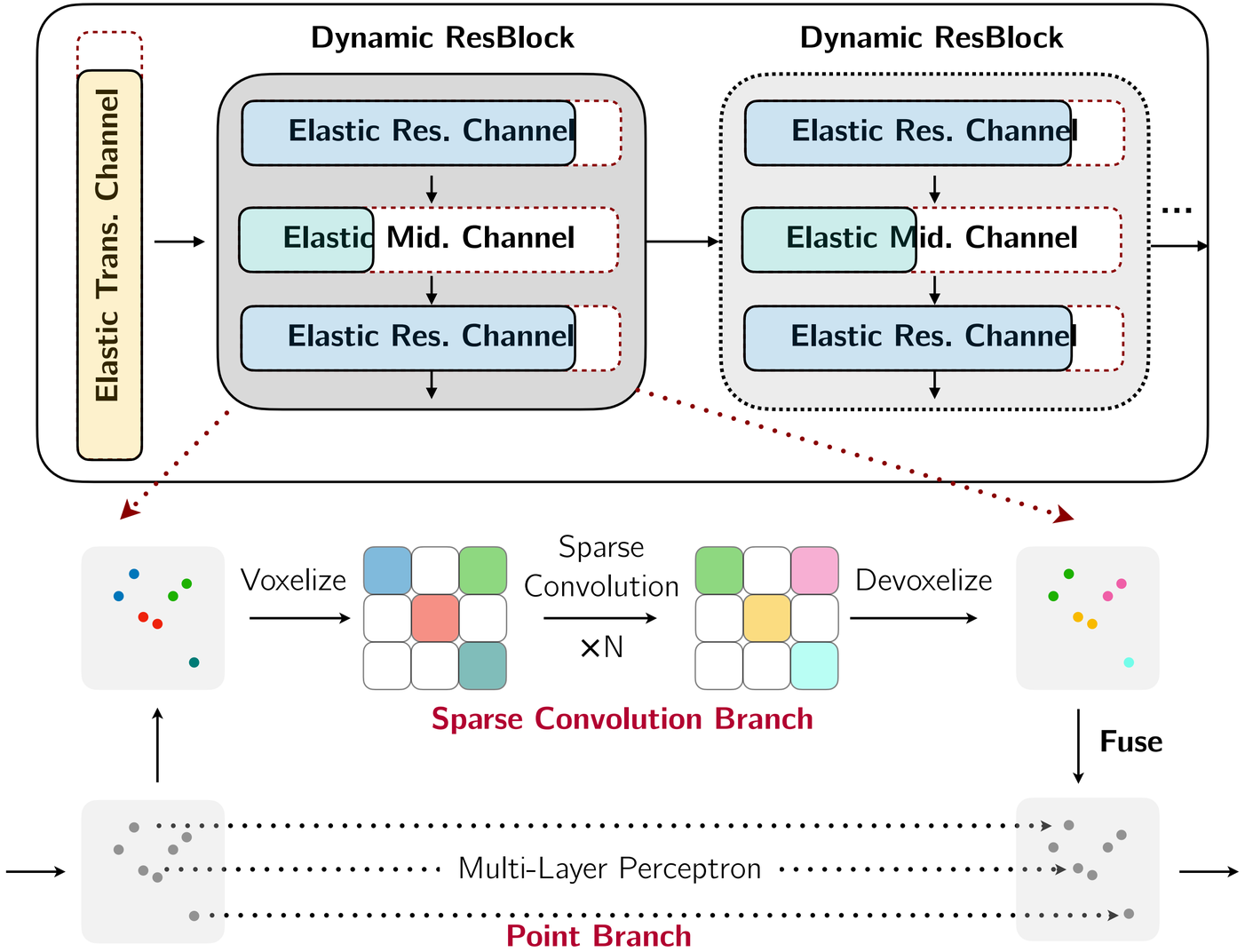
Designing Efficient 3D Modules (SPVConv)



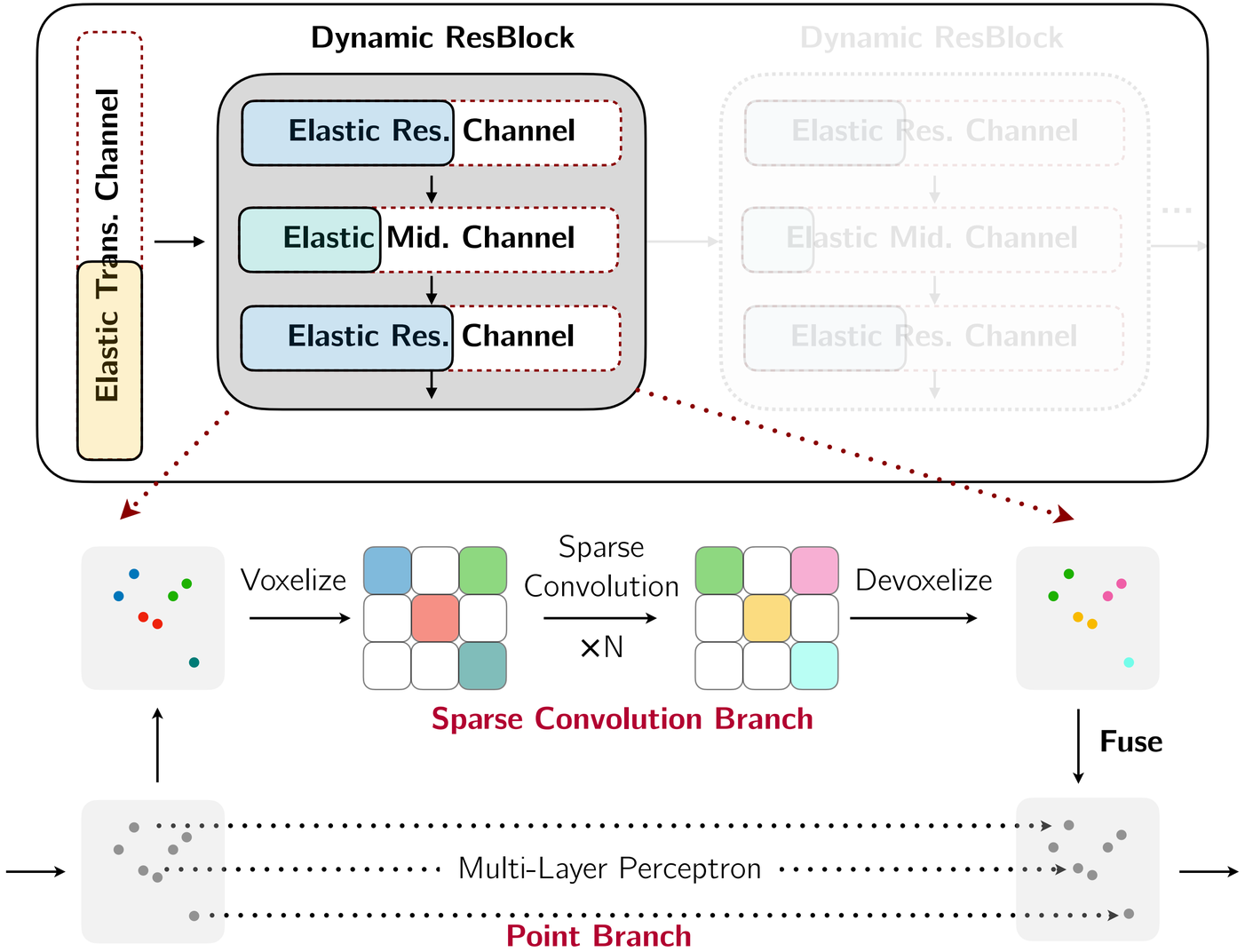
Searching Efficient 3D Architectures (3D-NAS)



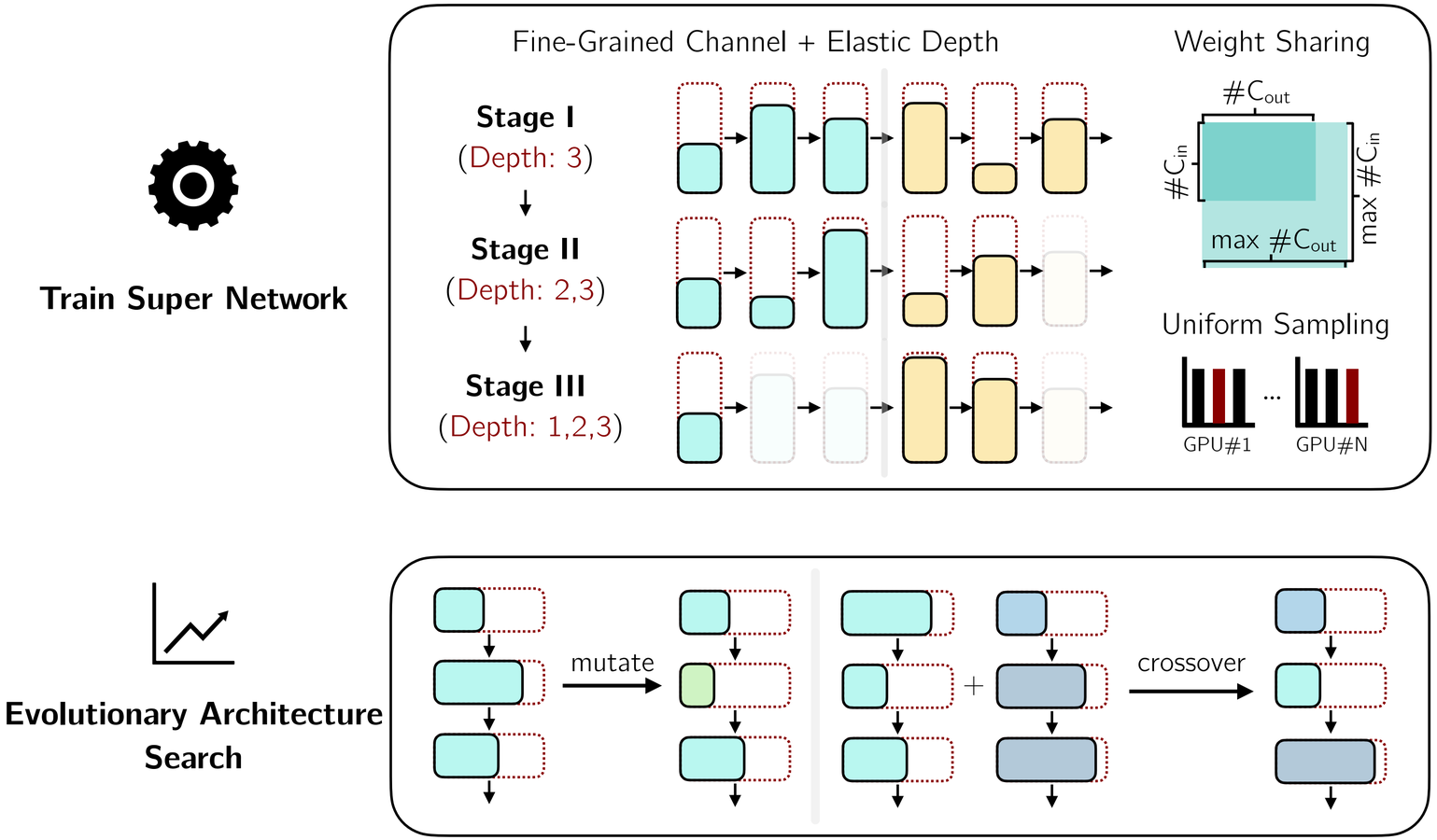
Searching Efficient 3D Architectures (3D-NAS)



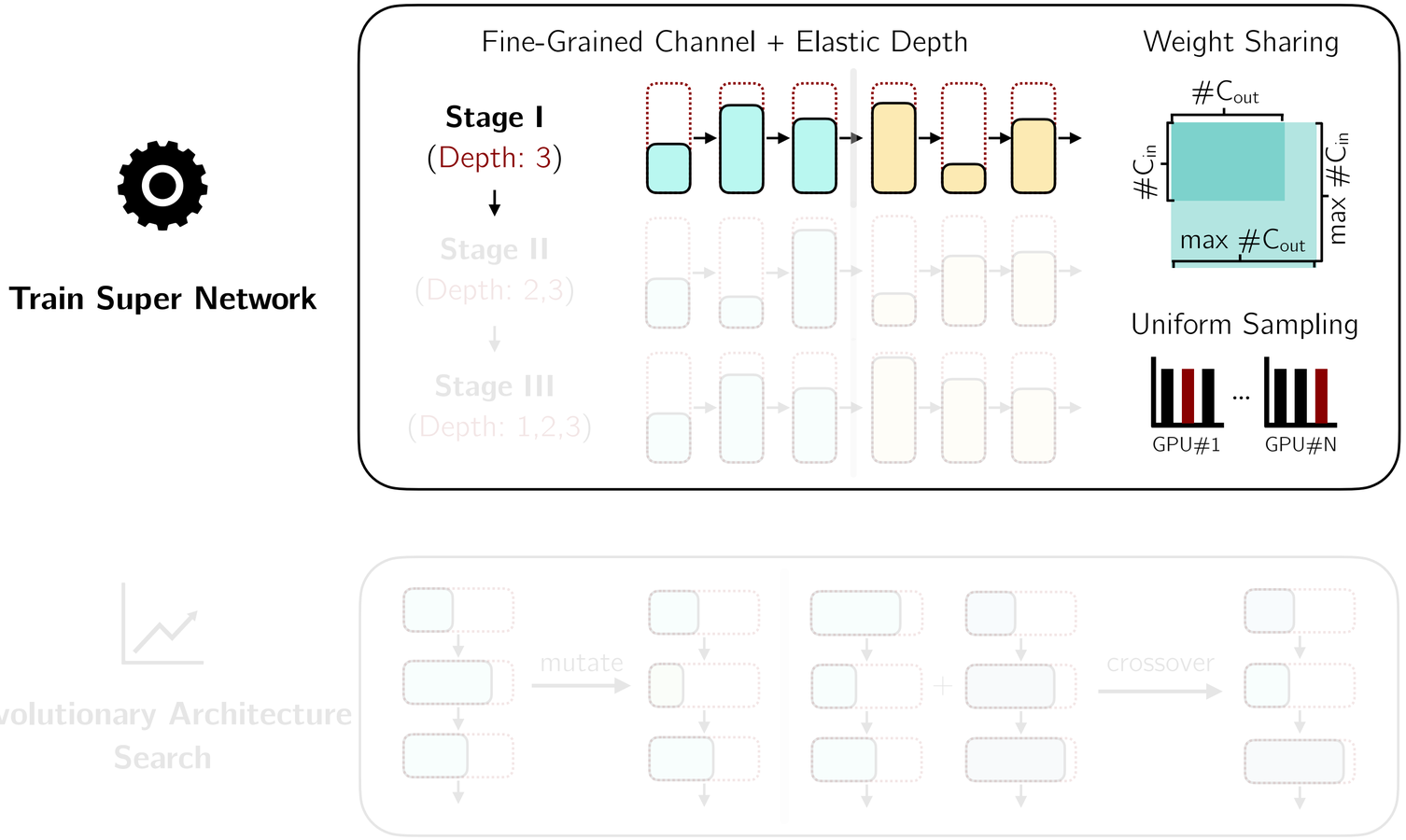
Searching Efficient 3D Architectures (3D-NAS)



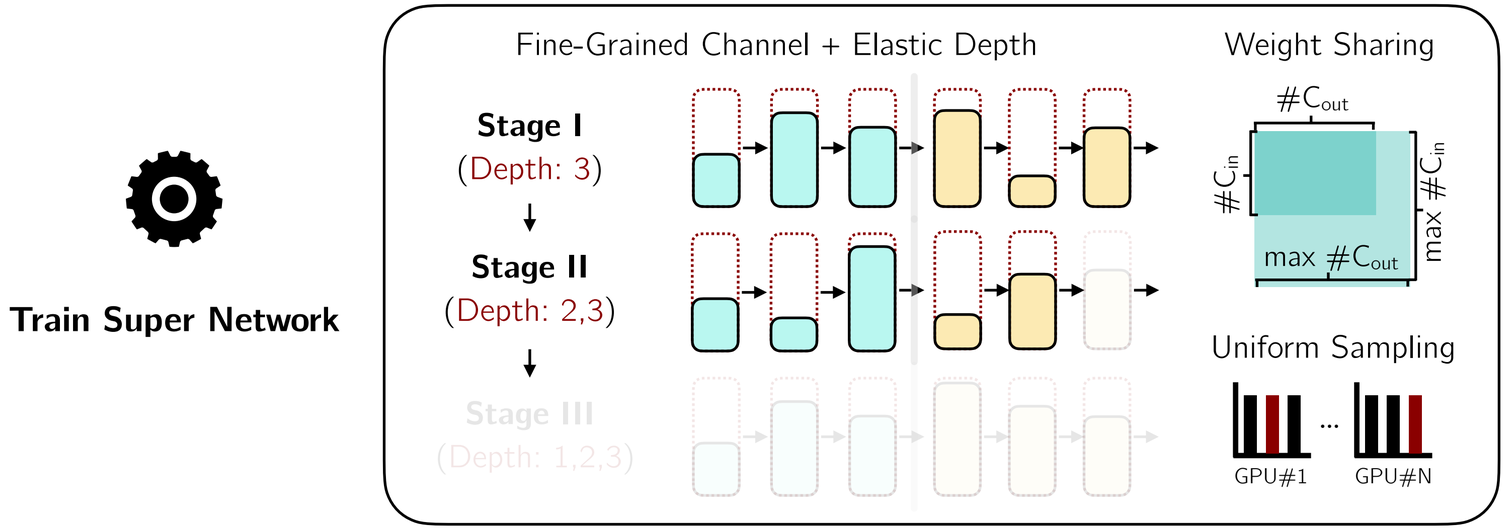
Searching Efficient 3D Architectures (3D-NAS)



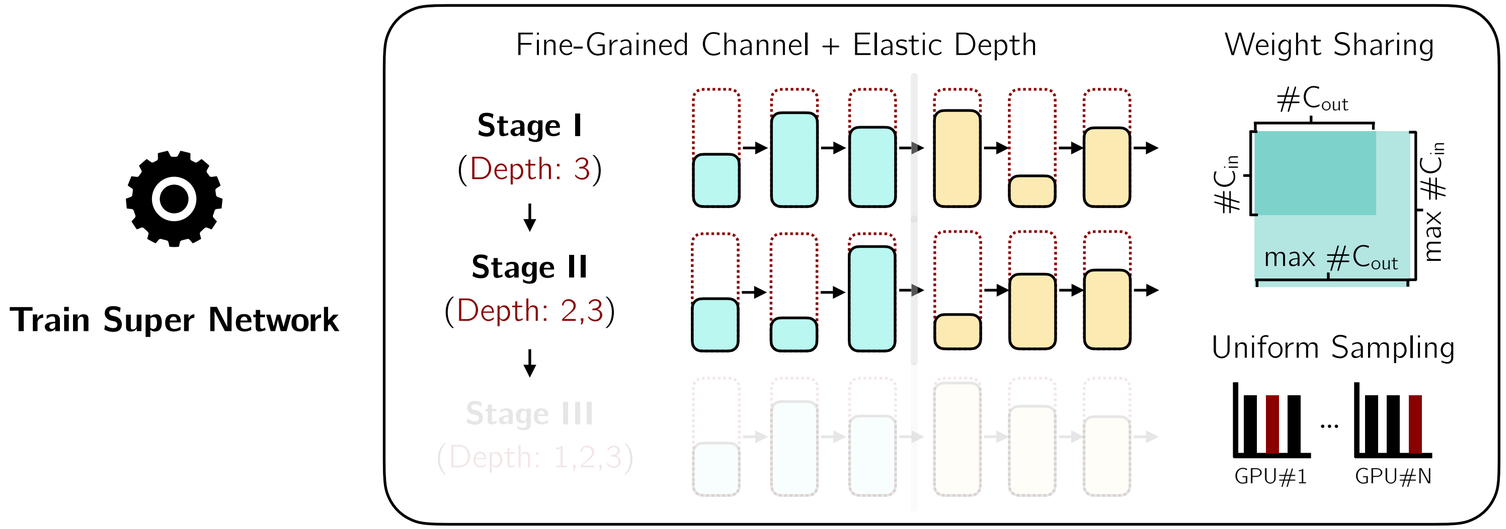
Searching Efficient 3D Architectures (3D-NAS)



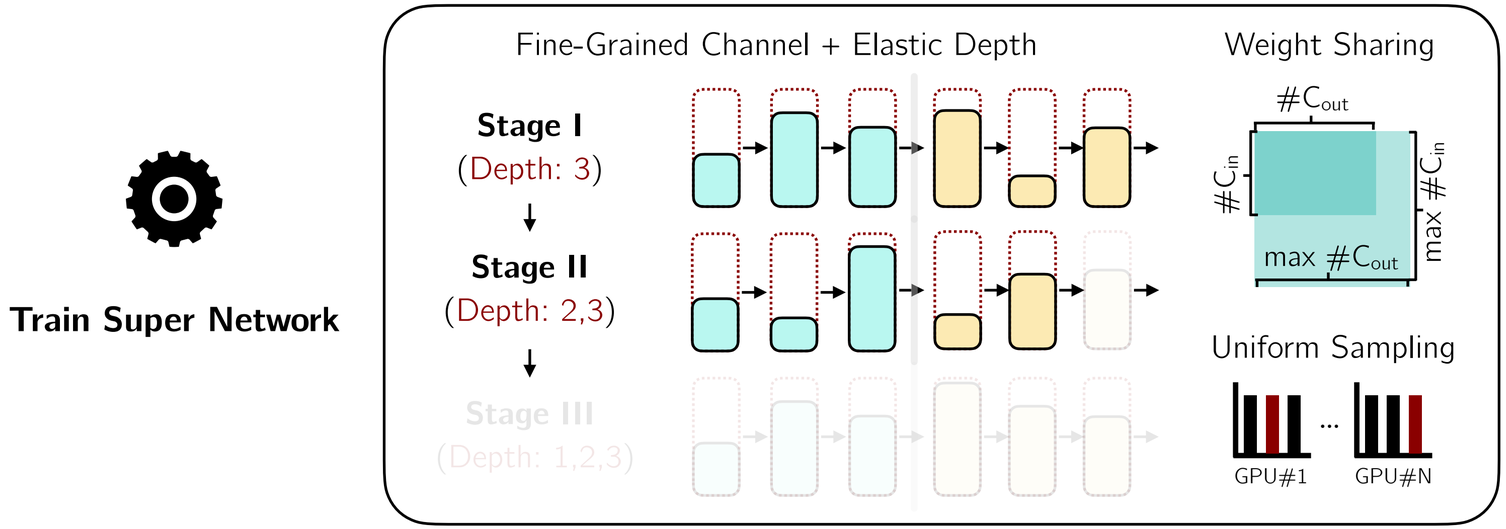
Searching Efficient 3D Architectures (3D-NAS)



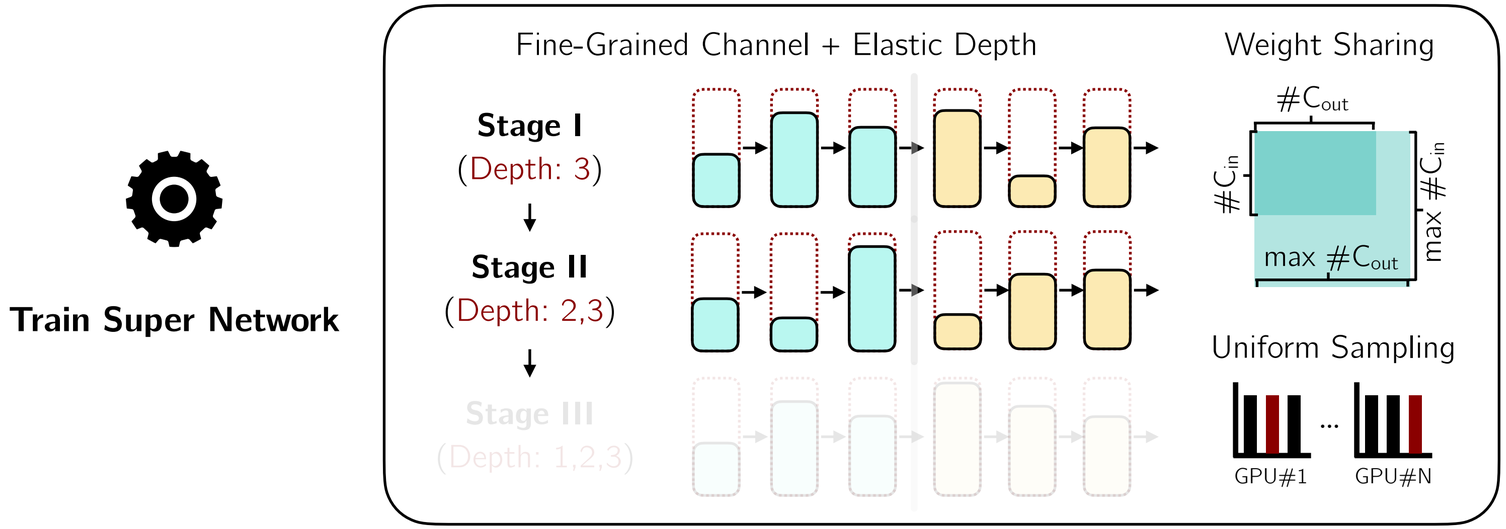
Searching Efficient 3D Architectures (3D-NAS)



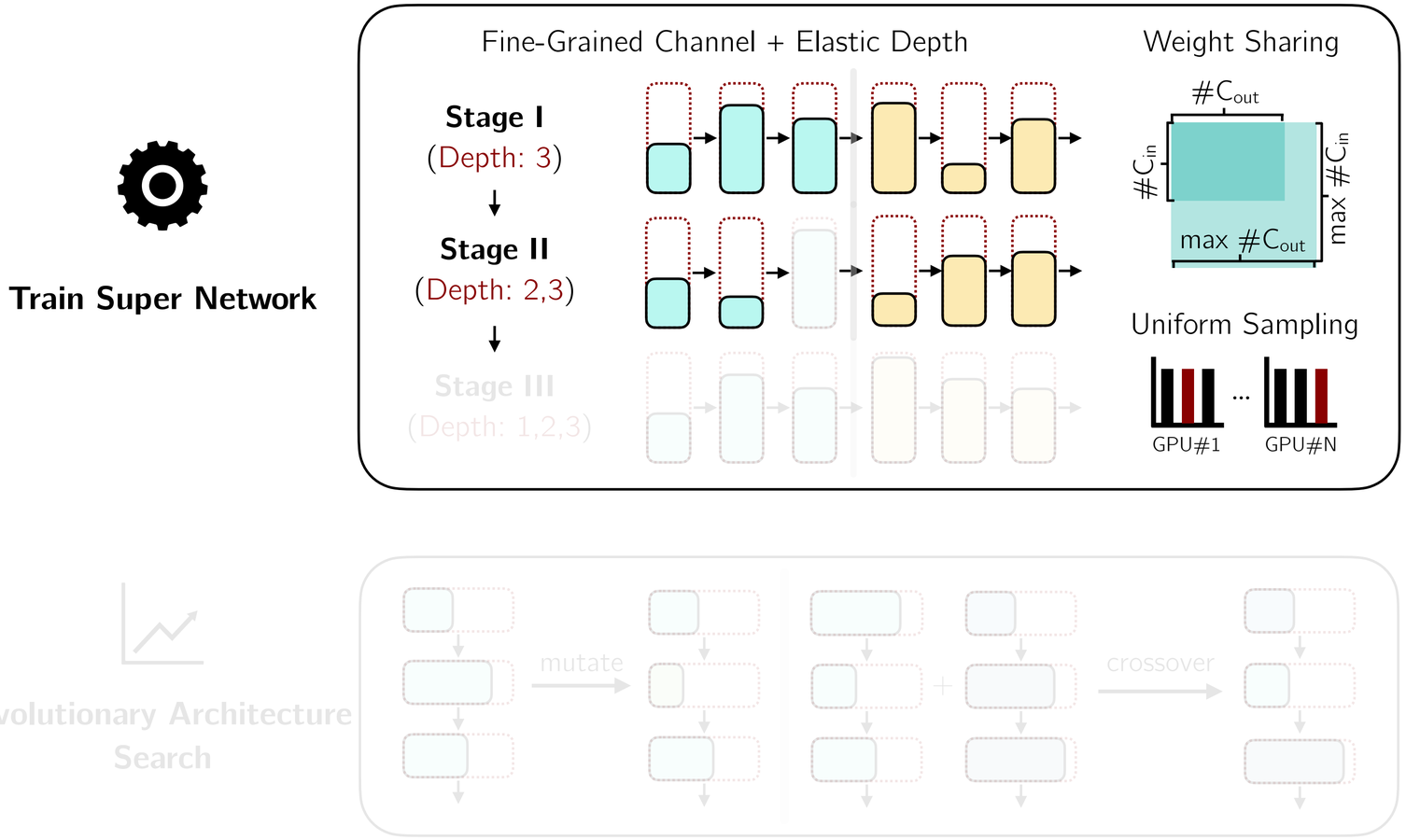
Searching Efficient 3D Architectures (3D-NAS)



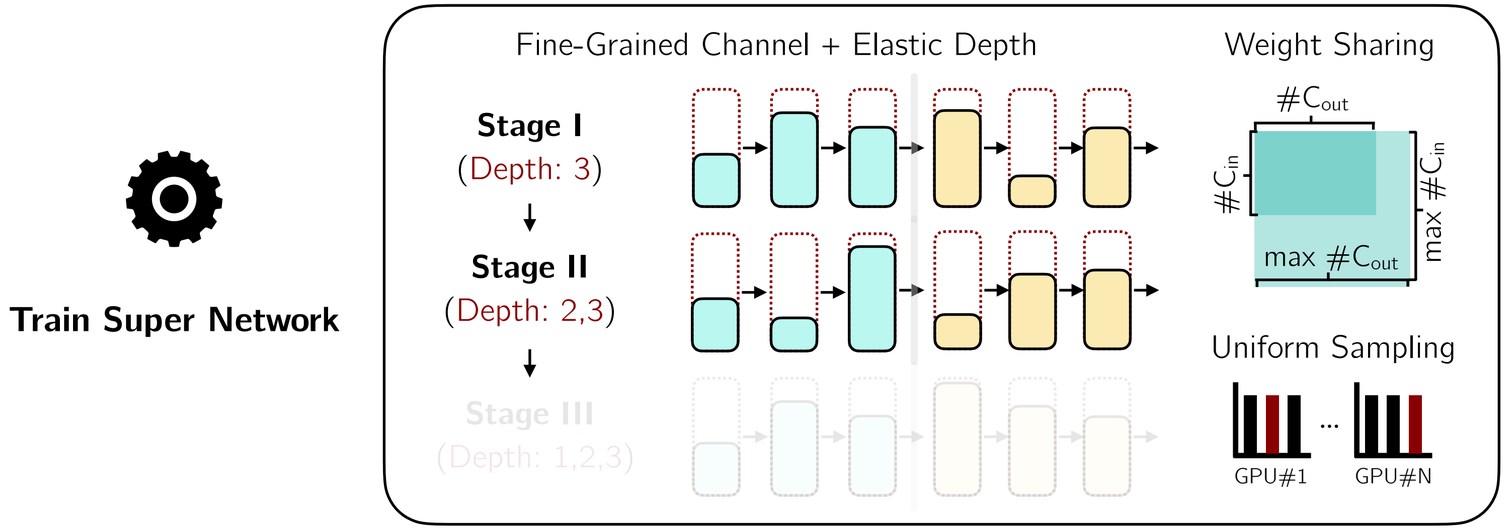
Searching Efficient 3D Architectures (3D-NAS)



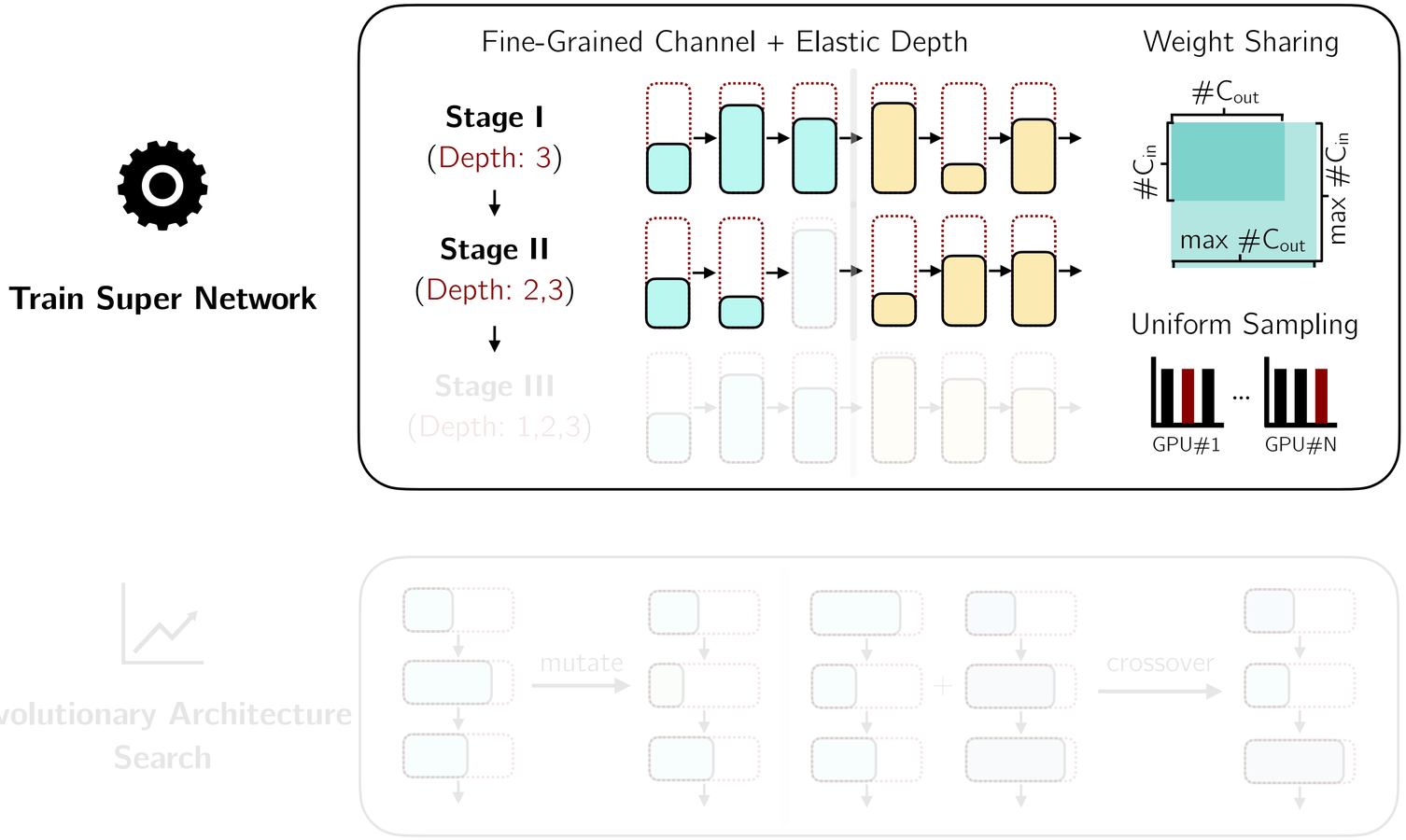
Searching Efficient 3D Architectures (3D-NAS)



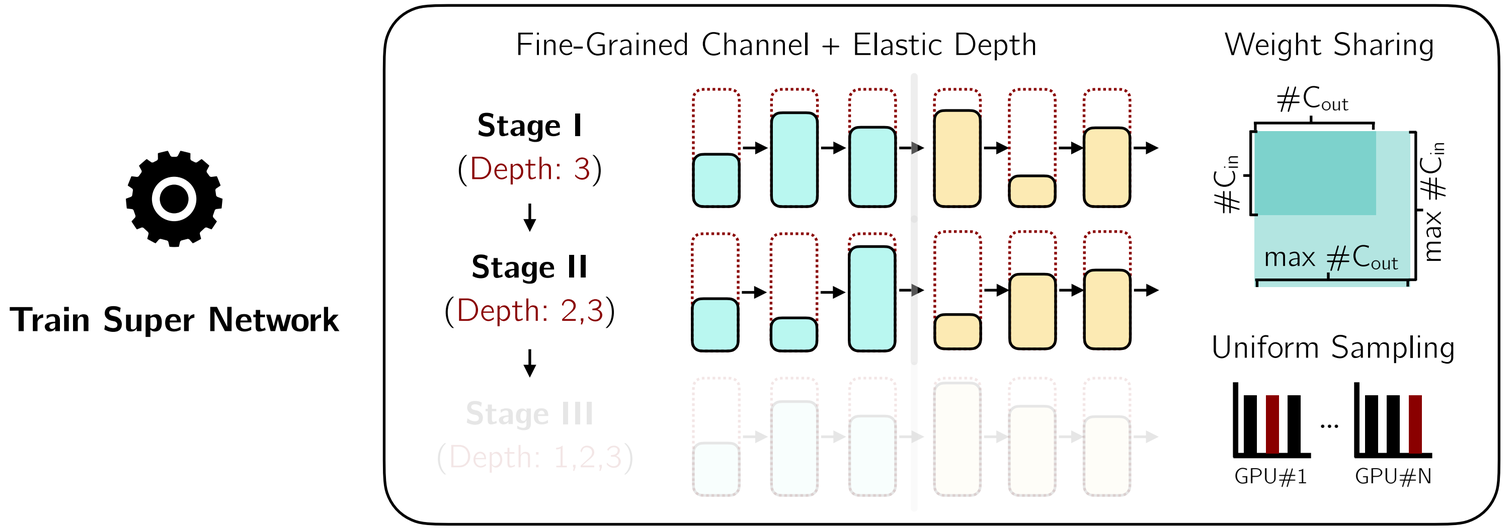
Searching Efficient 3D Architectures (3D-NAS)



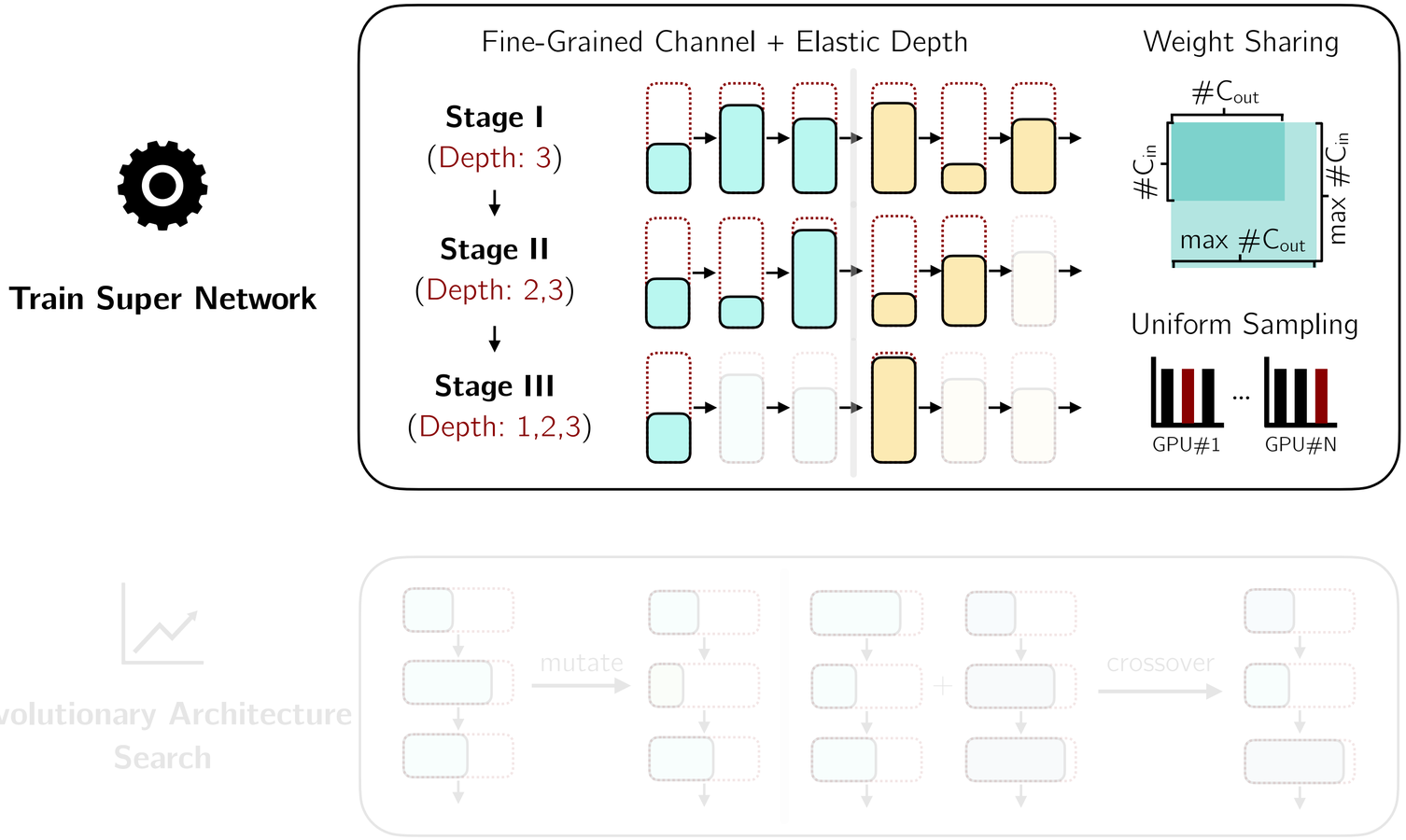
Searching Efficient 3D Architectures (3D-NAS)



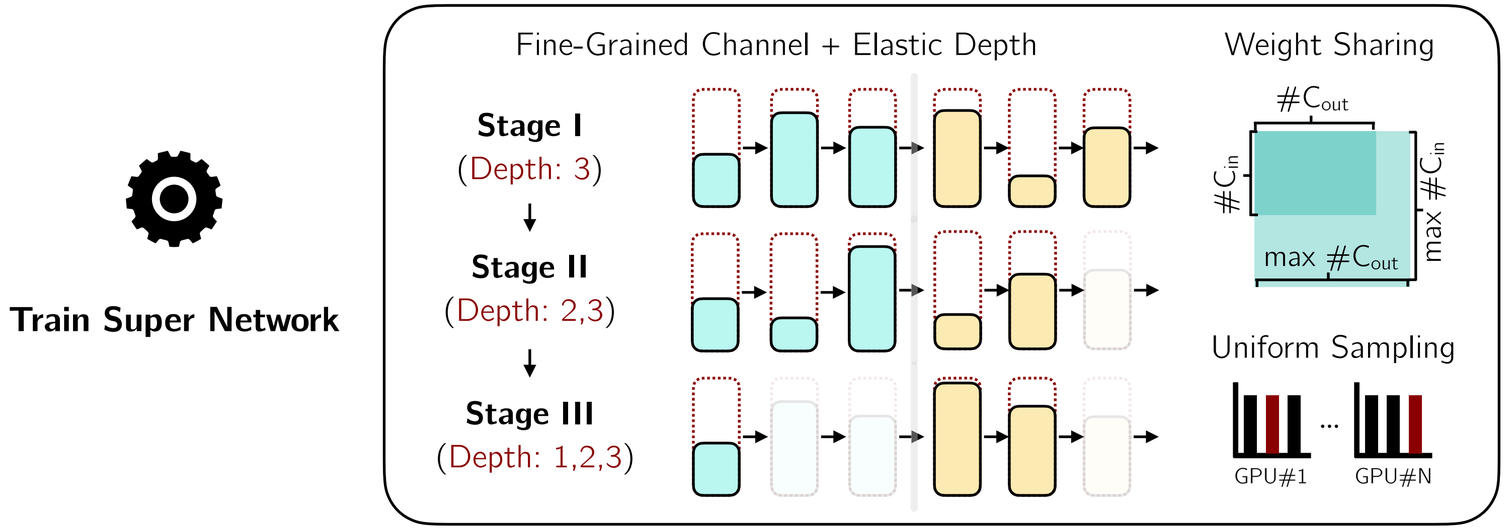
Searching Efficient 3D Architectures (3D-NAS)



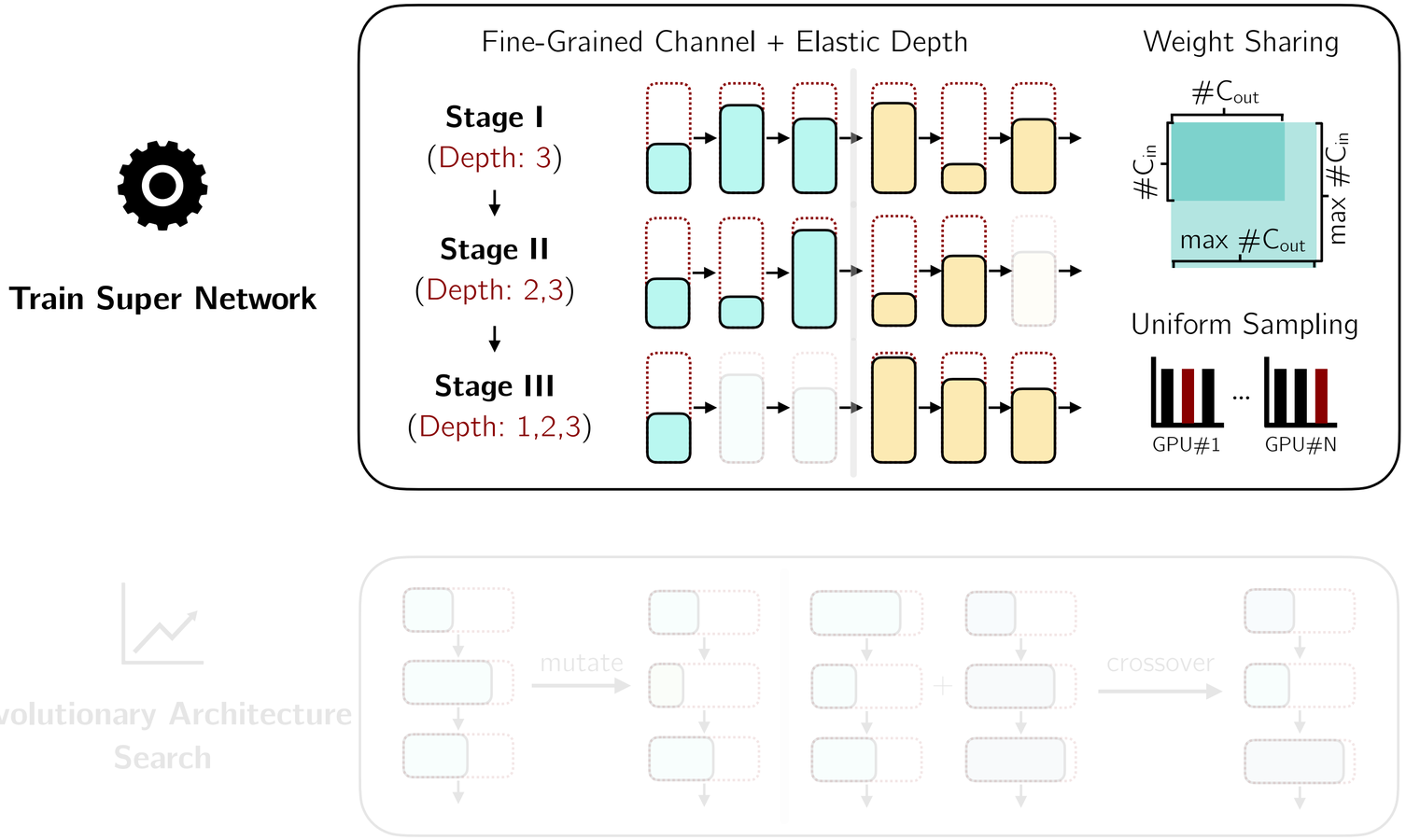
Searching Efficient 3D Architectures (3D-NAS)



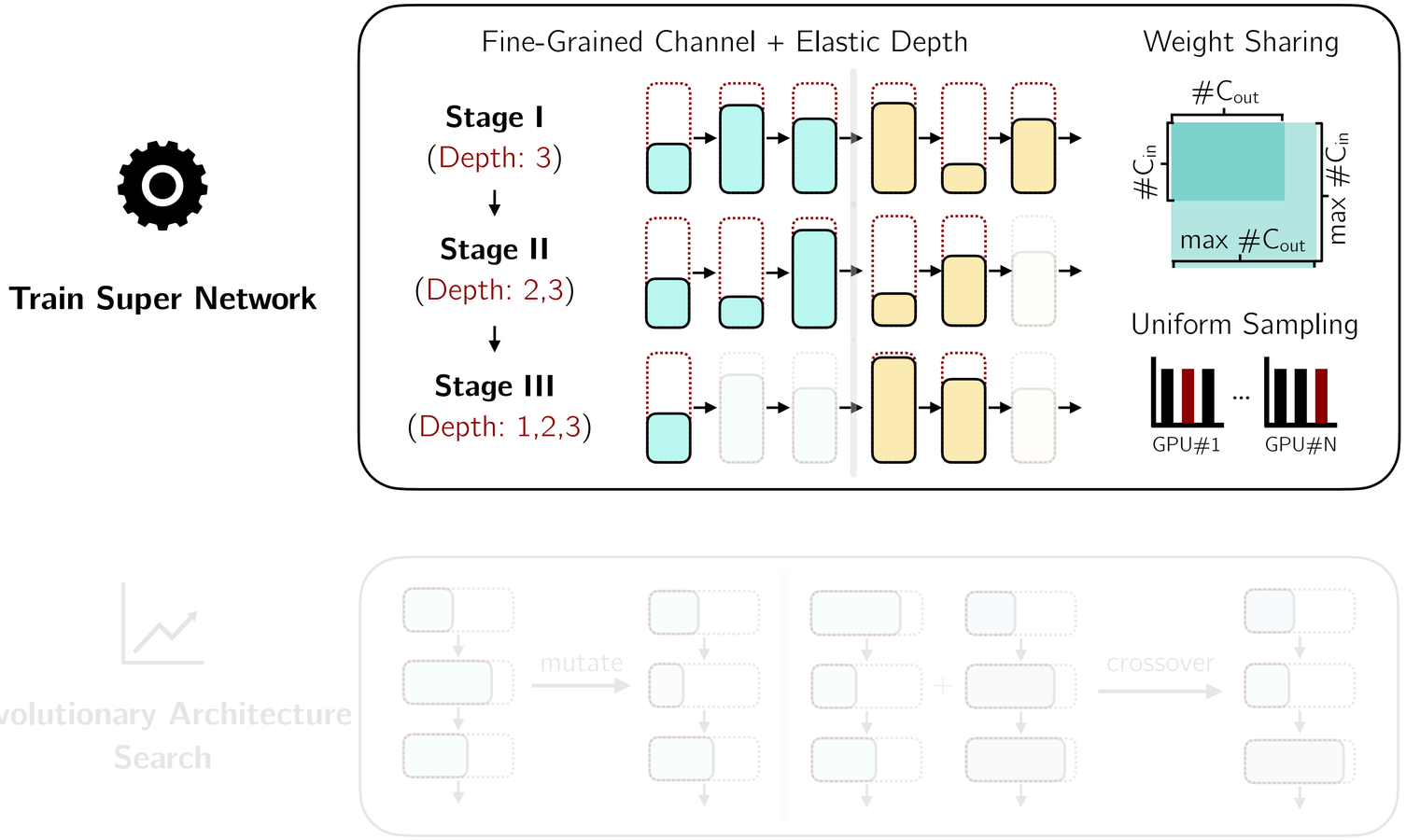
Searching Efficient 3D Architectures (3D-NAS)



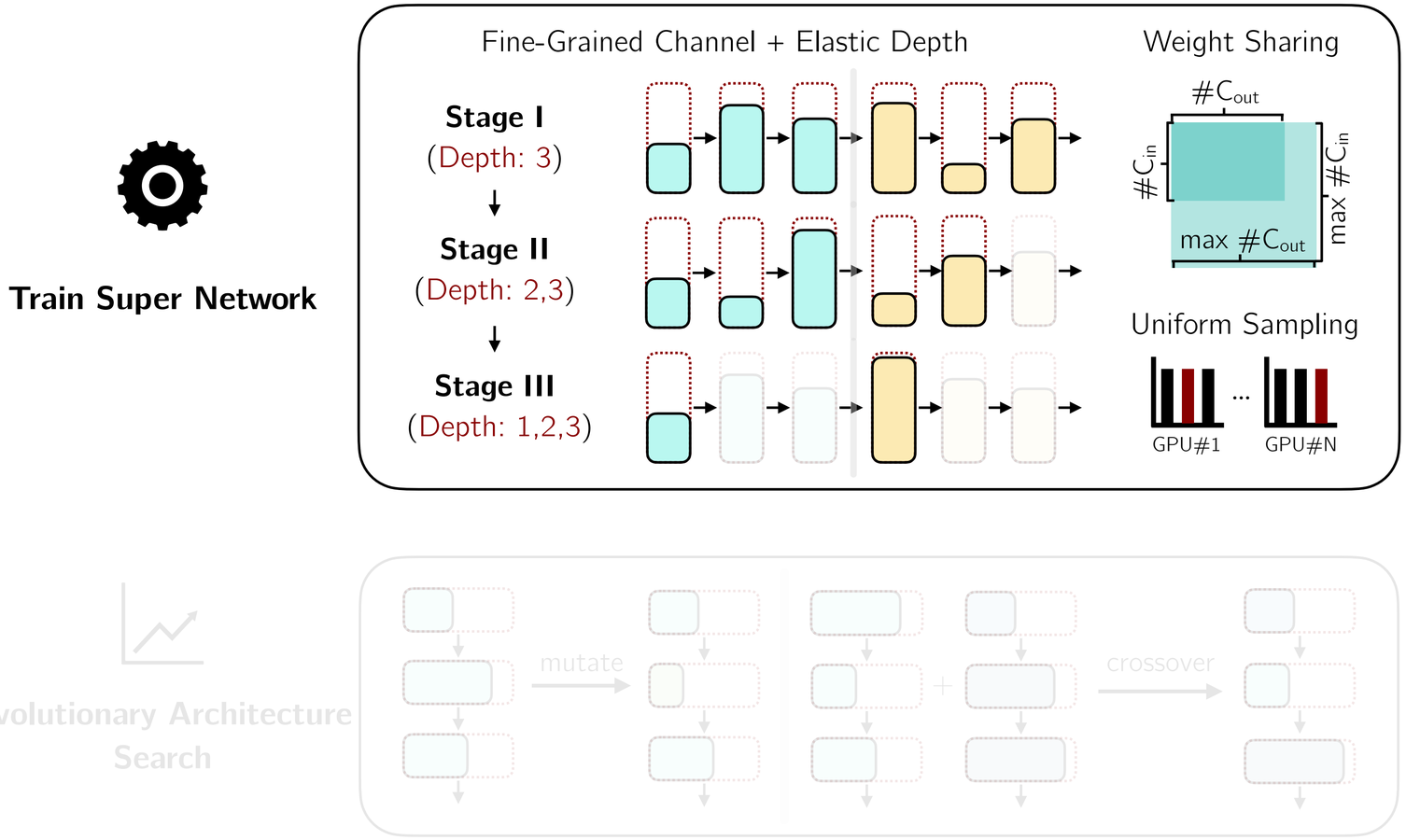
Searching Efficient 3D Architectures (3D-NAS)



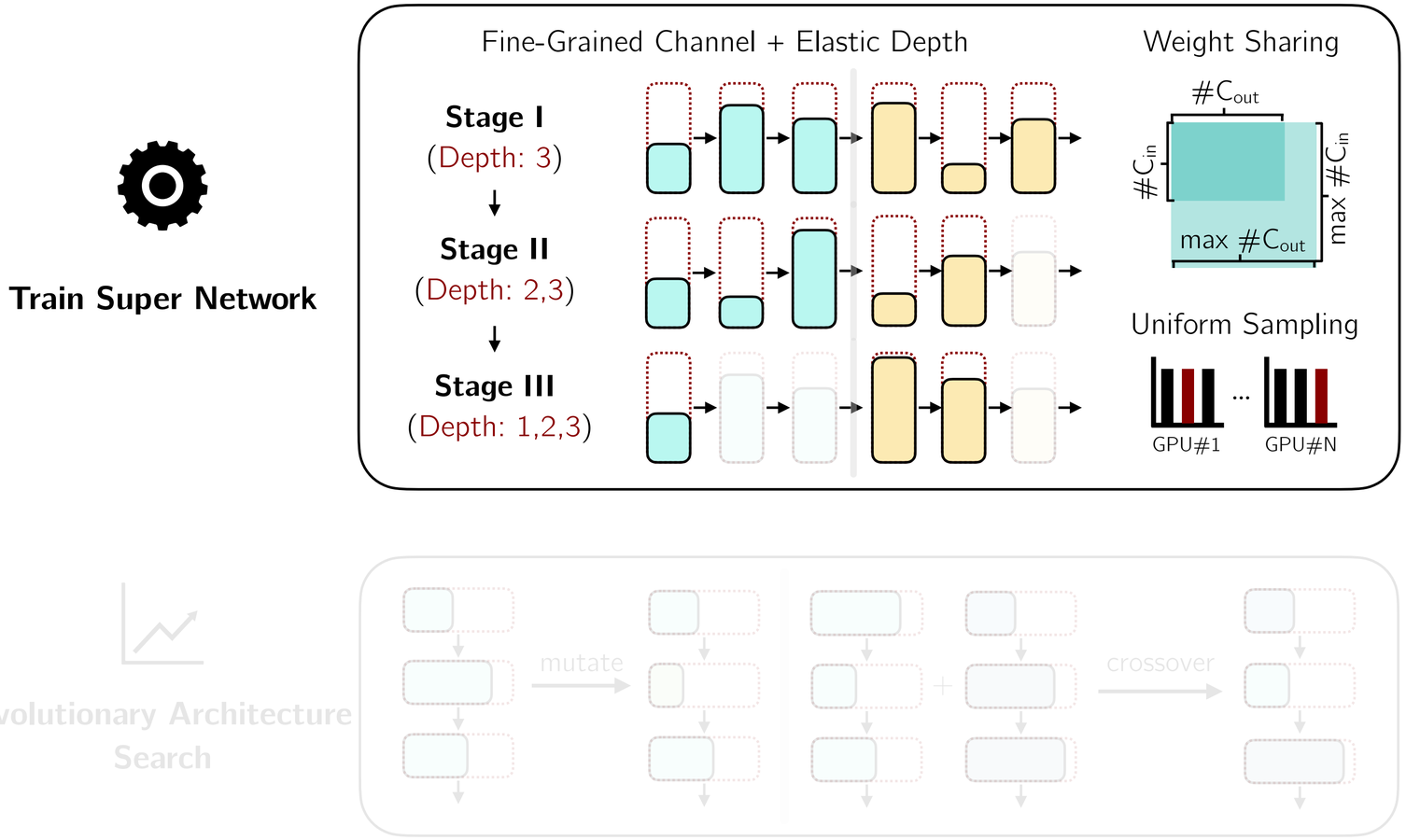
Searching Efficient 3D Architectures (3D-NAS)



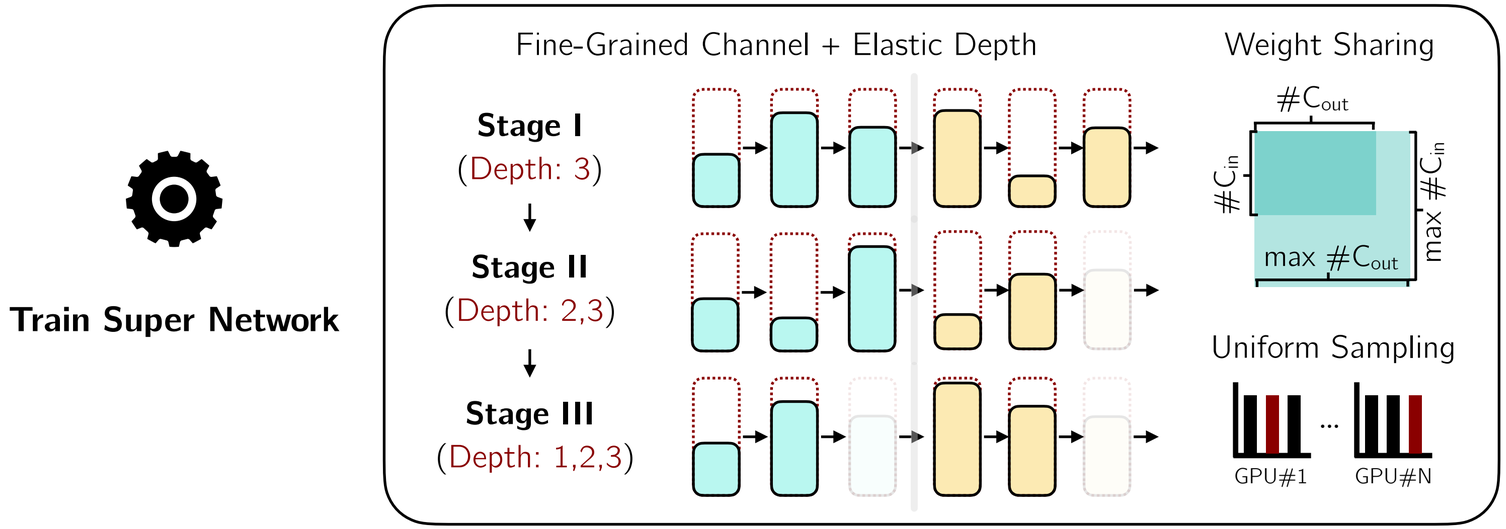
Searching Efficient 3D Architectures (3D-NAS)



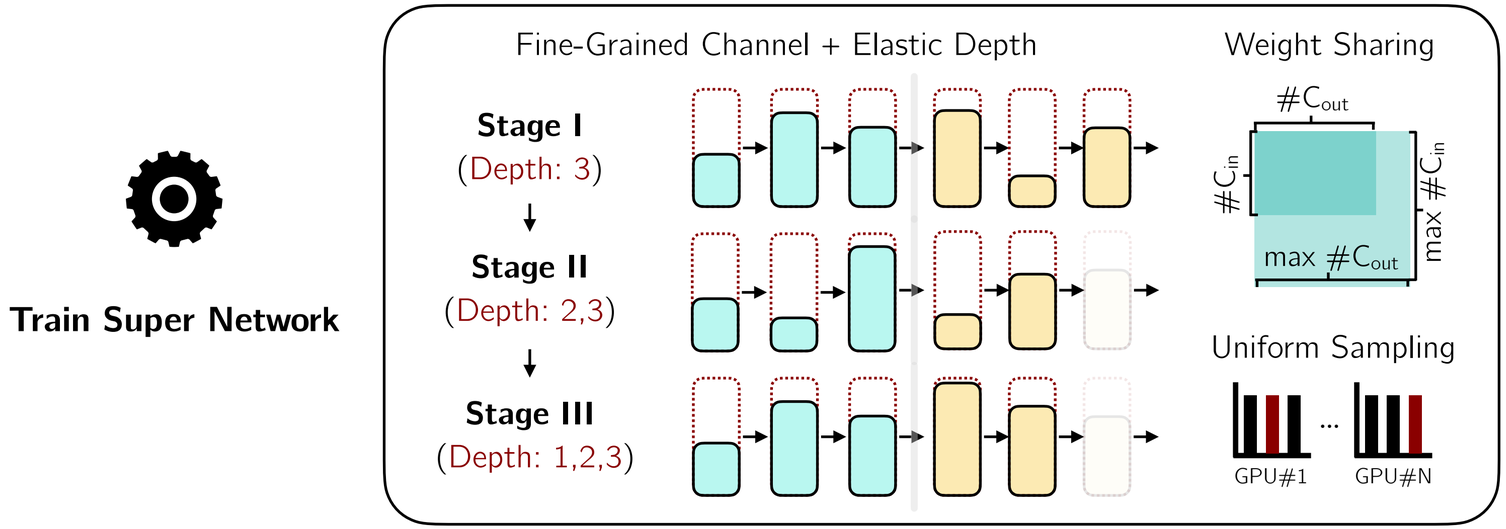
Searching Efficient 3D Architectures (3D-NAS)



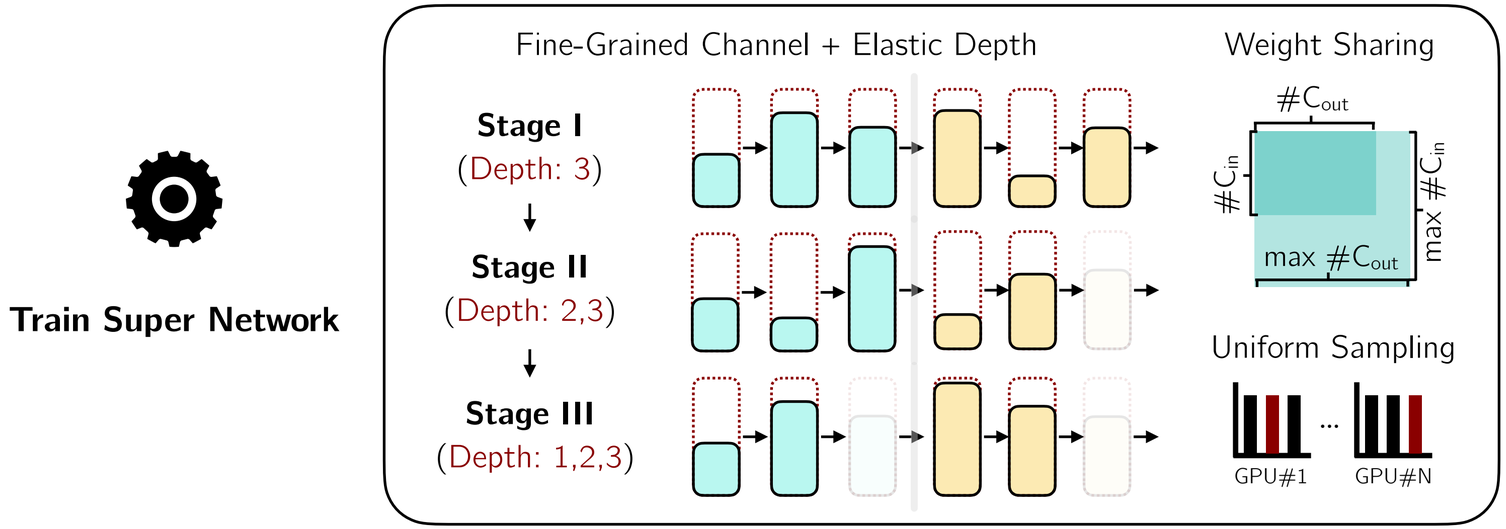
Searching Efficient 3D Architectures (3D-NAS)



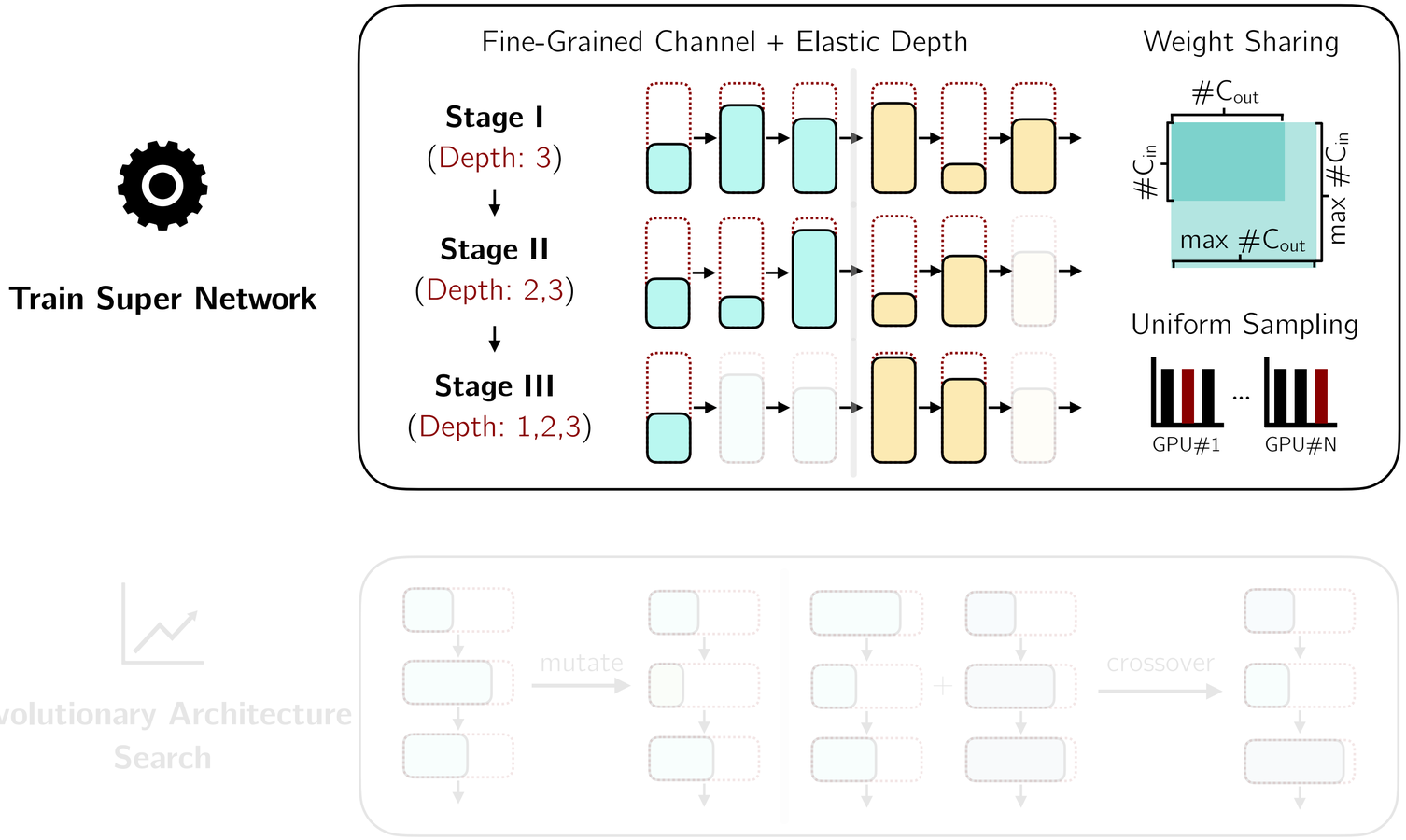
Searching Efficient 3D Architectures (3D-NAS)



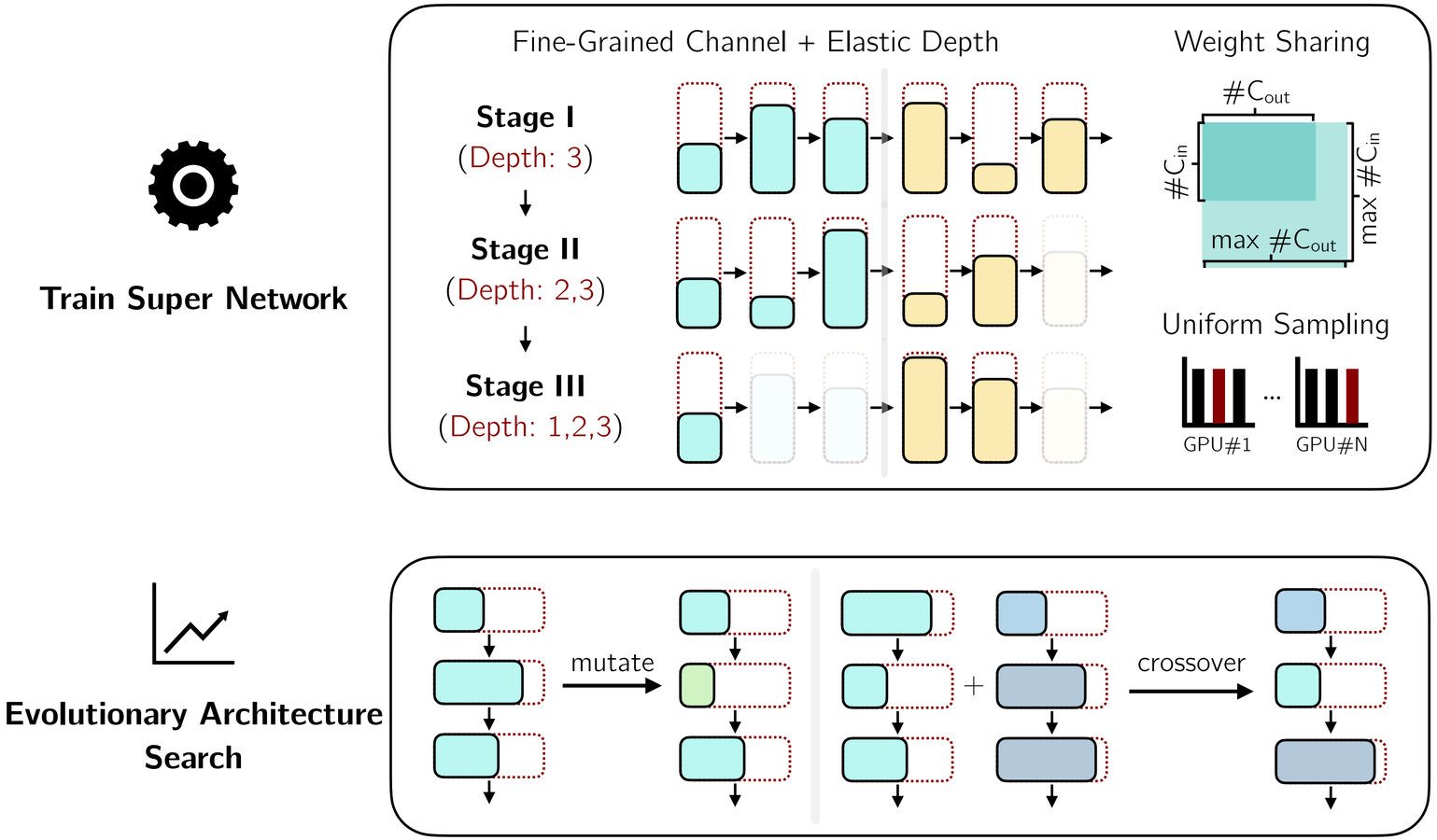
Searching Efficient 3D Architectures (3D-NAS)



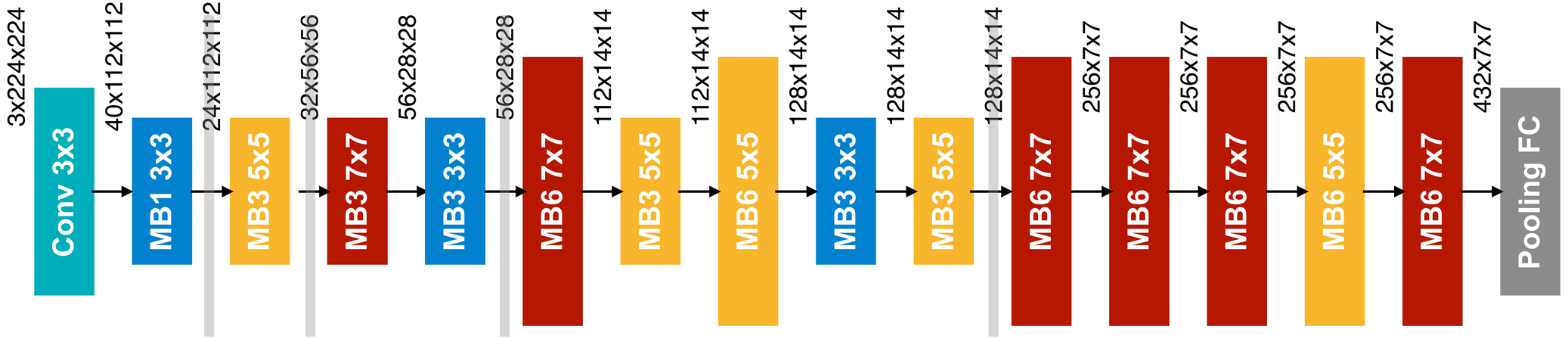
Searching Efficient 3D Architectures (3D-NAS)



Searching Efficient 3D Architectures (3D-NAS)

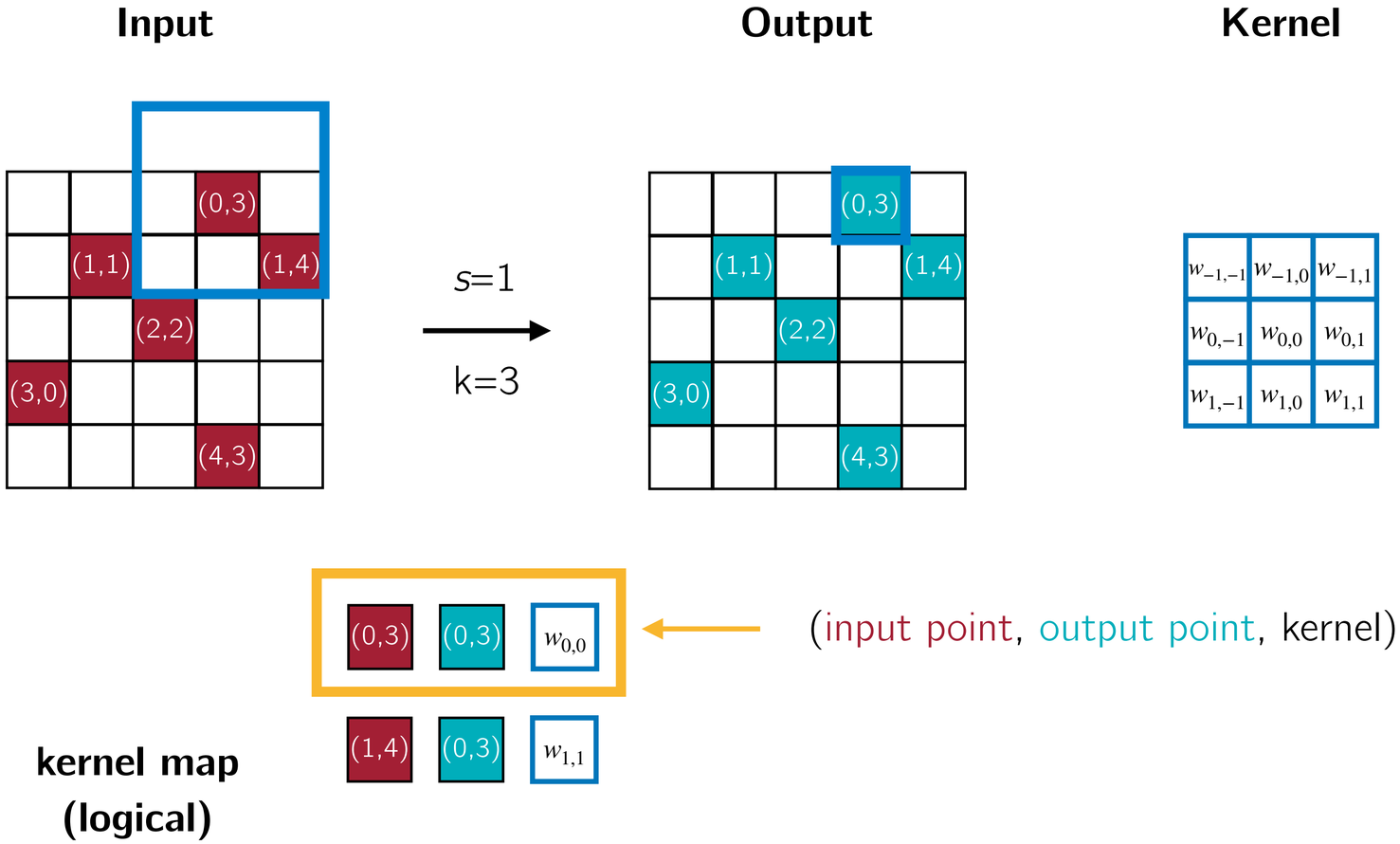


Details: Small Kernel Matters

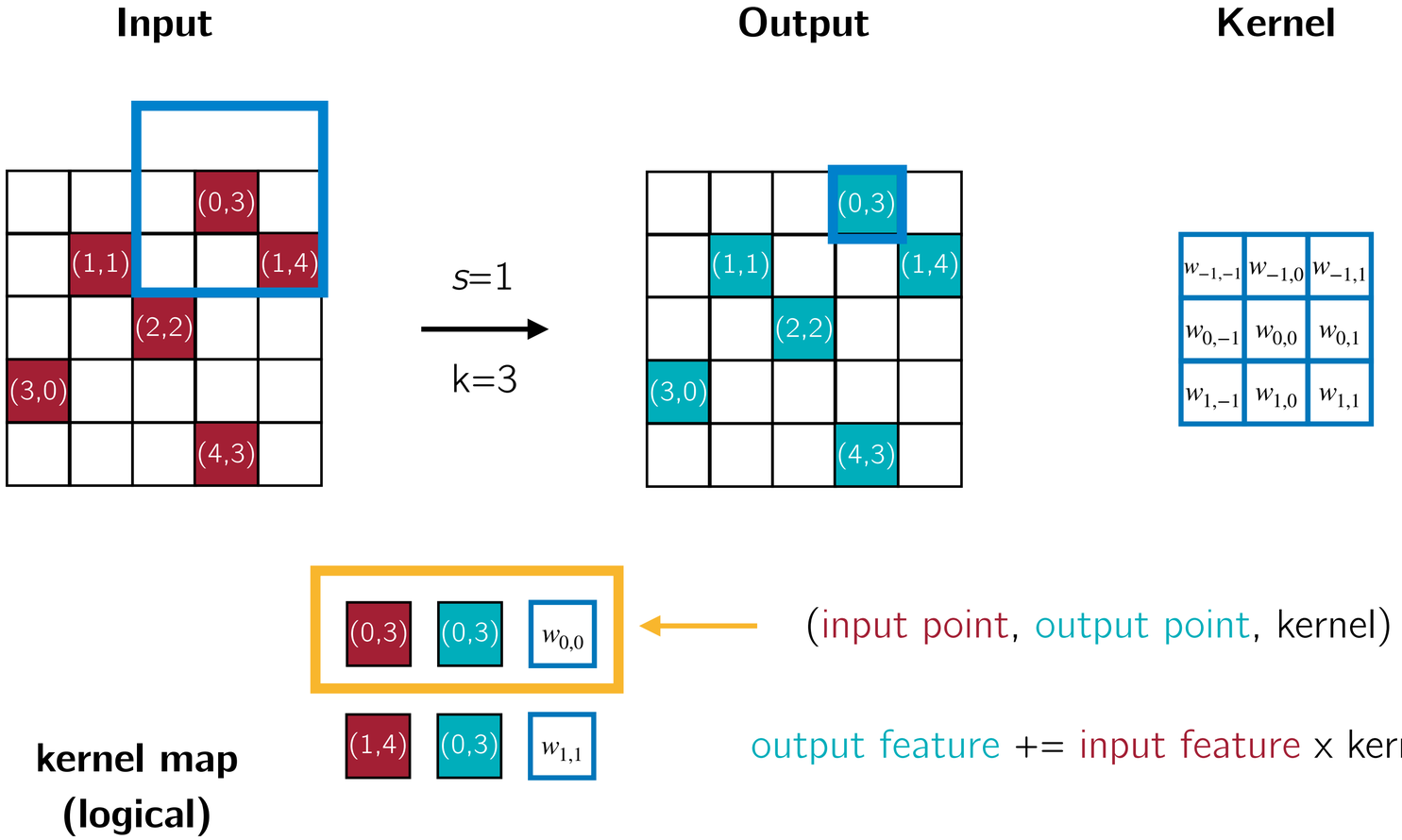


Large kernels are efficient in 2D-NAS

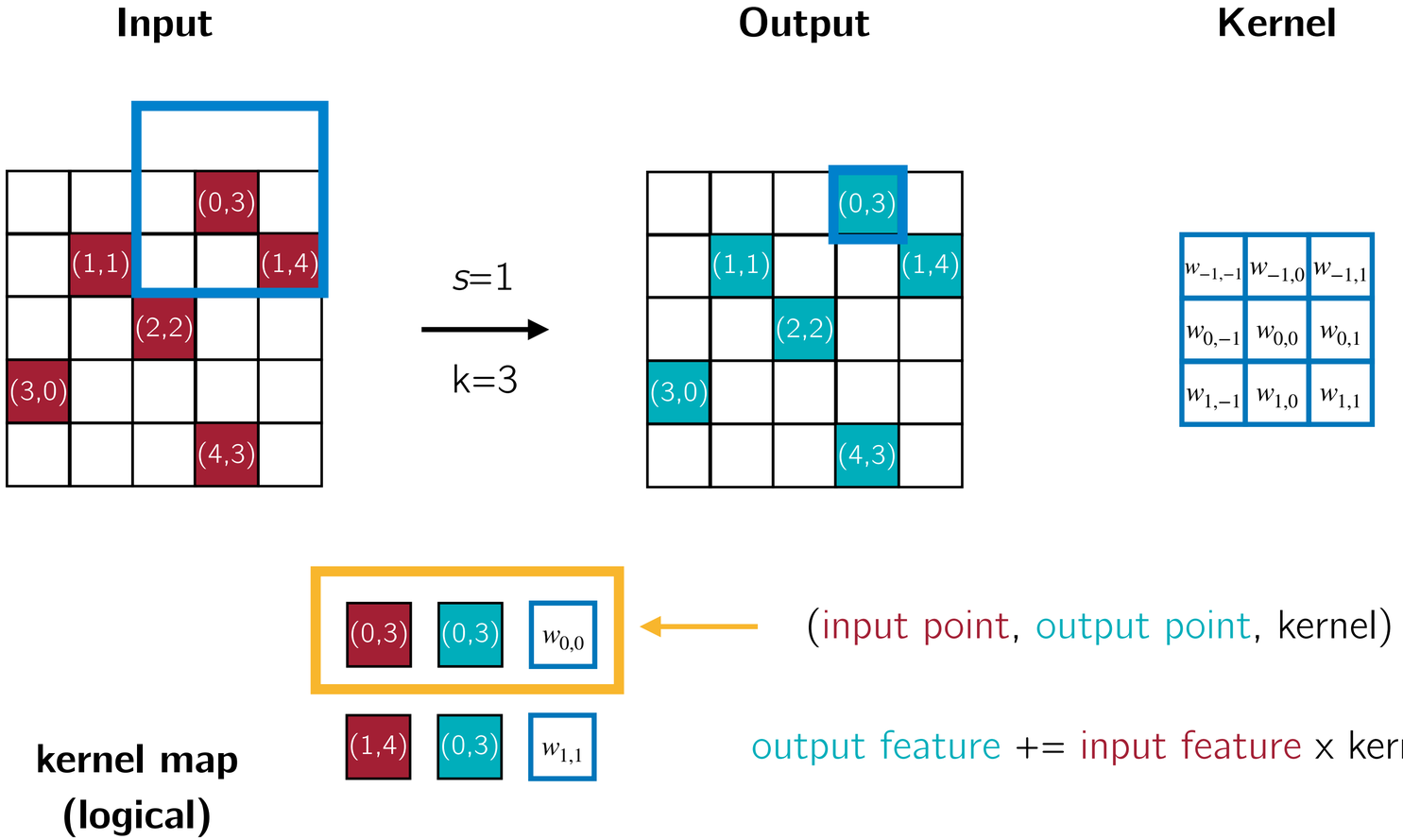
Details: Small Kernel Matters



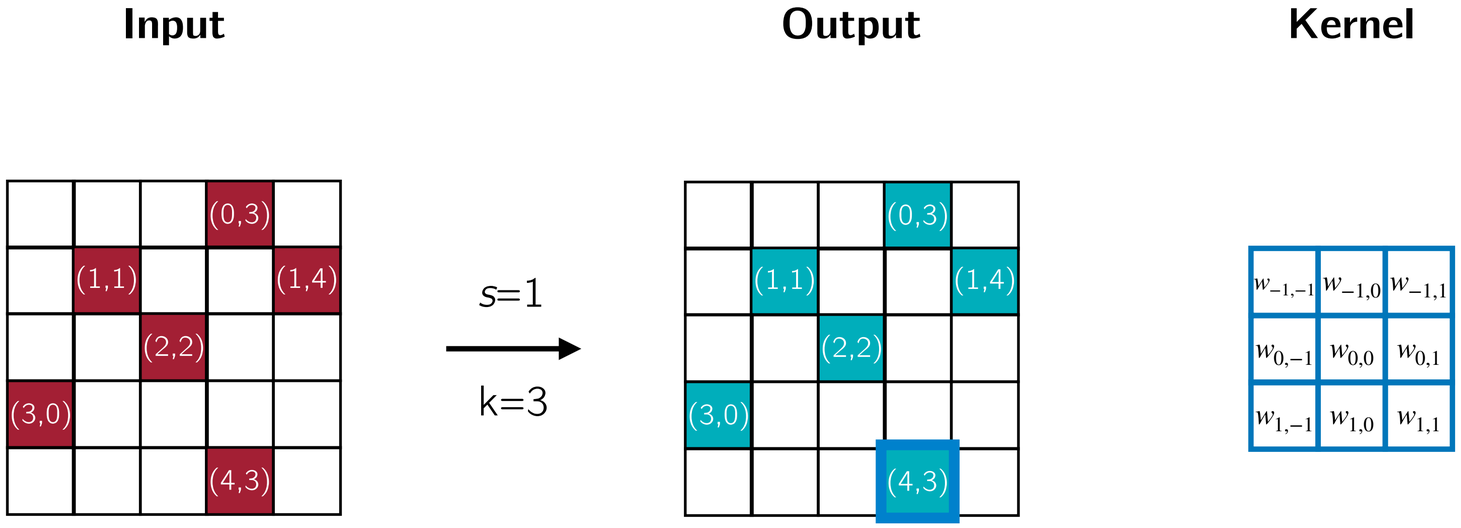
Details: Small Kernel Matters



Details: Small Kernel Matters



Details: Small Kernel Matters

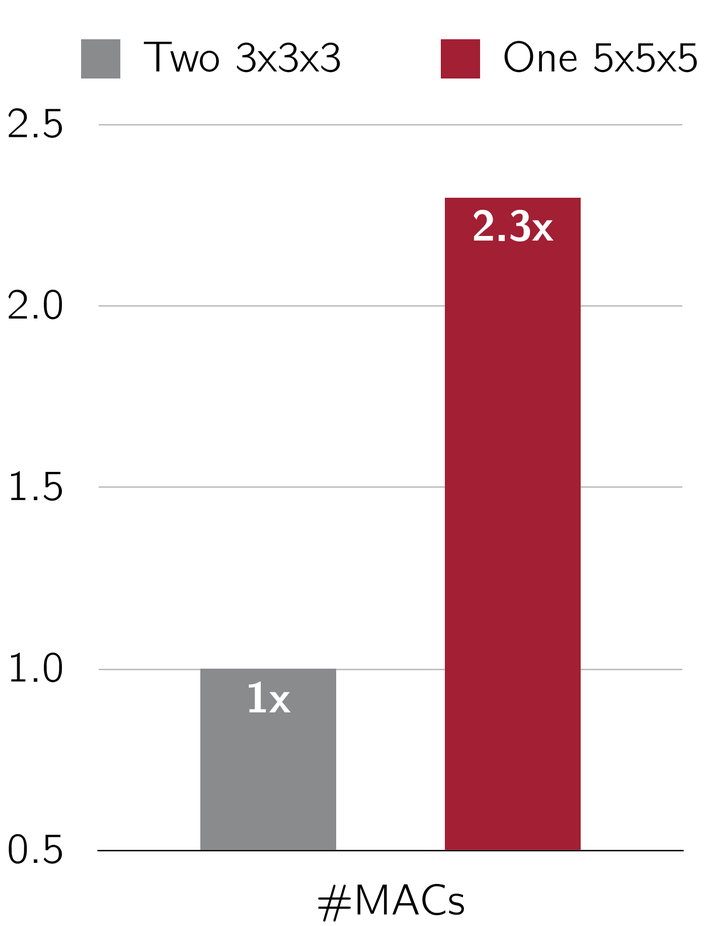


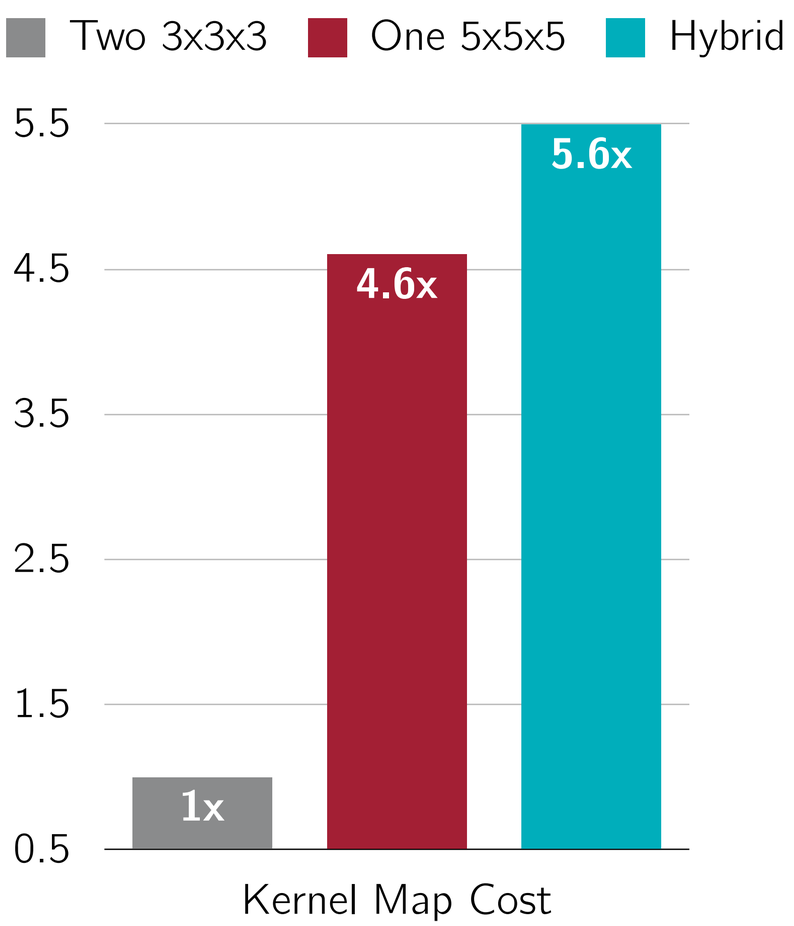
kernel map (logical) (0,3) (0,3) $w_{0,0}$ (input point, output point, kernel)

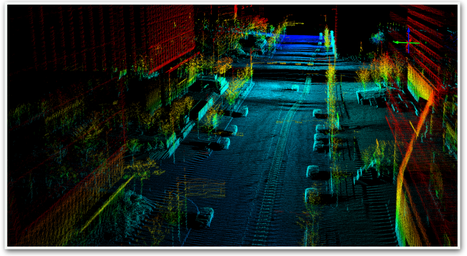
3D sparse hash map query
$O(k^3N)$

kernel map (physical) 0 0 0 (input point index, output point index, kernel index)

Details: Small Kernel Matters







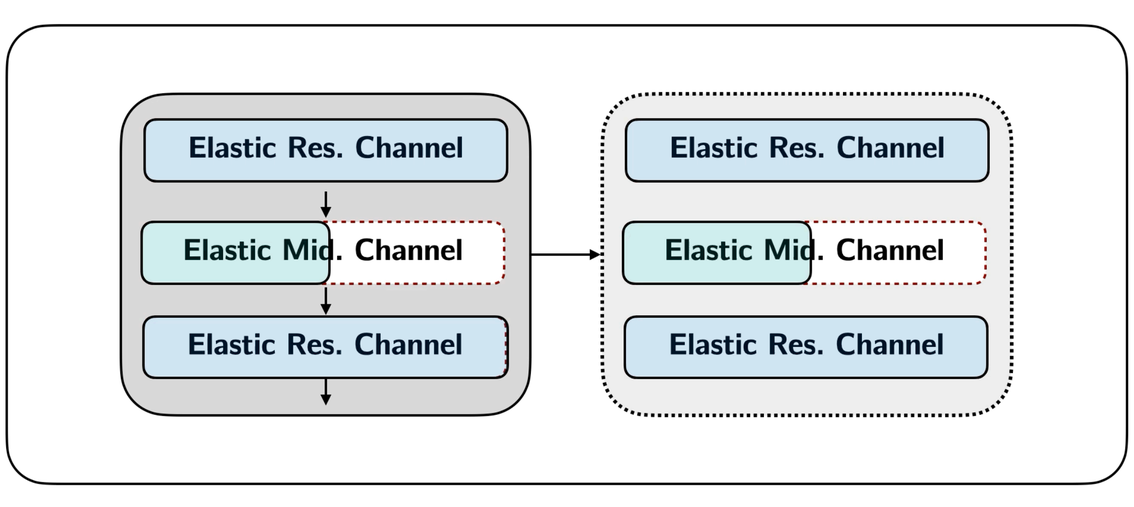
3D Deep Learning

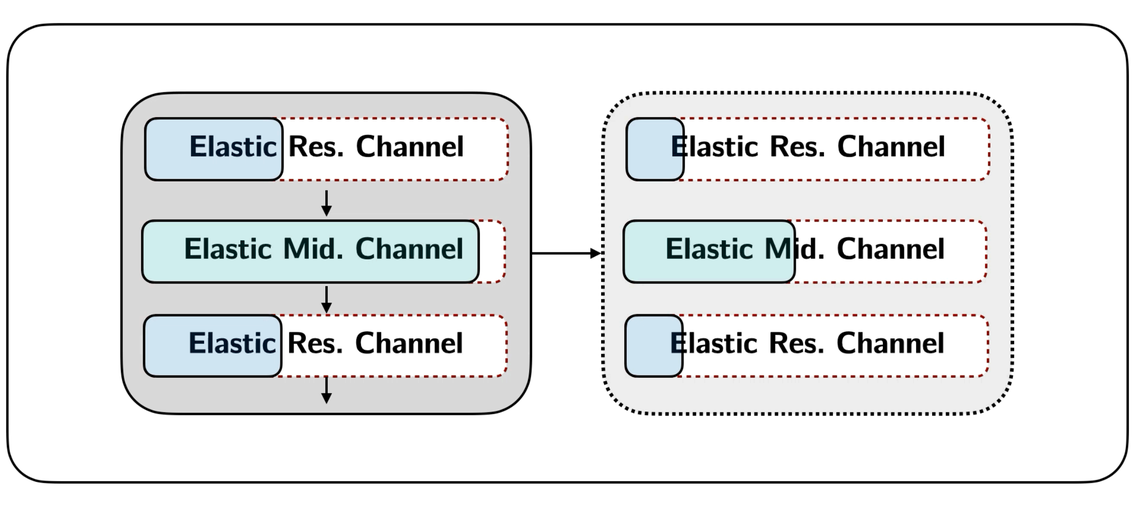
Small Kernels

Large Kernels

Cost of large kernels in 3D deep learning is more prohibitive than 2D.

Details: Fine-Grained Channel Numbers

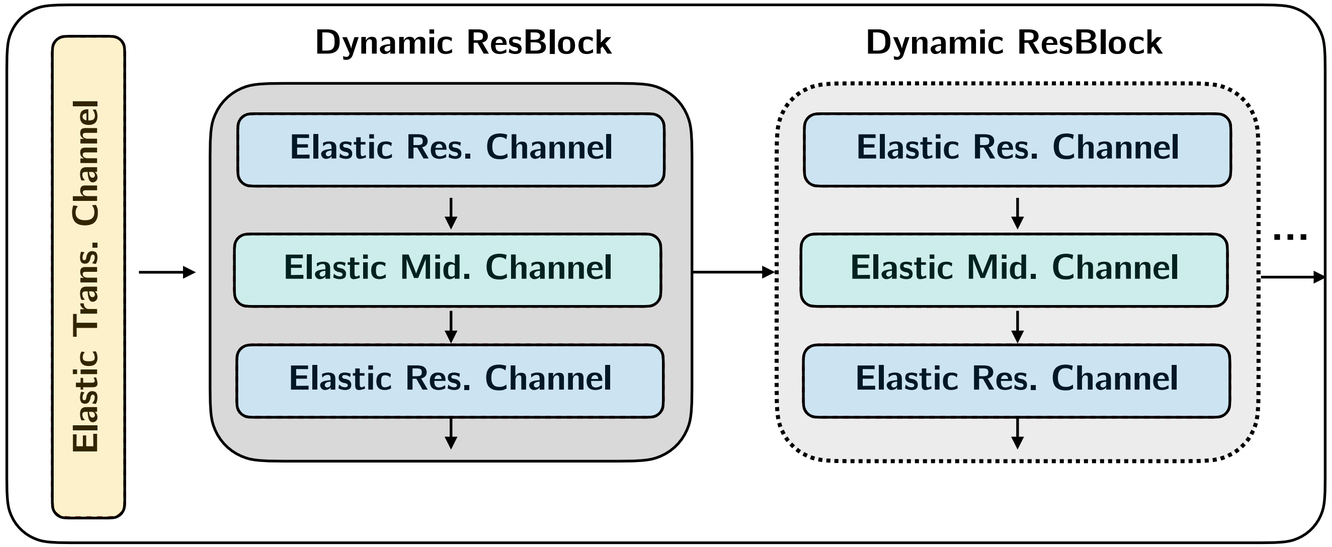




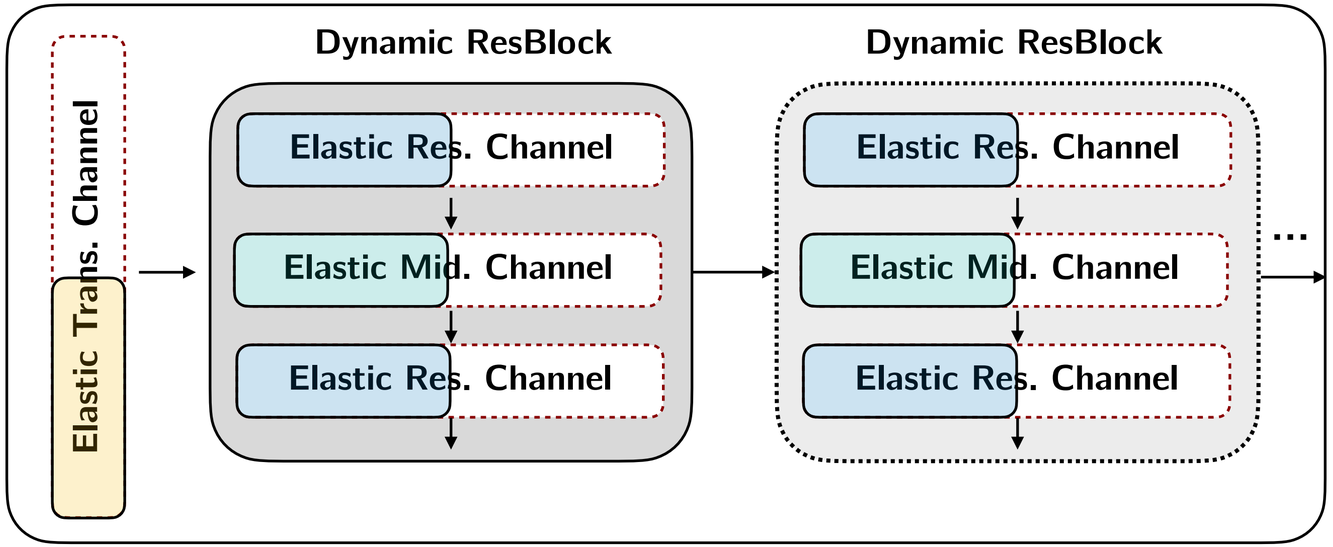
	2D-NAS Network Channel Design Space	3D-NAS Network Channel Design Space
C_{in}/C_{out}	Fixed	**Flexible**
C_{mid}	Only 2-3 Choices	**O(n) choices**
# MACs range	2 x	**16 x**

3D-NAS explores a much larger channel design space comparing with 2D-NAS.

Details: Elastic Network Depths



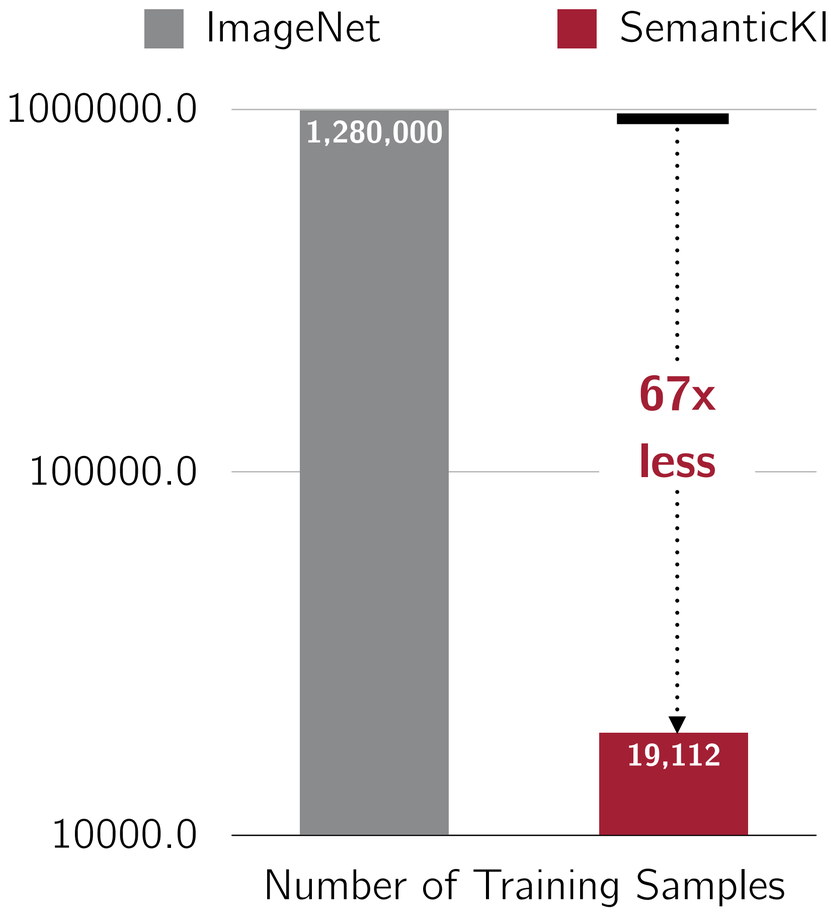
Details: Elastic Network Depths

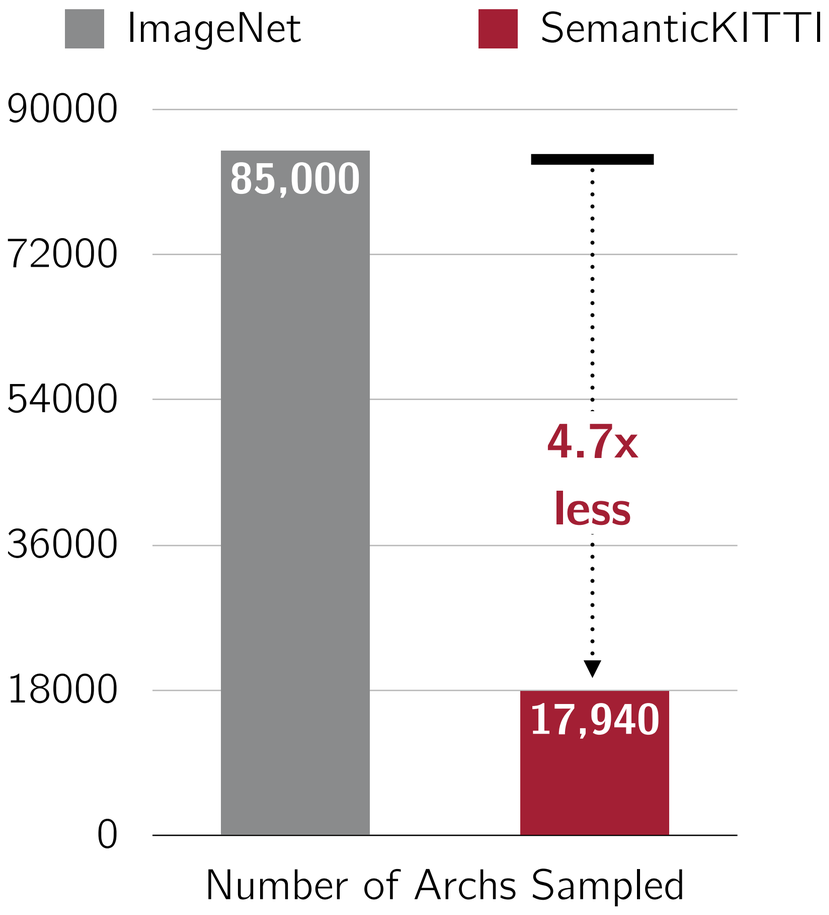


FLOPs: 7.5G → 1.9G (4x)

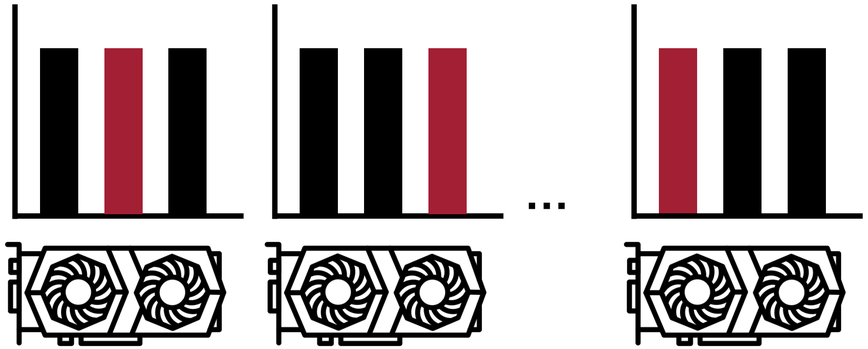
Latency: 105 ms → 96 ms (1.1x)

Challenge: Sample Efficiency



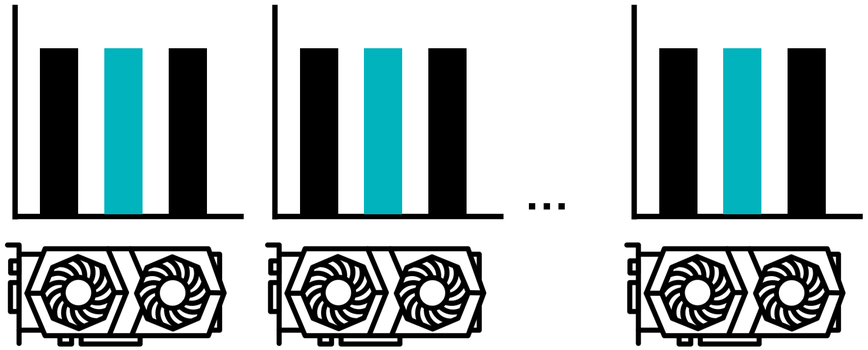


Solution: Distributed Sampling Across GPUs



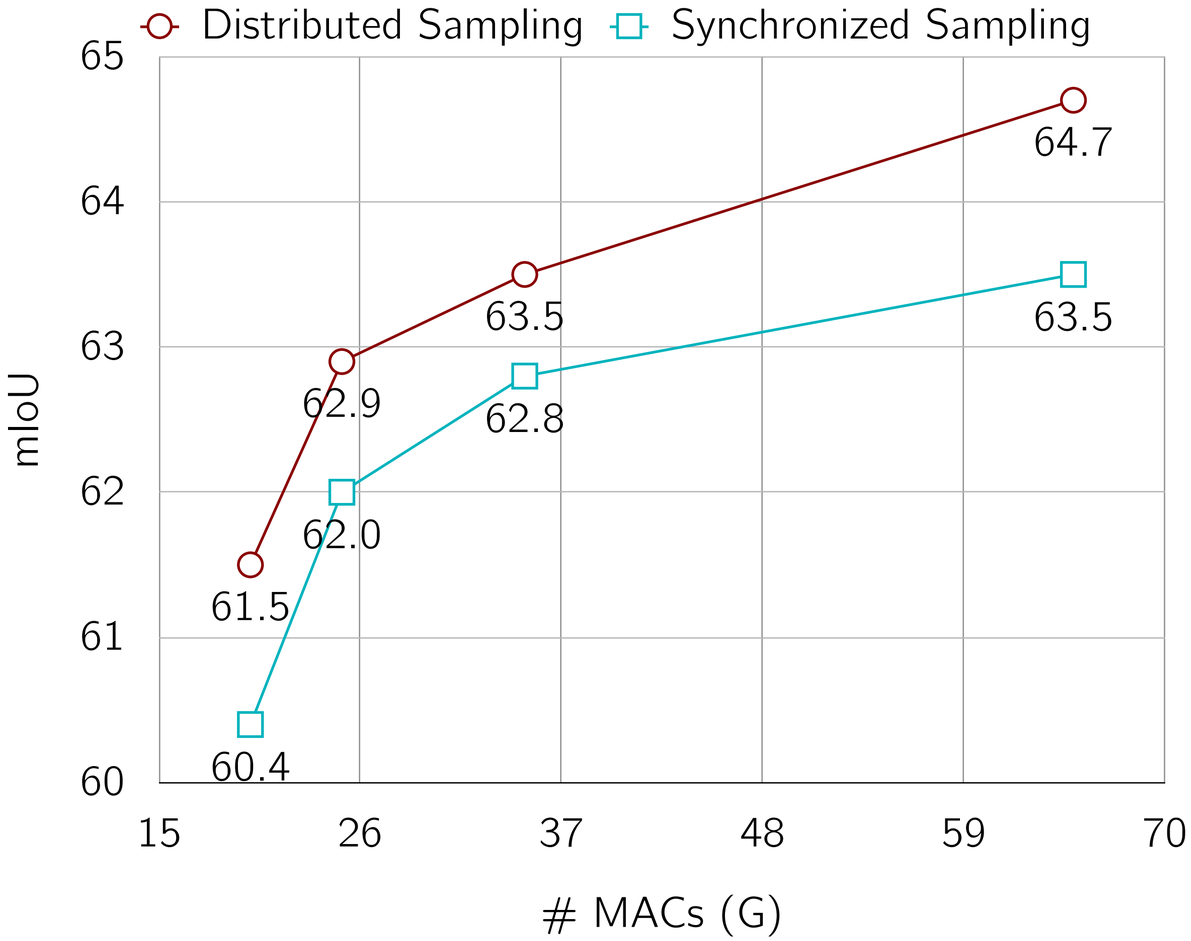
Distributed Sampling

Different sub-networks on different GPUs



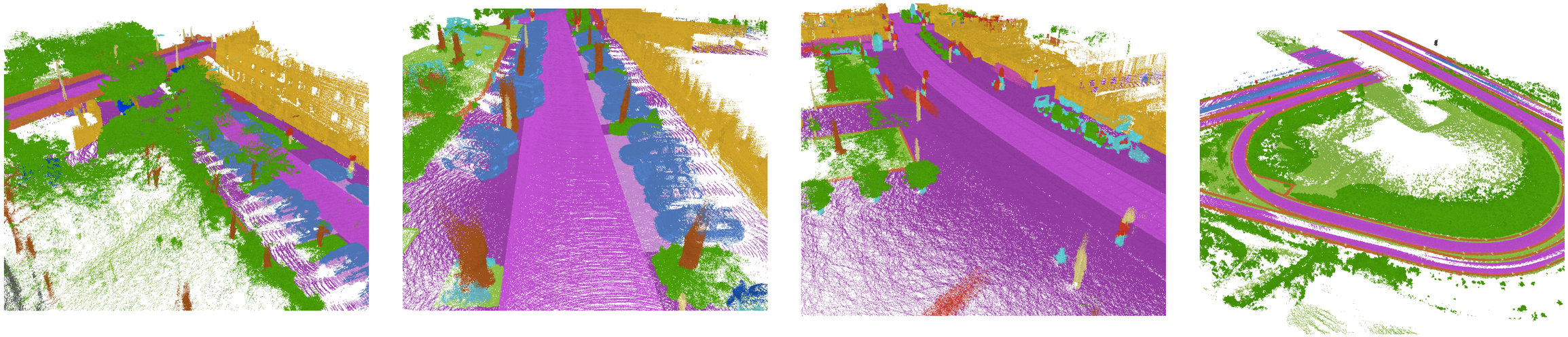
Synchronized Sampling

The same sub-networks on different GPUs



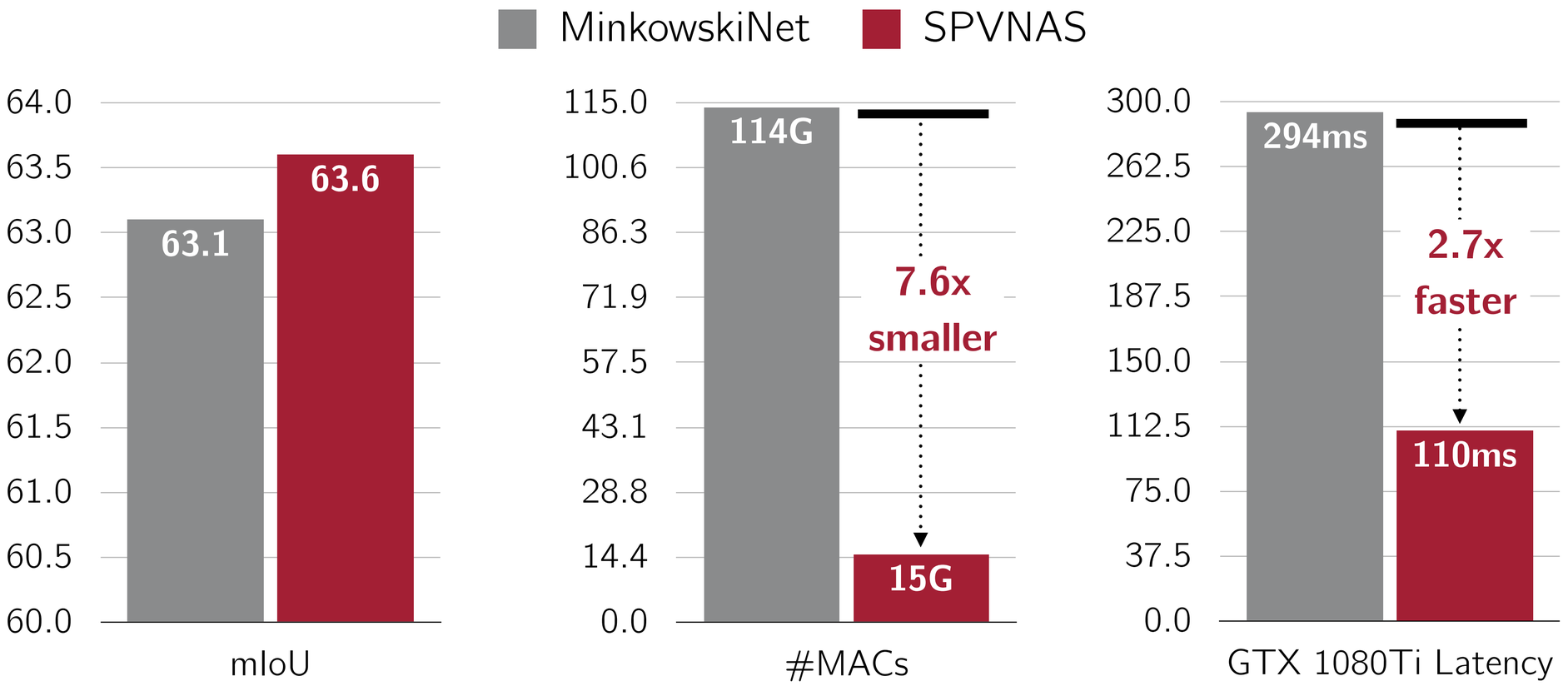
Results: 3D Semantic Scene Segmentation

- The resulting **SPVNAS** achieves new **state-of-the-art** results on **SemanticKITTI** leaderboard.



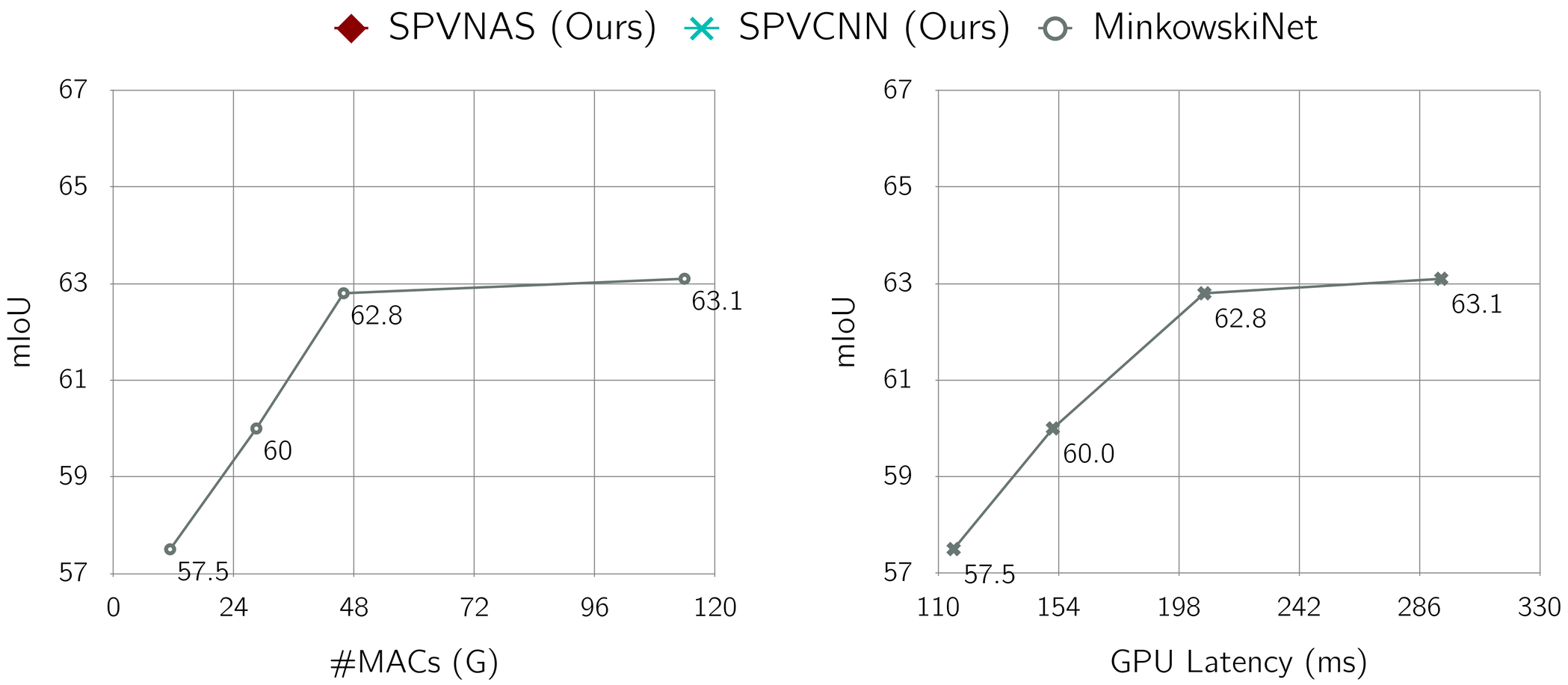
- SemanticKITTI is the **largest** 3D point cloud semantic segmentation dataset. It is **29x** larger than ScanNet, **160x** larger than S3DIS.
- SemanticKITTI is collected from **real driving scenarios**, and provides **point-level** annotation for **video sequences**.

Results: 3D Semantic Scene Segmentation



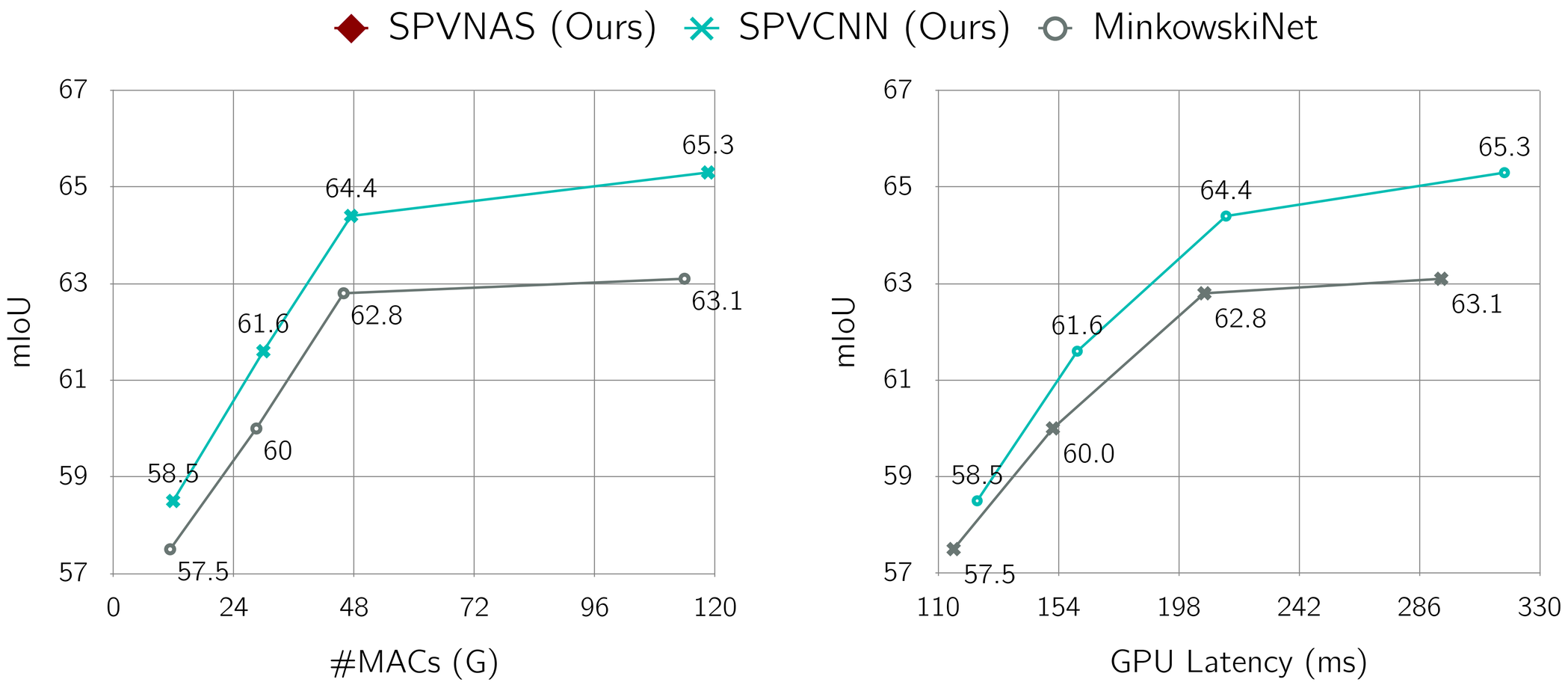
We achieve **8x** MACs reduction and **3x** speedup over MinkowskiNet with **SPVNAS**

Results: 3D Semantic Scene Segmentation



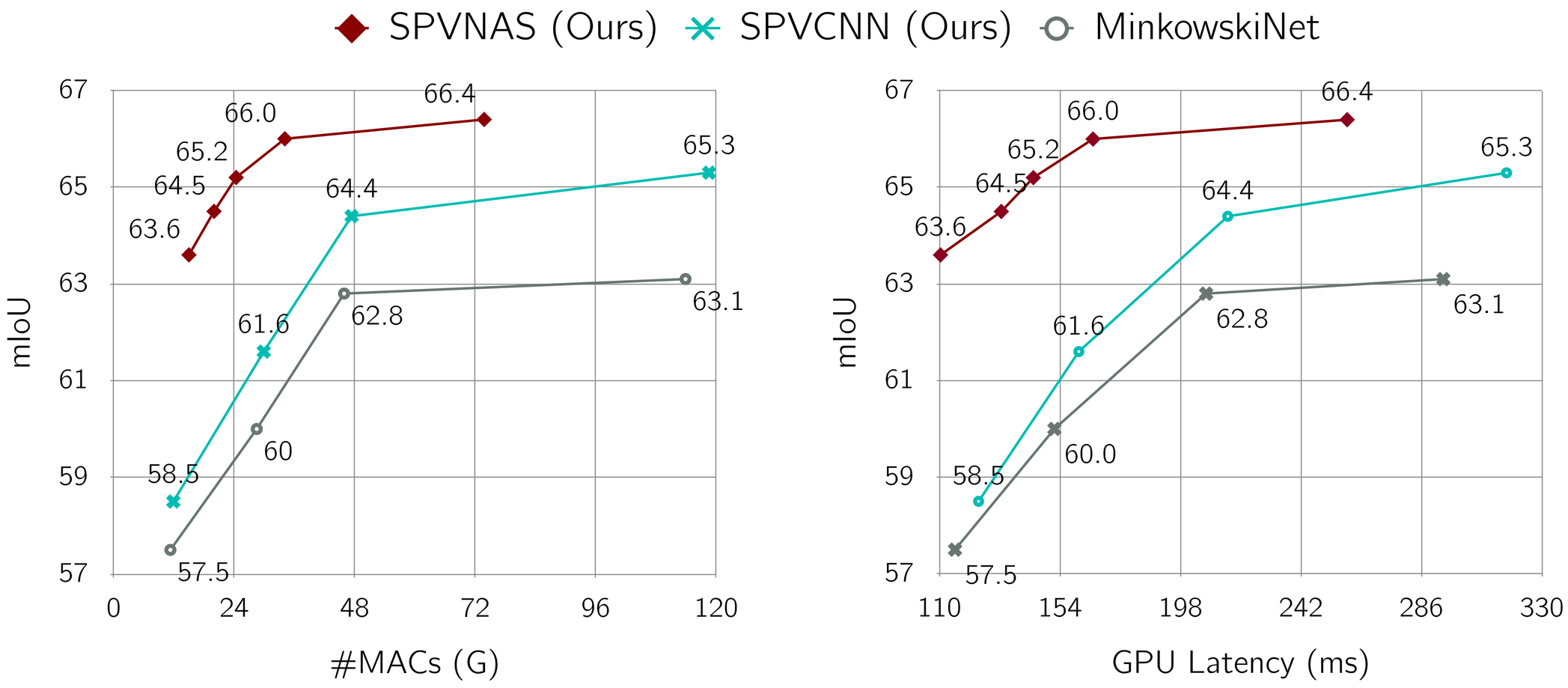
Both a better module (**SPVConv**) and **3D-NAS** improve the performance of MinkowskiNet.

Results: 3D Semantic Scene Segmentation



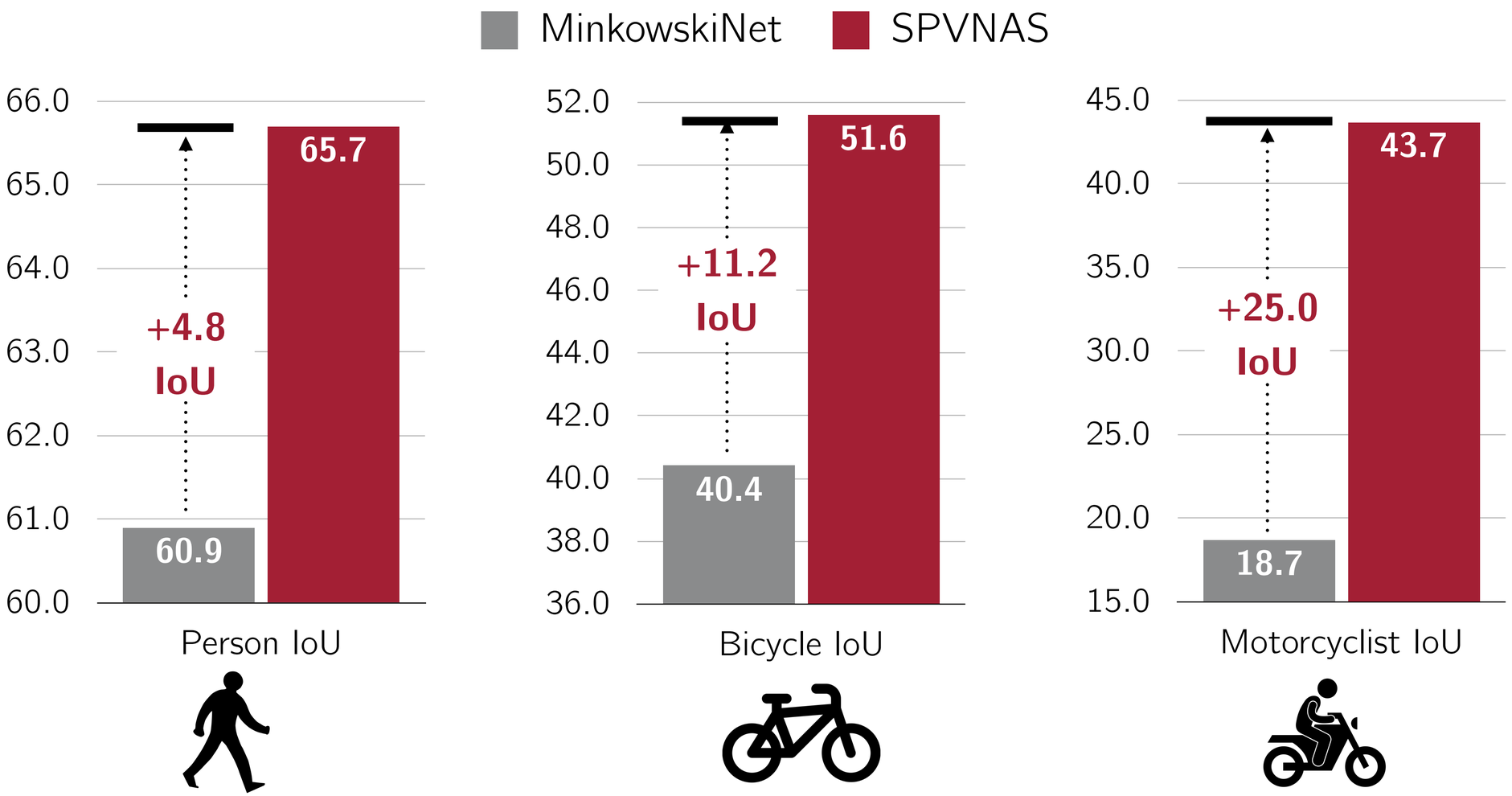
Both a better module (**SPVConv**) and **3D-NAS** improve the performance of MinkowskiNet.

Results: 3D Semantic Scene Segmentation



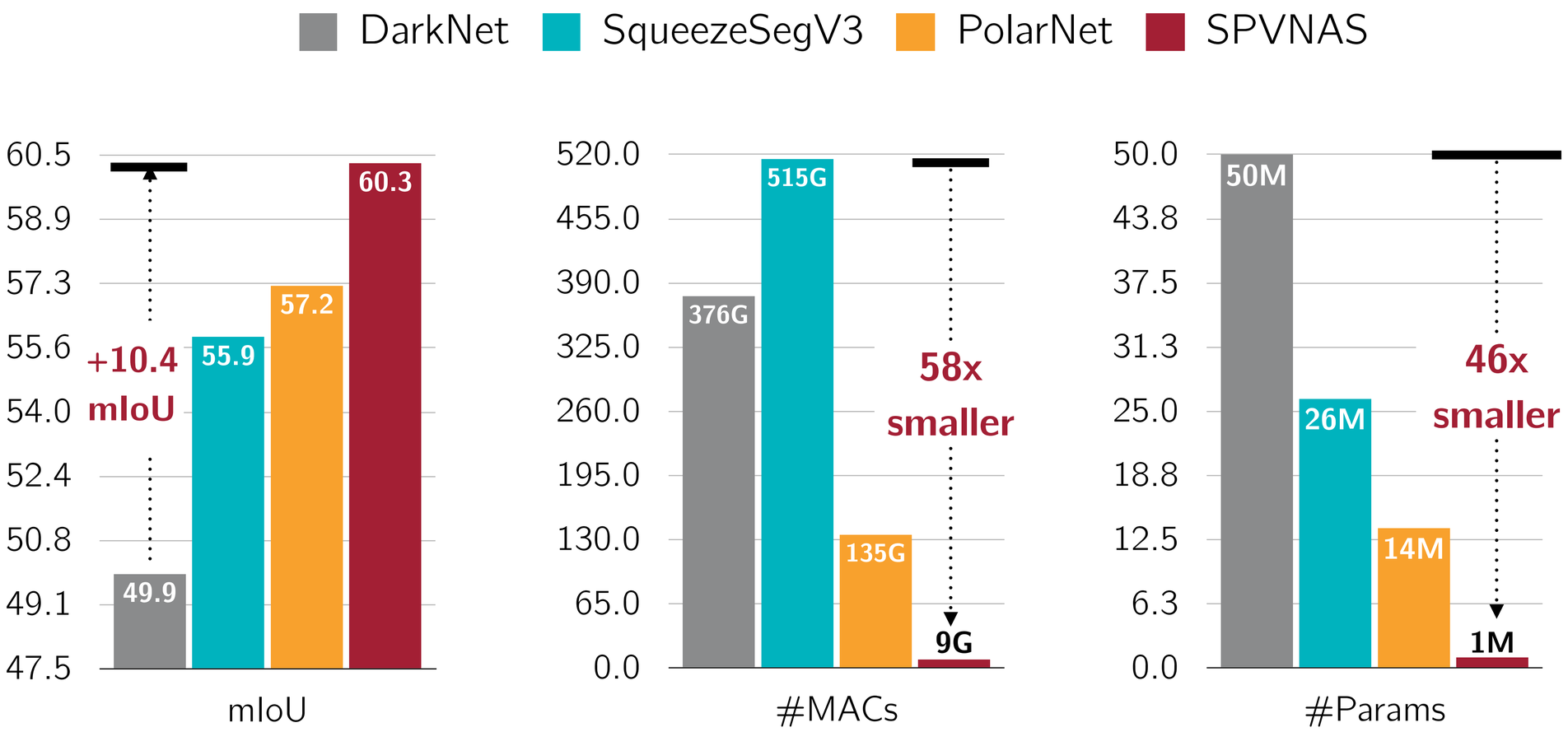
Both a better module (**SPVConv**) and **3D-NAS** improve the performance of MinkowskiNet.

Results: 3D Semantic Scene Segmentation



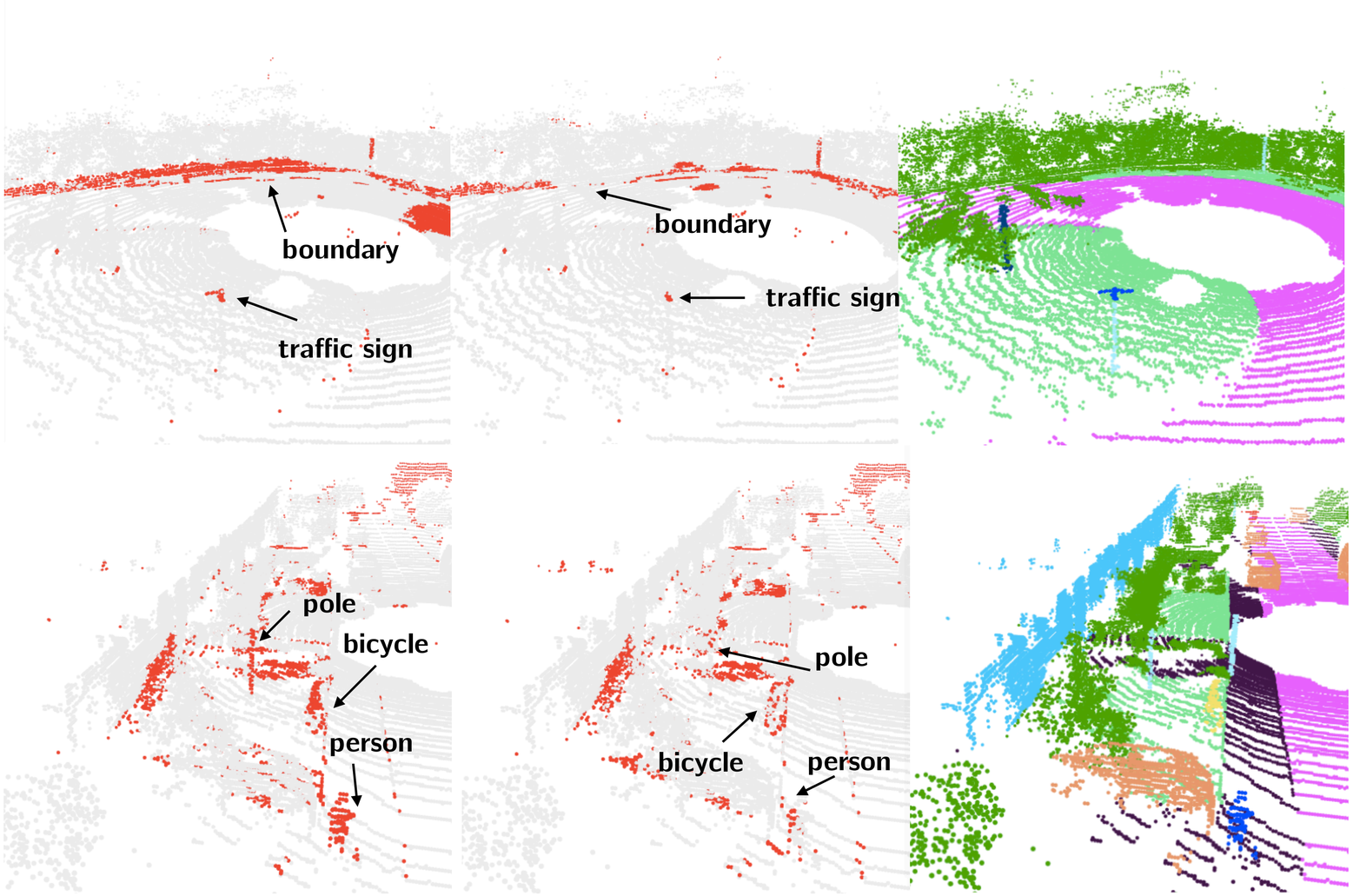
We achieve up to **25 mIoU** improvement on **safety-critical** small objects.

Results: 3D Semantic Scene Segmentation



We achieve up to **58x** MACs reduction and **46x** params reduction over projection-based methods.

Results: 3D Semantic Scene Segmentation



MinkowskiNet

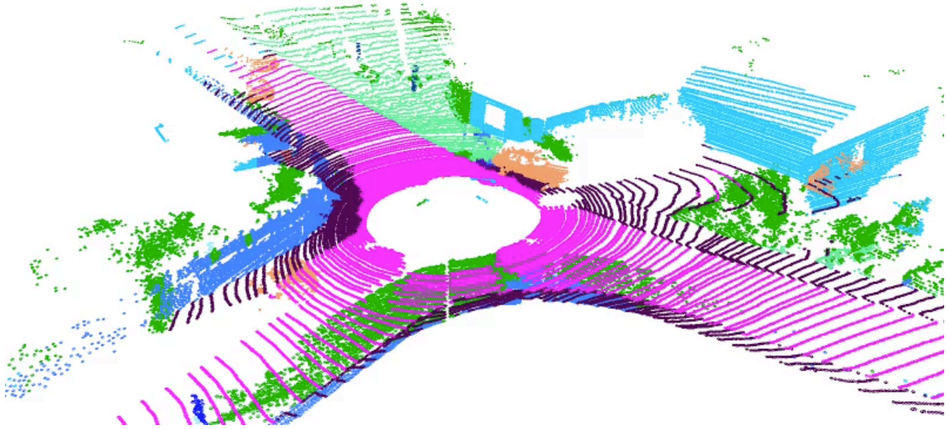
SPVNAS (**Ours**)

less error (**red**)

Ground Truth

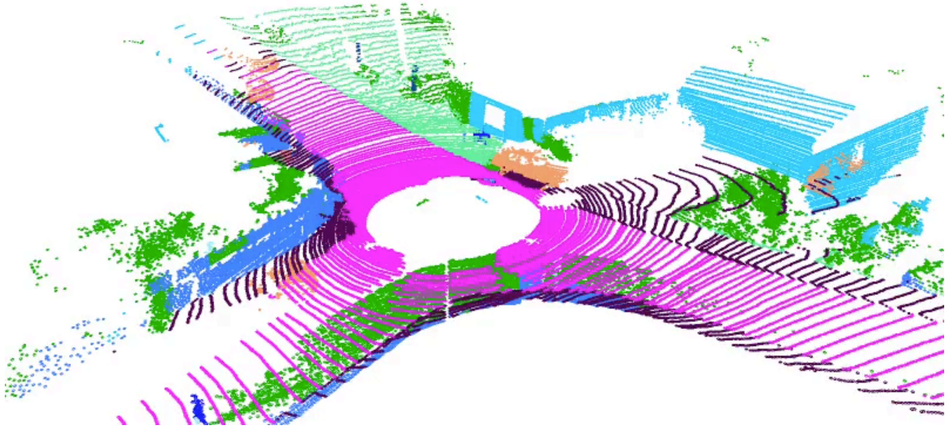
Demo: Significantly Faster than MinkowskiNets

MinkowskiNet



Mean IoU: **63.1** Throughput: **3.4** FPS
(**21.7M** Params **114.0G** FLOPs)

SPVNAS (Ours)



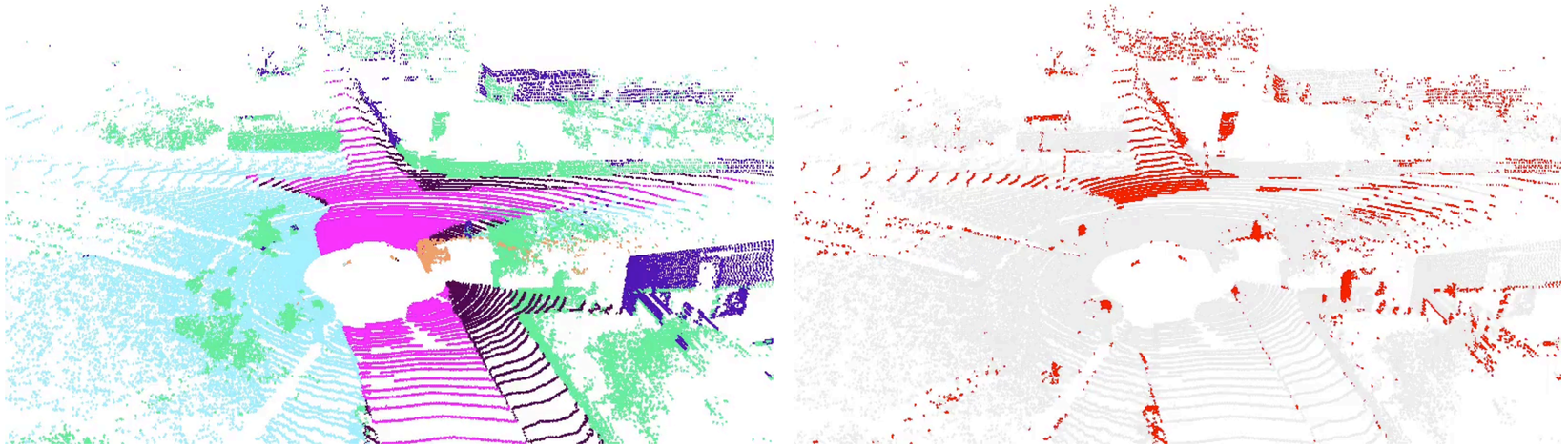
Mean IoU: **63.6** Throughput: **9.1** FPS
(**2.6M** Params **15.0G** FLOPs)

SPVNAS outperforms the state-of-the-art MinkowskiNet (with **3x** measured speedup and **8x** model size reduction).

Demo: Faster than 2D, Accuracy of 3D

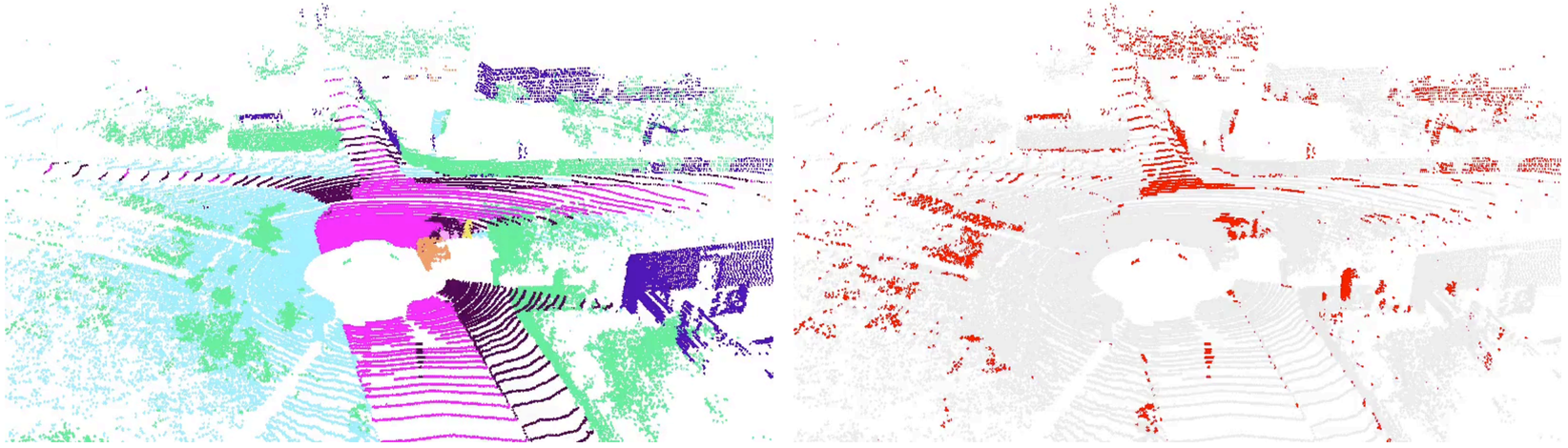
DarkNet53Seg

Mean IoU: **49.9**
Throughput: **9.7** FPS
50.4M Params
376.3G FLOPs



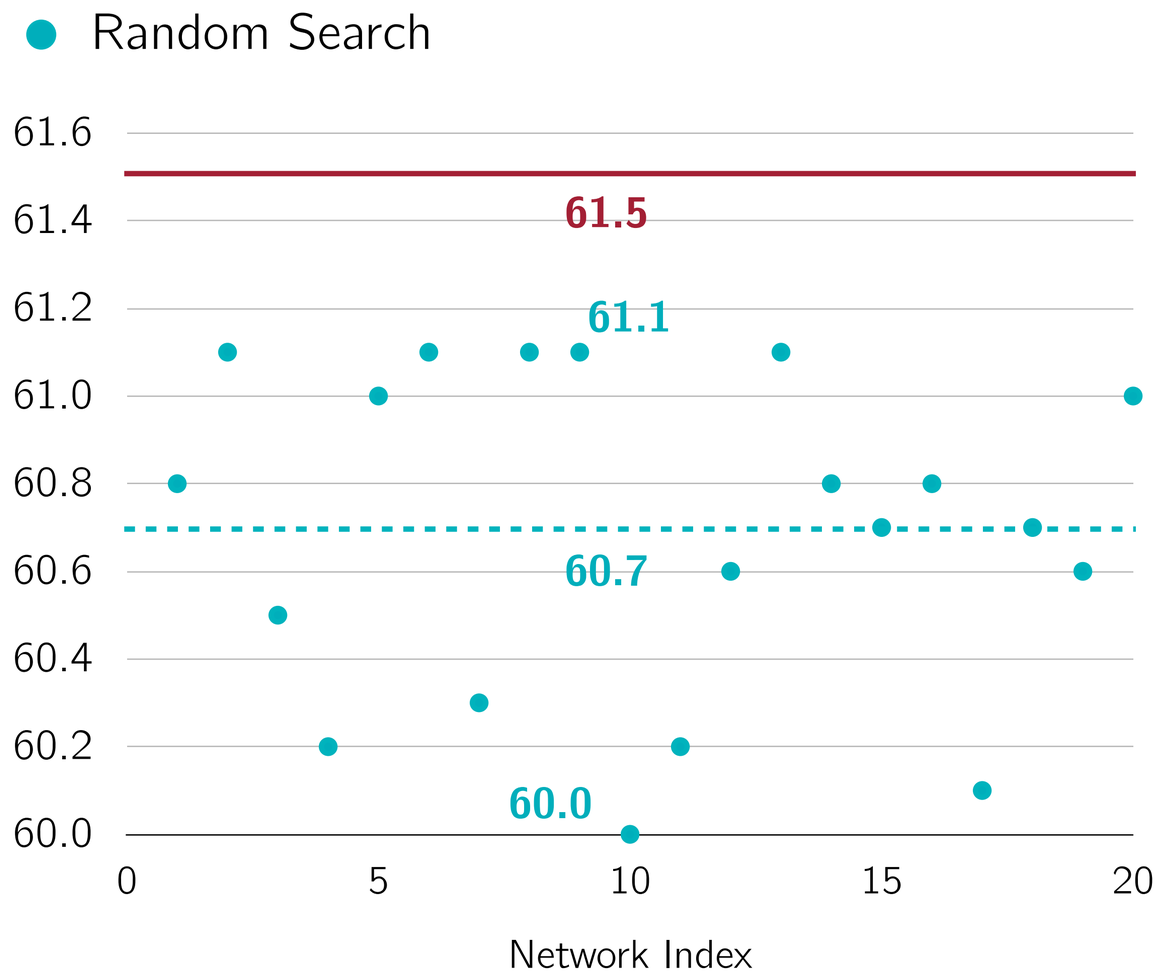
SPVNAS (Ours)

Mean IoU: **60.3** (> **KPConv**)
Throughput: **11.2** FPS
1.1M Params
8.9G FLOPs



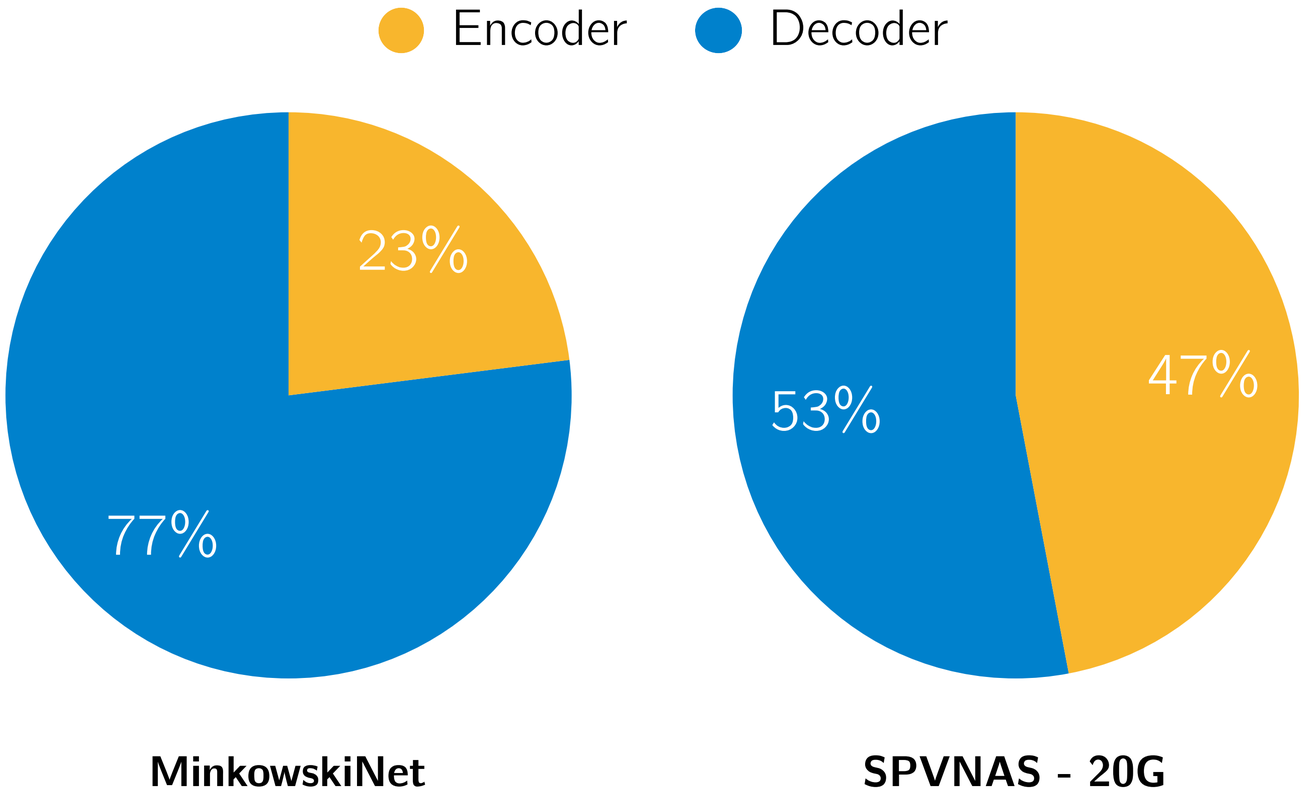
SPVNAS makes **fewer errors** (**in red**) than the 2D baseline model (with **46x** model size reduction and **42x** computation reduction).

Results: Evolutionary Architecture Search



Evolutionary Search discovers better model comparing with **Random Search**.

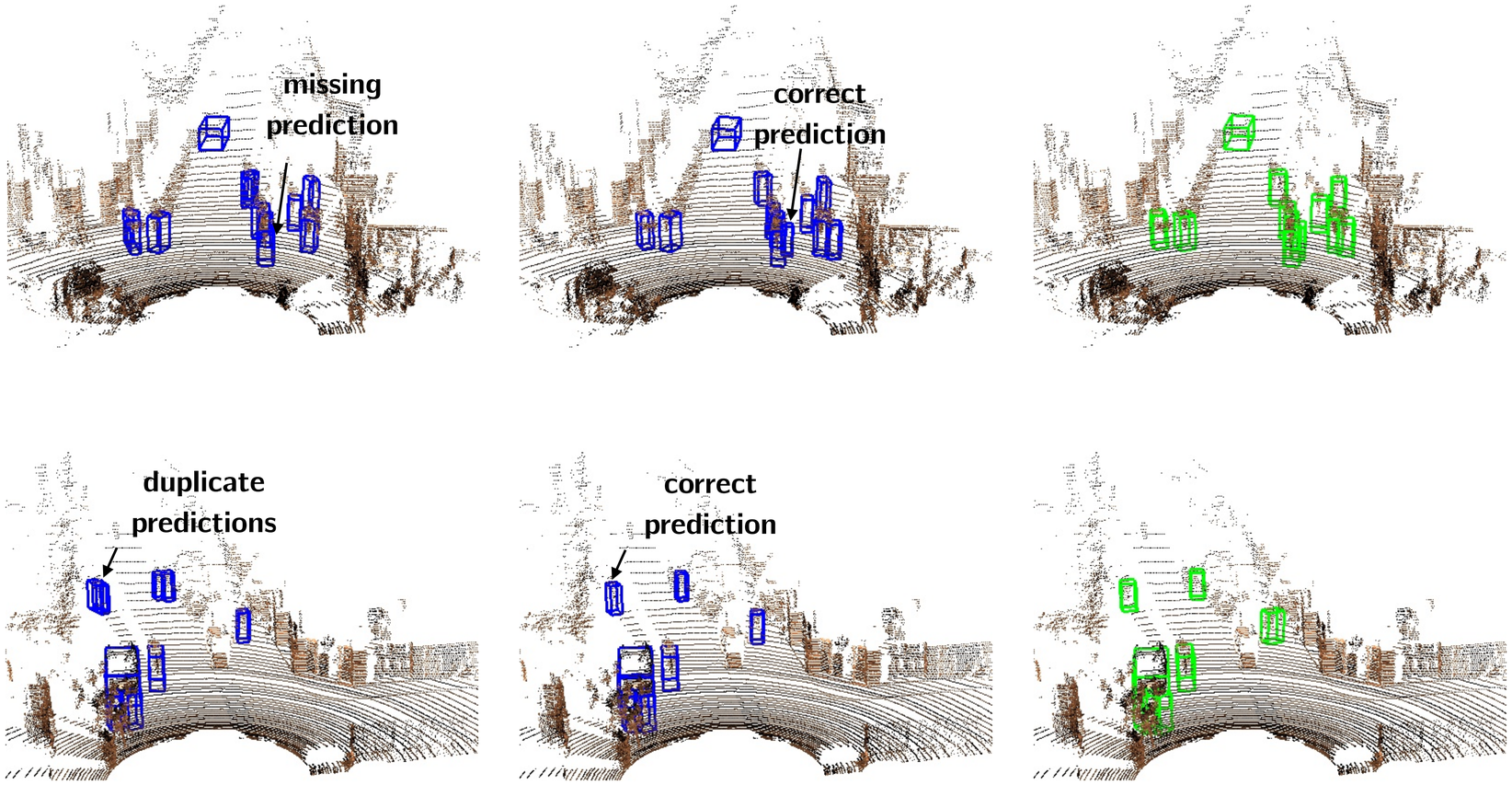
Results: Evolutionary Architecture Search



MinkowskiNet **SPVNAS - 20G**

SPVNAS balances the encoder / decoder computation ratio.

Results: 3D Object Detection



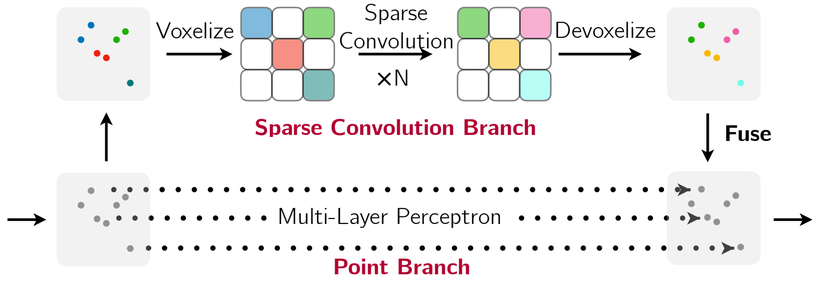
Detection By
SECOND

More Accurate Detection By
SPVCNN

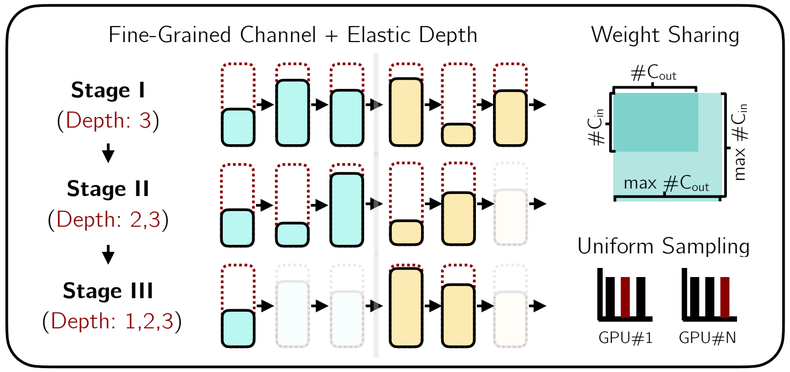
Ground Truth

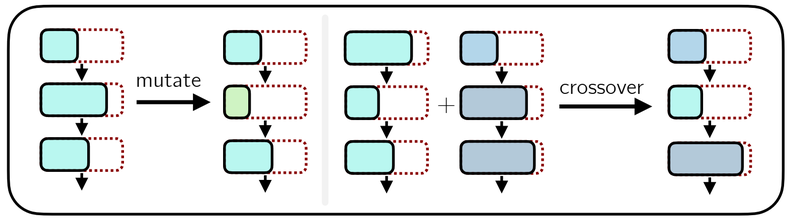
Summary

- We introduce **SPVConv** for **efficiently modeling small objects in large-scale 3D scenes**.



- We present **3D-NAS** to automatically design 3D architectures built with SPVConv, achieving **3 times speedup** and **8 times computation reduction** comparing with previous state of the art.





Summary

- The resulting **SPVNAS** achieves new **state-of-the-art** results on **SemanticKITTI** leaderboard.

Single Scan | Multiple Scans

Approach	Paper	Code	mIoU	Classes (IoU)	Details
SPVNAS			67.0		
KPRNet			63.1		
SalsaNext			59.5		
KPConv			58.8		
SqueezeSegV3			55.9		
3D-MiniNet			55.8		
PolarNet			54.3		
RandLA-Net			53.9		

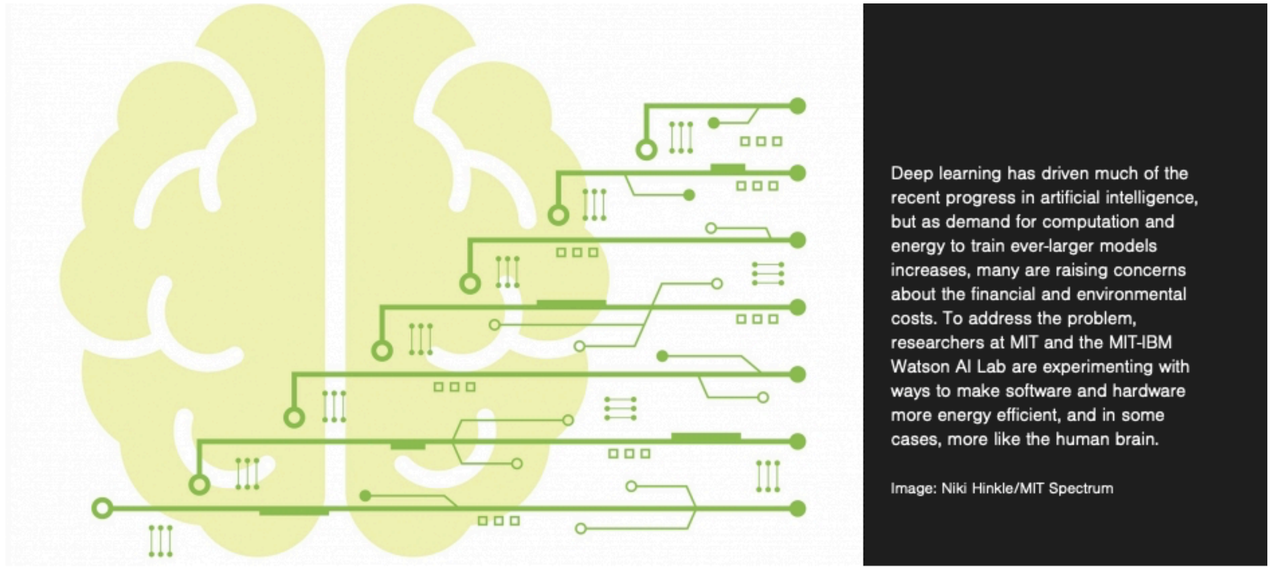
Rank **1st** among published methods

#	User	Entries	Date of Last Entry	mIoU ▲	accuracy ▲	road ▲	sidewalk ▲	parking ▲	other-ground ▲	building ▲	car ▲	car (moving) ▲	truck ▲
1	MIT-HAN-LAB	3	07/16/20	0.670 (1)	0.914 (3)	0.902 (48)	0.754 (15)	0.676 (5)	0.218 (51)	0.916 (7)	0.972 (1)	- (-)	0.566 (1)
2	DEEPROUTE.AI-DL	3	07/02/20	0.666 (2)	0.908 (7)	0.896 (60)	0.731 (49)	0.627 (29)	0.078 (78)	0.912 (13)	0.969 (2)	- (-)	0.544 (2)
3	Alibaba-ADLab	7	04/27/20	0.655 (3)	0.914 (2)	0.916 (17)	0.764 (5)	0.678 (4)	0.283 (16)	0.914 (10)	0.965 (3)	- (-)	0.535 (3)
4	JS3C-Net	6	06/20/20	0.640 (4)	0.905 (15)	0.894 (65)	0.723 (58)	0.611 (41)	0.284 (15)	0.921 (2)	0.951 (15)	- (-)	0.530 (4)
5	Shuangjie	1	06/11/20	0.637 (5)	0.907 (9)	0.916 (14)	0.764 (4)	0.636 (22)	0.309 (8)	0.896 (30)	0.960 (7)	- (-)	0.364 (30)
6	Noah_Canada	3	07/21/20	0.631 (6)	0.907 (10)	0.897 (57)	0.745 (27)	0.663 (6)	0.287 (13)	0.913 (12)	0.942 (24)	- (-)	0.400 (20)

Rank **1st** among all methods on the real-time leaderboard

Our Project Webpage: https://spvnas.mit.edu

Summary



Deep learning has driven much of the recent progress in artificial intelligence, but as demand for computation and energy to train ever-larger models increases, many are raising concerns about the financial and environmental costs. To address the problem, researchers at MIT and the MIT-IBM Watson AI Lab are experimenting with ways to make software and hardware more energy efficient, and in some cases, more like the human brain.

Image: Niki Hinkle/MIT Spectrum

Shrinking deep learning's carbon footprint

Through innovation in software and hardware, researchers move to reduce the financial and environmental costs of modern artificial intelligence.

Media Coverage: MIT News

http://news.mit.edu/2020/shrinking-deep-learning-carbon-footprint-0807

Our Project Webpage: https://spvnas.mit.edu

TinyML and Efficient AI

MIT HAN Lab

Hardware, AI and Neural-nets

github.com/mit-han-lab

youtube.com/c/MITHANLab

songhan.mit.edu

Media:

MIT Technology Review

IEEE SPECTRUM

WIRED

engadget

MIT News
ON CAMPUS AND AROUND THE WORLD

AI DAILY

VentureBeat

ScienceDaily

Analytics Insight

AI Business