LargeKernel3D

Scaling up Kernels in 3D Sparse CNNs

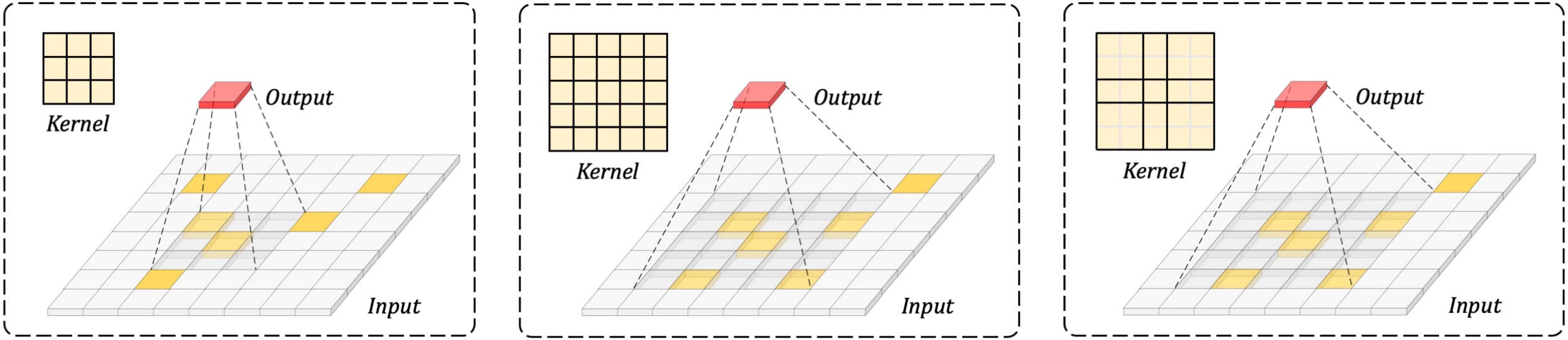
Yukang Chen[1], Jianhui Liu[2], Xiangyu Zhang[3], Xiaojuan Qi[2], Jiaya Jia[1]

[1] Chinese University of Hong Kong, [2] University of Hong Kong, [3] MEGVII Technology

https://github.com/dvlab-research/LargeKernel3D

LargeKernel3D: Scaling up Kernels in 3D Sparse CNNs

- 1. Motivation



Small-kernel sparse conv | Large-kernel sparse conv | Spatial-wise group conv

- Small-kernel sparse conv:
  <u>*Limited receptive field*</u> - not only by kernel size, but also by **feature disconnection.**
- Large-kernel sparse conv:
  <u>Large parameters and computation cost</u> - 3^3 --> 7^3
- **Spatial-wise group conv: Large receptive field & limited cost.**

LargeKernel3D: Scaling up Kernels in 3D Sparse CNNs

- 1. Motivation

- **Issues in plain Large-kernel sparse conv**

- 1. Efficiency issue

  Large amount of parameters and computation cost

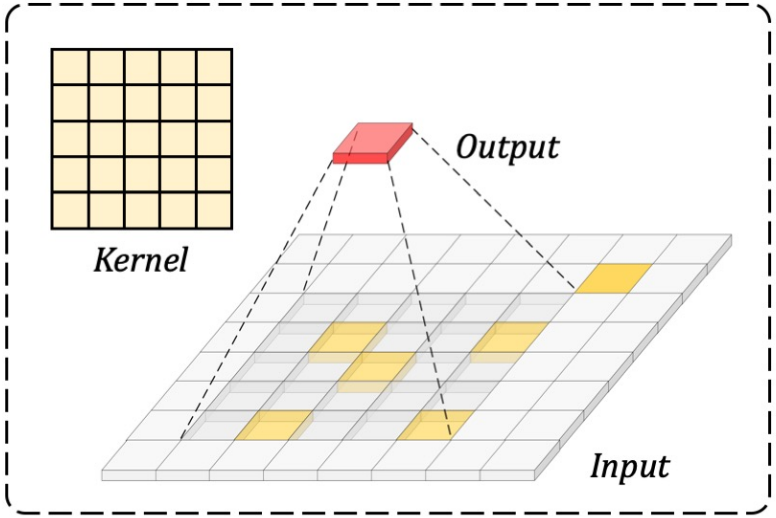
  - *e.g.* kernels from 3^3 to 7^3, from 27x to 343x

- 2. Optimization issue

  Large model size *v.s.* limited sparse data.

  - *Amount*: Point cloud data amounts are limited, (compared large-scale datasets on 2D vision tasks).

  - *Sparsity*: Not all weights are activated each time.



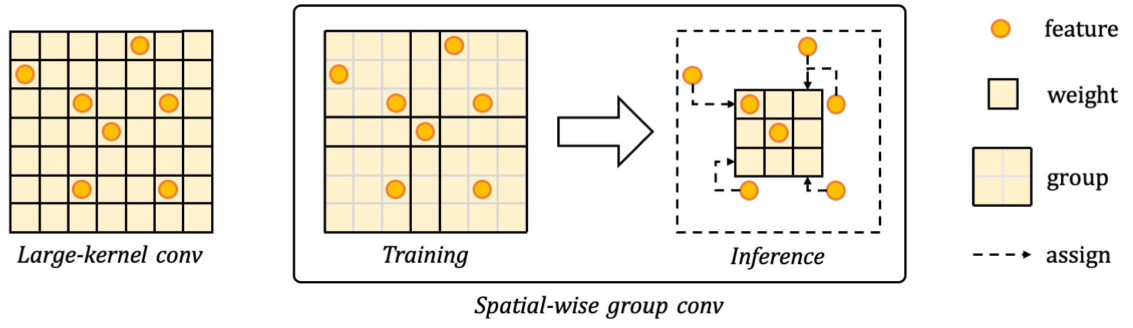
Large-kernel sparse conv

- **Results of MinkowskiNet-34 on ScanNetv2 semantic segmentation.**

Method	Params	FLOPs	Runtime	mIoU (%)
Baseline (Kernel 3^3)	37.9 M	182.8 G	108 ms	71.7
Baseline (Kernel 5^3)	170.3 M	537.5 G	212 ms	70.7
Baseline (Kernel 7^3)	465.0 M	1089.5 G	487 ms	68.6

LargeKernel3D: Scaling up Kernels in 3D Sparse CNNs

- 2. Our solution



Method	Params	FLOPs	Runtime	mIoU (%)
Baseline (Kernel 3^3)	37.9 M	182.8 G	108 ms	71.7
Baseline (Kernel 5^3)	170.3 M	537.5 G	212 ms	70.7
Baseline (Kernel 7^3)	465.0 M	1089.5 G	487 ms	68.6
Dilated Conv [4]	37.9 M	100.1 G	98 ms	64.6
Pooling + Dilated Conv	37.9 M	183.2 G	115 ms	nan
Spatial group conv [61]	37.9 M	127.2 G	96 ms	70.0
Deformable Conv [10]	42.5 M	250.1 G	238 ms	70.4
SW-LKNet-34	45.3 M	209.3 G	152 ms	**73.2**

- Efficiency:

    Training: 7x7 kernel (spatial weight sharing).

    Inference: 7x7 indice assign → Atomic Add → 3x3 conv.

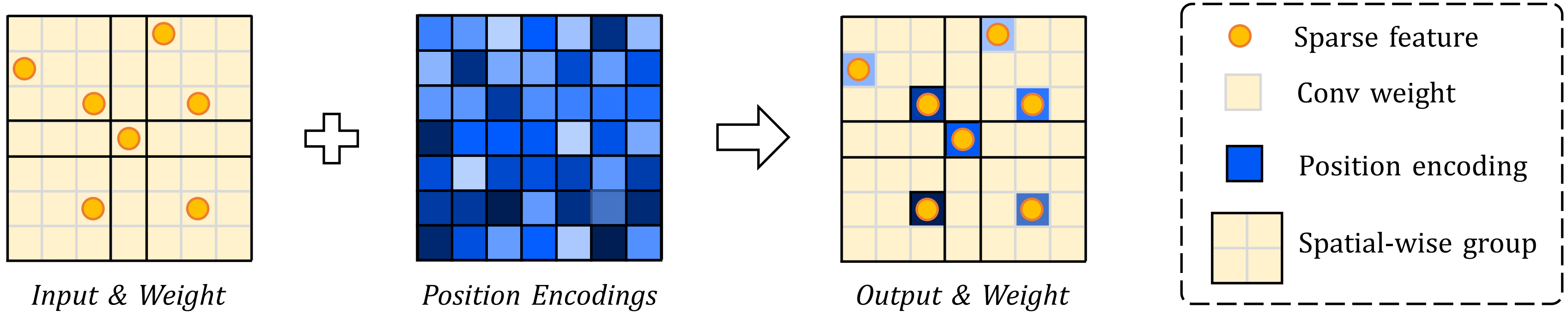
- Performance:

    Not all kernel weights are optimized during training + data insufficient.

    Weight sharing --> better learning.

    Comparison to other related convolutional schemes.

- 2. Our solution



Kernel Size		3×3×3	5×5×5	7×7×7	9×9×9	11×11×11	13×13×13	15×15×15	17×17×17
Plain	Params	6.9 K	32.0 K	87.8 K	186.6 K	340.7 K	562.4 K	864.0 K	1.3 M
	Latency	2.5 ms	4.2 ms	8.9 ms	17.5 ms	31.1 ms	55.1 ms	81.1 ms	106.3 ms
Ours	Params	-	8.9 K	12.4 K	18.6 K	28.2 K	42.1 K	60.9 K	85.5 K
	Latency	-	3.4 ms	3.9 ms	4.8 ms	6.2 ms	8.4 ms	11.4 ms	15.8 ms

LargeKernel3D: Scaling up Kernels in 3D Sparse CNNs

• 3. Main results

Table 6: Comparison with other methods on nuScenes *test* split.

Method	NDS	mAP	Car	Truck	Bus	Trailer	C.V.	Ped	Mot	Byc	T.C.	Bar
PointPillars [30]	45.3	30.5	68.4	23.0	28.2	23.4	4.1	59.7	27.4	1.1	30.8	38.9
3DSSD [62]	56.4	42.6	81.2	47.2	61.4	30.5	12.6	70.2	36.0	8.6	31.1	47.9
CBGS [68]	63.3	52.8	81.1	48.5	54.9	42.9	10.5	80.1	51.5	22.3	70.9	65.7
CenterPoint [63]	65.5	58.0	84.6	51.0	60.2	53.2	17.5	83.4	53.7	28.7	76.7	70.9
HotSpotNet [6]	66.0	59.3	83.1	50.9	56.4	53.3	23.0	81.3	63.5	36.6	73.0	71.6
CVCNET [5]	66.6	58.2	82.6	49.5	59.4	51.1	16.2	83.0	61.8	38.8	69.7	69.7
TransFusion [2]	70.2	65.5	86.2	56.7	66.3	58.8	28.2	86.1	68.3	44.2	82.0	78.2
Focals Conv [9]	70.0	63.8	86.7	56.3	67.7	59.5	23.8	87.5	64.5	36.3	81.4	74.1
Focals Conv-F‡ [9]	73.6	70.1	87.5	60.0	69.9	64.0	32.6	89.0	81.1	59.2	85.5	71.8
LargeKernel3D	70.5	65.3	85.9	55.3	66.2	60.2	26.8	85.6	72.5	46.6	80.0	74.3
LargeKernel3D‡	72.8	68.8	87.3	59.1	68.5	65.6	30.2	88.3	77.8	53.5	82.4	75.0
LargeKernel3D-F‡	**74.2**	**71.1**	88.1	60.3	69.1	66.5	34.3	89.6	82.0	60.3	85.7	75.5

‡ Flipping and rotation testing-time augmentations.

- Effective on both 3D semantic segmentation and object detection.
- Semantic segmentation: ScanNetv2.
- Object Detection: KITTI, nuScenes, Waymo.

Table 4: Comparisons on ScanNetv2 mIoU on 3D semantic segmentation. † Sliding-window testing.

Method	*val*	***test***
PointCNN [31]	-	45.8
PointNet++ [47]	53.5	55.7
RandLA-Net [27]	-	64.5
PointConv [58]	61.0	66.6
PointASNL [59]	63.5	66.6
KPConv [54]	69.2	68.6
FusionNet [65]	-	68.8
Point Transformer† [67]	70.6	-
Fast Point Transformer [43]	72.1	-
SparseConvNet [20]	69.3	72.5
MinkowskiNet-42 [10]	-	73.4
Stratified Transformer† [29]	74.3	73.7
MinkowskiNet-34 (baseline)	71.7	-
LargeKernel3D	73.2	**73.9**

Table 6: Comparison on KITTI *val* split in AP_{3D} in Recall 11 for the *Car* category.

Method	Easy	**Mod.**	Hard
VoxelNet [39]	81.97	65.46	62.85
PointPillars [28]	86.62	76.06	68.91
SECOND [55]	88.61	78.62	77.22
Point R-CNN [45]	88.88	78.63	77.38
Part-A^2 [47]	89.47	79.47	78.54
3DSSD [57]	89.71	79.45	78.67
Pointformer [36]	90.05	79.65	78.89
SA-SSD [22]	90.15	79.91	78.78
PV-RCNN [46]	89.35	83.69	78.70
VoTr-TSD [35]	89.04	84.04	78.68
Pyramid-PV [34]	89.37	84.38	78.84
Focals Conv [7]	89.52	84.93	79.18
Voxel R-CNN [12]	89.41	84.52	78.93
SW-LKNet	89.52	**85.07**	79.32

LargeKernel3D: Scaling up Kernels in 3D Sparse CNNs

- 3. Main results

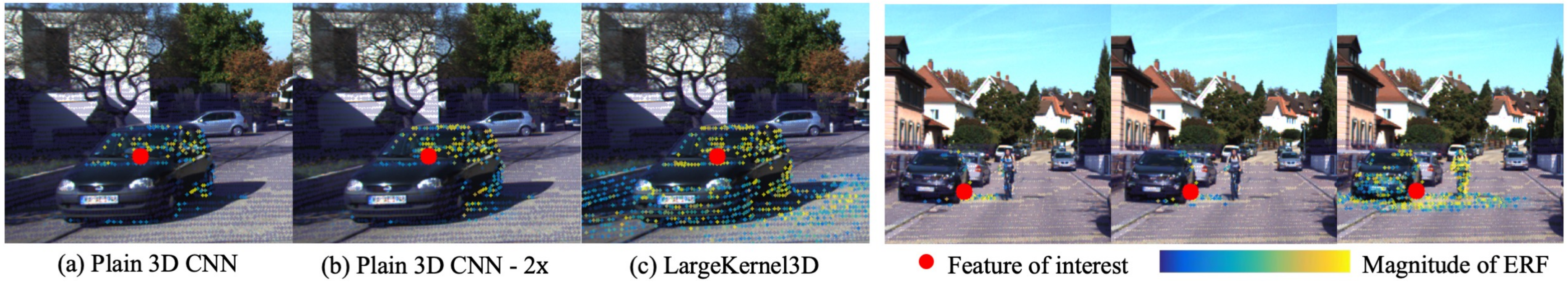


Table 4: Improvements over various kernel sizes on SW-LKNet upon CenterPoint and Waymo $\frac{1}{5}$.

Kernel	Runtime	Veh. L1	Veh. L2	Ped. L1	Ped. L2	Cyc. L1	Cyc. L2
3^3	109 ms	70.90	62.86	71.46	63.50	69.06	66.52
7^3	124 ms	71.87	63.80	71.66	63.73	70.40	67.82
11^3	145 ms	72.24	64.20	71.83	63.87	70.19	68.29
13^3	156 ms	72.46	64.35	73.71	65.81	70.85	68.25
15^3	168 ms	72.71	64.65	73.81	65.76	70.83	68.21
17^3	175 ms	**73.12** (+2.22)	**65.03** (+2.17)	**74.28** (+2.82)	**65.92** (+2.42)	**71.18** (+2.12)	**68.42** (+1.90)

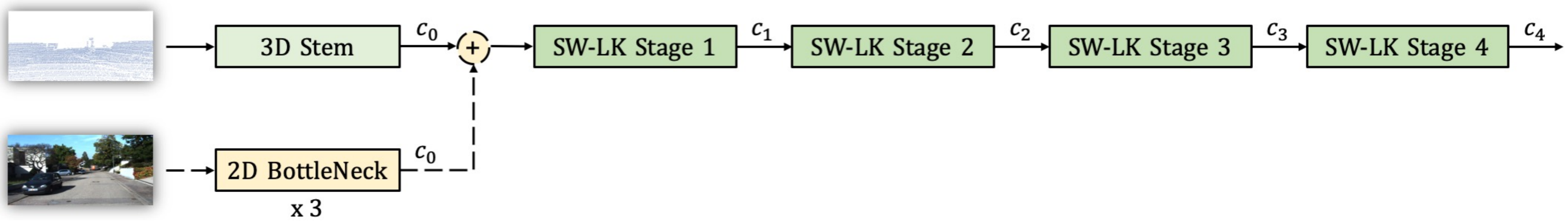
- Larger Receptive fields than plain 3D CNN and its 2x deep version.
- Scalable to 17x17x17 on the large-scale Waymo datasets.

LargeKernel3D: Scaling up Kernels in 3D Sparse CNNs

- 3. Main results

	Method					Metrics									
	Date	Name	Modalities	Map data	External data	mAP	mATE (m)	mASE (1-IOU)	mAOE (rad)	mAVE (m/s)	mAAE (1-acc)	NDS	PKL *	FPS (Hz)	Stats
			Any	All	All										
>	2022-06-03	BEVFusion-e	Camera, Lidar	no	no	0.750	0.242	0.227	0.320	0.222	0.130	0.761	0.518	n/a	
>	2022-01-13	FusionVPE	Camera, Lidar	no	no	0.733	0.235	0.227	0.284	0.243	0.128	0.755	0.529	n/a	
>	2021-05-25	Centerpoint-Fusion	Camera, Lidar, Rad	no	yes	0.724	0.237	0.227	0.318	0.211	0.133	0.749	0.491	n/a	
>	2022-06-16	LargeKernel-F	Camera, Lidar	no	no	0.711	0.236	0.228	0.298	0.241	0.131	0.742	0.555	n/a	
>	2021-12-29	PAI3D	Camera, Lidar	no	no	0.714	0.245	0.233	0.308	0.233	0.131	0.742	0.535	n/a	
>	2022-05-02	BEVFusion	Camera, Lidar	no	no	0.702	0.261	0.239	0.329	0.260	0.134	0.729	0.583	n/a	
>	2022-05-30	LargeKernel-L	Lidar	no	no	0.688	0.244	0.230	0.312	0.241	0.132	0.728	0.581	n/a	

- **1st Lidar, 4th multi-modal.**
- **Single-model results.**



Reference

[1] Benjamin Graham, Martin Engelcke, and Laurens van der Maaten. 3d semantic segmentation with submanifold sparse convolutional networks. In CVPR, pages 9224–9232, 2018.

[2] Shaoshuai Shi, Chaoxu Guo, Li Jiang, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. PV-RCNN: pointvoxel feature set abstraction for 3d object detection. In CVPR, pages 10526–10535, 2020.

[3] Jiajun Deng, Shaoshuai Shi, Peiwei Li, Wengang Zhou, Yanyong Zhang, and Houqiang Li. Voxel R-CNN: towards high performance voxel-based 3d object detection. In AAAI, pages 1201–1209, 2021.

[4] Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. Centerbased 3d object detection and tracking. In CVPR, pages 11784–11793, 2021.

[5] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The KITTI dataset. Int. J. Robotics Res., 32(11):1231–1237, 2013.

[6] Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In CVPR, pages 11618–11628, 2020.

Yukang Chen

- Third-year Ph.D student in CUHK
- Supervised by Jiaya Jia
- Research in Efficient Computer Vision
  - *AutoML, Autonomous driving, Multi-modality*
- More about me
  - https://yukangchen.com
  - https://scholar.google.com/citations?user=6p0ygKUAAAAJ&hl=en

- Thanks!