Sparse LiDAR and Stereo Fusion (SLS-Fusion) for Depth Estimation and 3D Object Detection

Nguyen Anh Minh Mai[1,2], Pierre Duthon[3], Louahdi Khoudour[1], Alain Crouzil[2], Sergio A. Velastin[4,5]

[1]Cerema, Equipe-projet STI, 31400 Toulouse, France

[2]IRIT, Université de Toulouse, UPS, 31062 Toulouse, France

[3]Cerema, Equipe-projet STI, 63017 Clermont-Ferrand Cedex 2, France

[4]Department of Computer Science and Engineering, Universidad Carlos III de Madrid, 28911 Leganés, Madrid, Spain

[5]School of Electronic Engineering and Computer Science, Queen Mary University of London, UK

March 17, 2021

Input

Camera & projected LiDAR 4 beam from two sides of stereo system

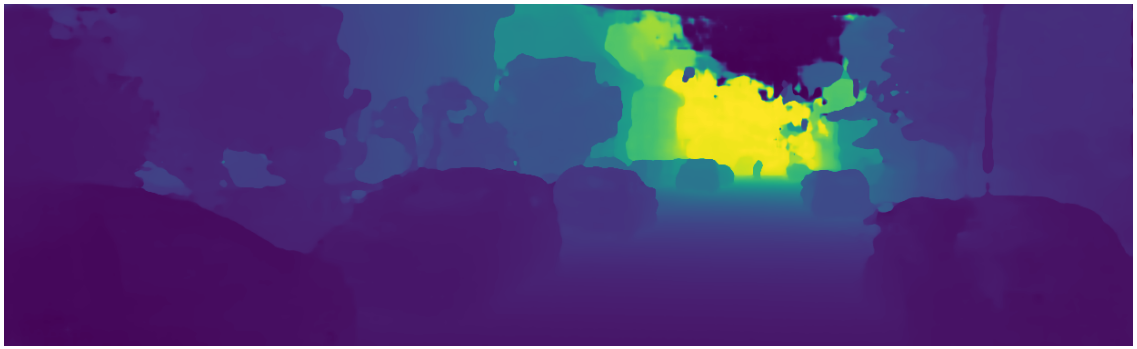
(a) Left side

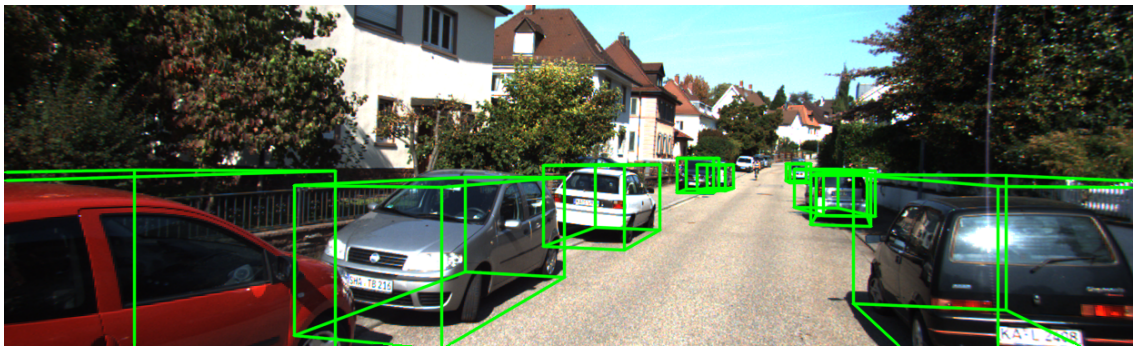
(b) Right side

Figure: KITTI dataset [CVPR, 2012]

Output

Depth map & 3D object detection output from our method





Overview

Overview

Background (1/ 3)

Why depth?

Background (1/ 3)

Why depth?

- Navigation & Mapping (spatial structure of a scene)

Background (1/ 3)

Why depth?

- Navigation & Mapping (spatial structure of a scene)
- Position, shape and appearance of objects in imagery

Figure: Example of depth map estimation result

Background (2/ 3)

Why Stereo & LiDAR 4 beams?

Background (2/ 3)

Why Stereo & LiDAR 4 beams?

- Stereo:

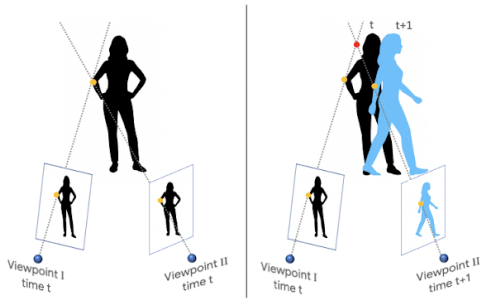


Figure: example of depth map prediction [CVPR, 2019]

Background (2/ 3)

Why Stereo & LiDAR 4 beams?

- Stereo:

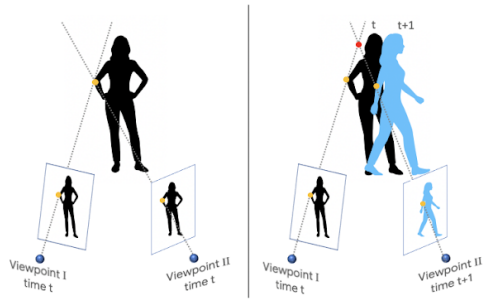


Figure: example of depth map prediction [CVPR, 2019]

- LiDAR: Accurate but expensive (75k $ for the most commonly LiDAR)

Figure: Velodyne HDL-64E: 64 beams, 75k $ (left)
ScaLa: 4 beams, 600 $ (right)

Background (3/ 3)

Why 3D object detection?

Background (3/ 3)

Why 3D object detection?

- Type of object (classification)

Background (3/ 3)

Why 3D object detection?

- Type of object (classification)
- Locating object in the image (2D object detection)

Background (3/ 3)

Why 3D object detection?

- Type of object (classification)
- Locating object in the image (2D object detection)
- Distance (from ego-vehicle) to the object (depth estimation)

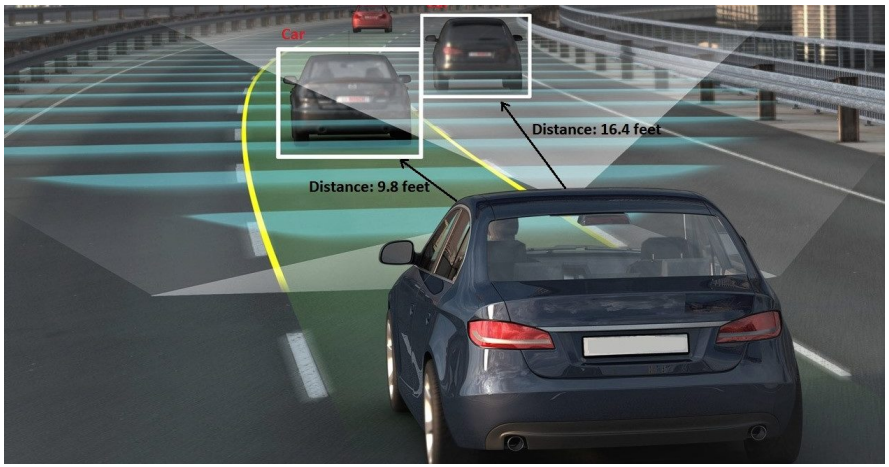


Figure: Image taken from https://icarus.csd.auth.gr

Overview

Previous work

3D Object Detection & Depth Estimation

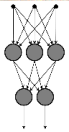
- Monocular/ Stereo based
- LiDAR based
  - Projection: MVCNN [ICCV, 2015], Complex-YOLO [CoRR, 2018]
  - Volumetric: VoxelNet [CoRR, 2017], Vote3deep [ICRA, 2017]
  - PointNet: PointNet [CVPR, 2017], PointNet++ [NIPS, 2017]
  - ...
- Fusing LiDAR & Camera
  - Early fusion
  - Late fusion
  - deep fusion, multi fusion

Pseudo LiDAR [CVPR, 2019] pipeline

Input

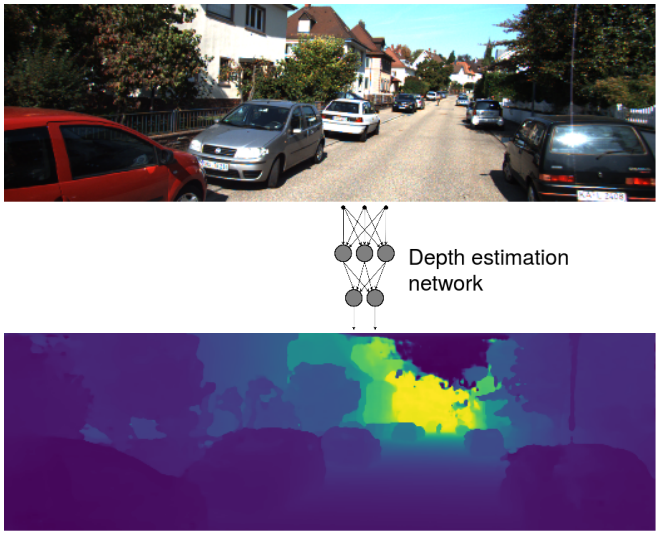
Pseudo LiDAR [CVPR, 2019] pipeline

Input



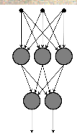
Depth estimation network

Pseudo LiDAR [CVPR, 2019] pipeline

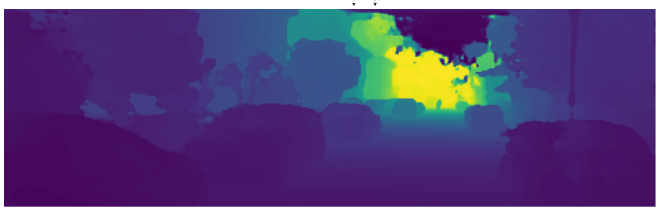


Pseudo LiDAR [CVPR, 2019] pipeline

Input



Depth estimation network

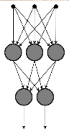


Depth map

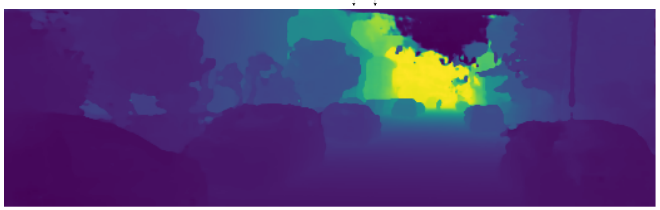
Converting

Pseudo LiDAR [CVPR, 2019] pipeline

Input

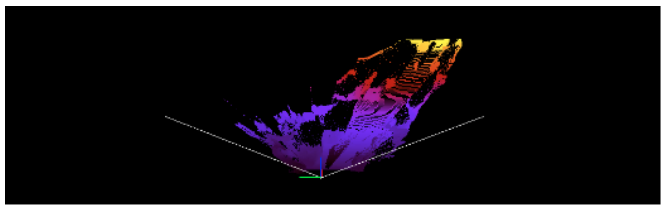


Depth estimation network



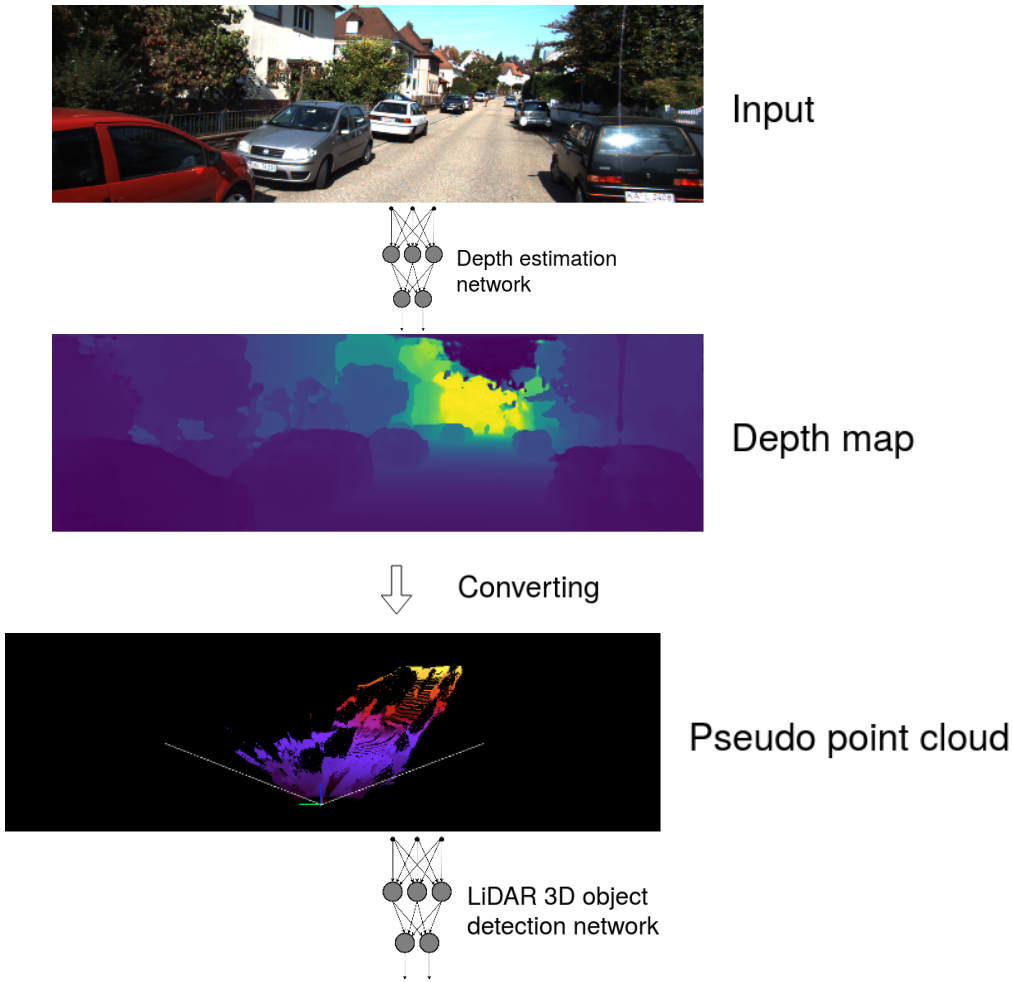
Depth map

Converting

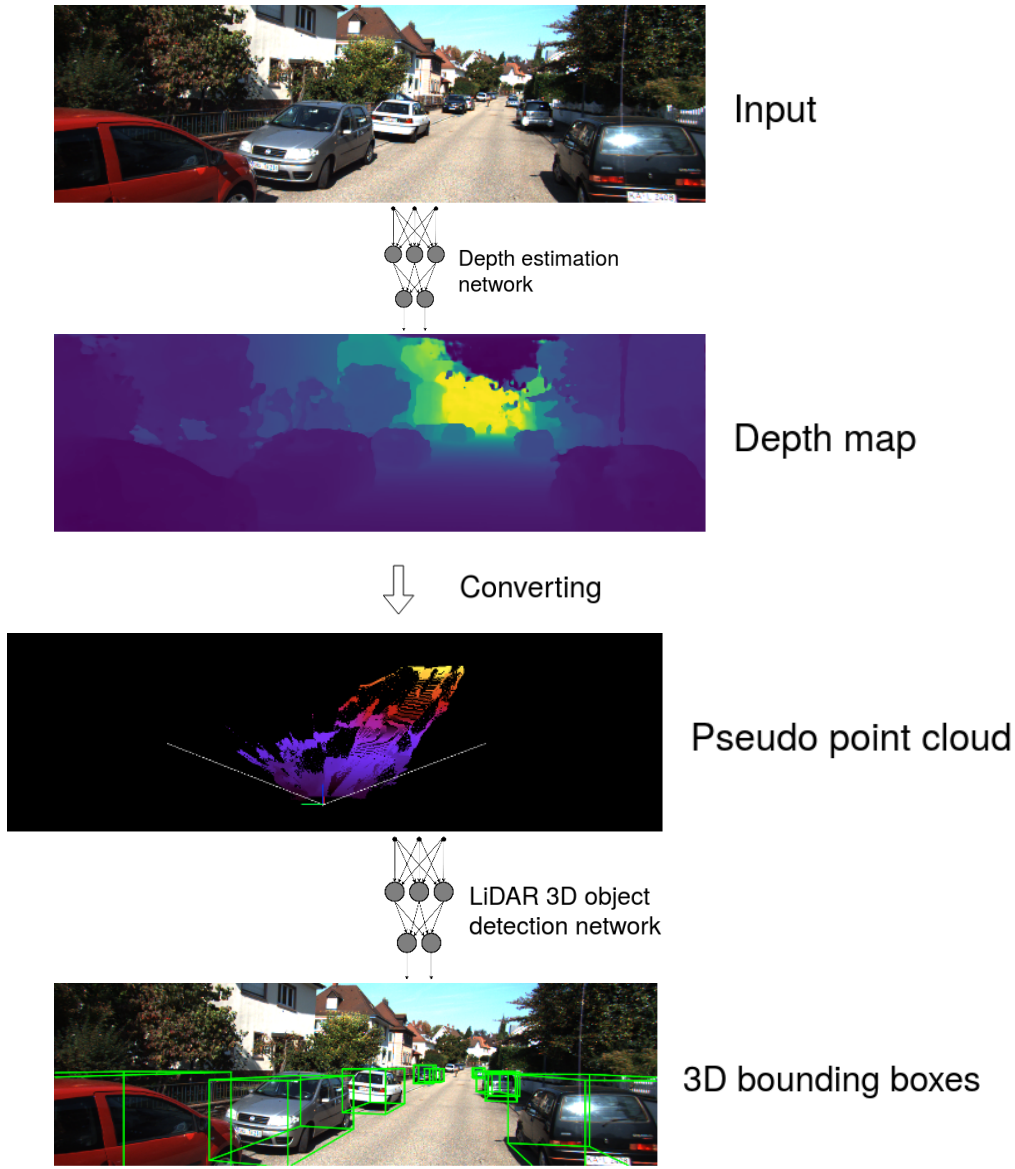


Pseudo point cloud

Pseudo LiDAR [CVPR, 2019] pipeline



Pseudo LiDAR [CVPR, 2019] pipeline

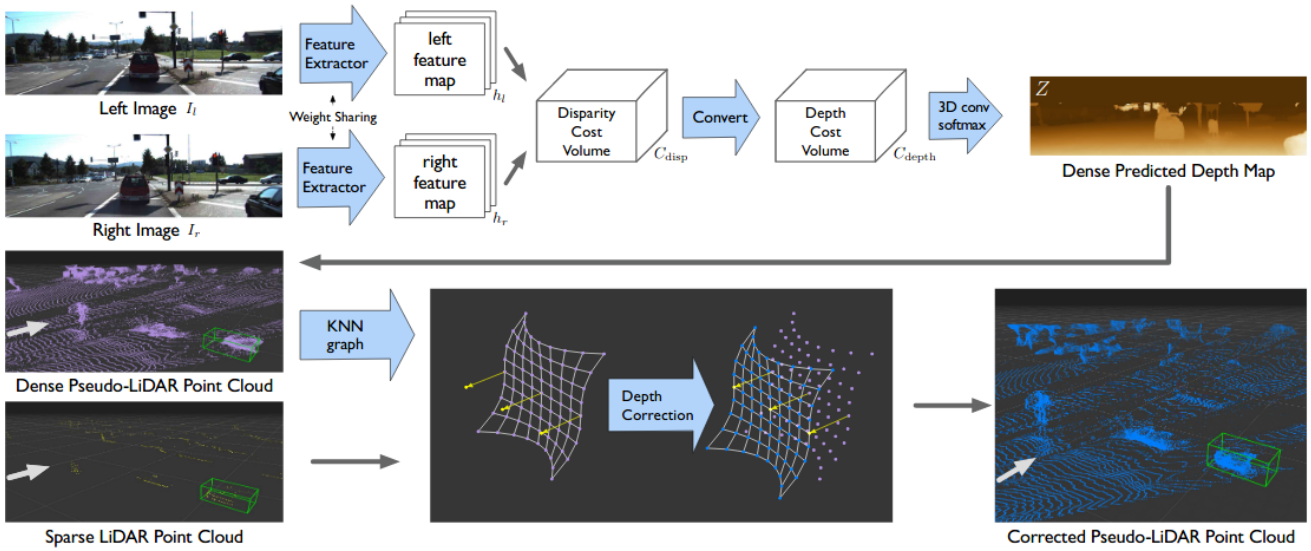


Research Question

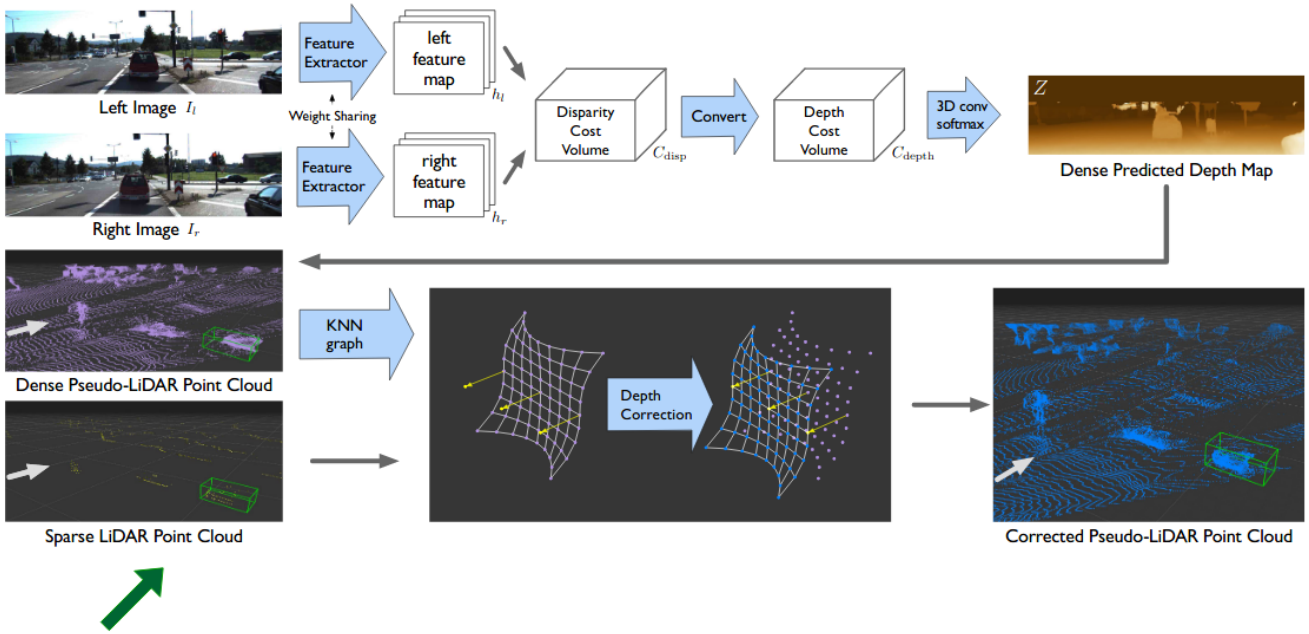
Can we design an architecture to fuse **cheap system: stereo + LiDAR 4 beams** to achieve:

- **Better performance than monocular/ stereo**
- **Close to performance of LiDAR based method???**

Pseudo LiDAR ++ [ICLR, 2020]

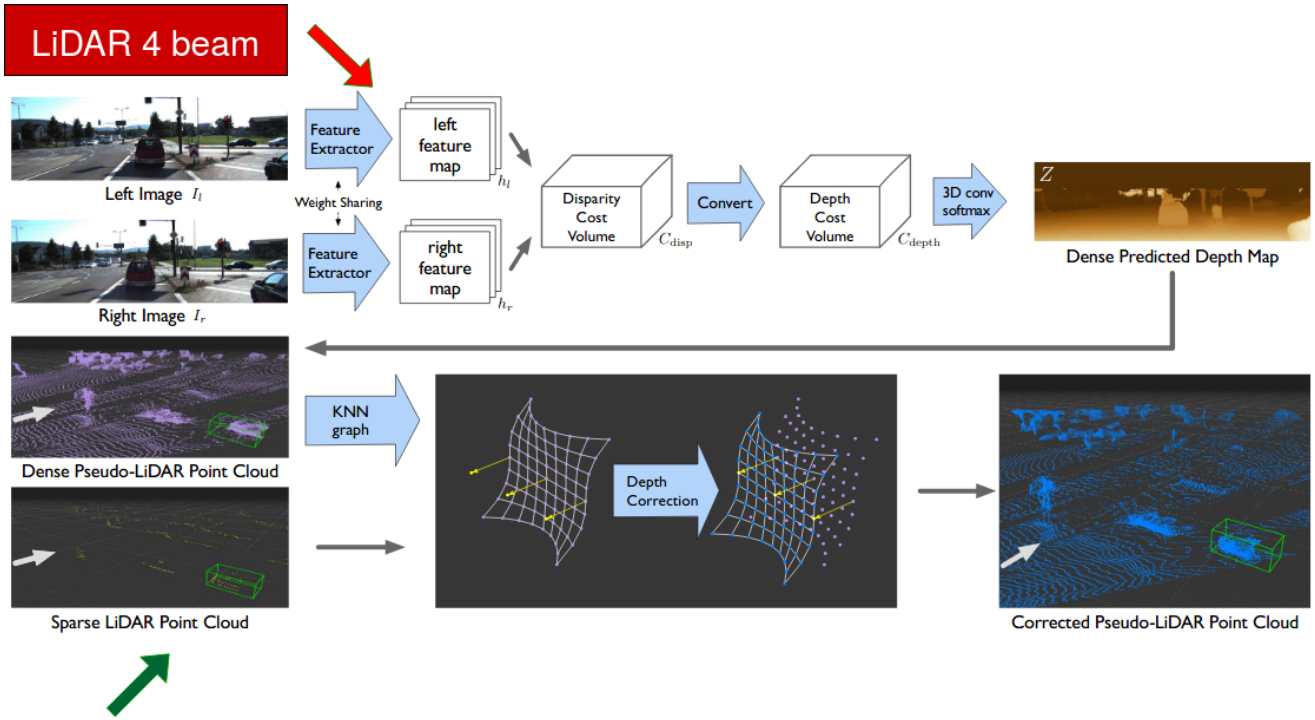


Pseudo LiDAR ++ [ICLR, 2020]



Pseudo LiDAR ++ [ICLR, 2020]

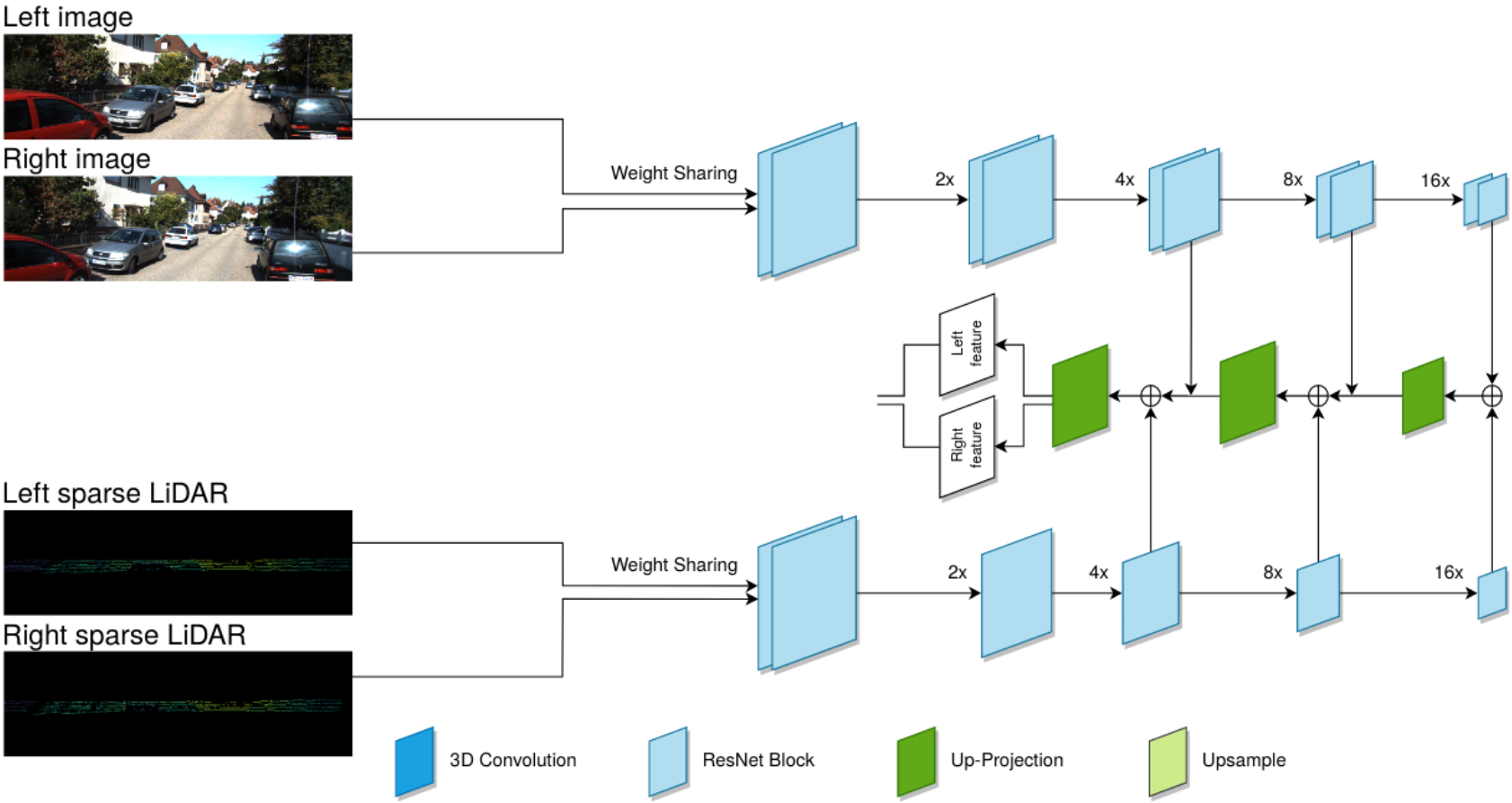
Our new one is here!



Overview

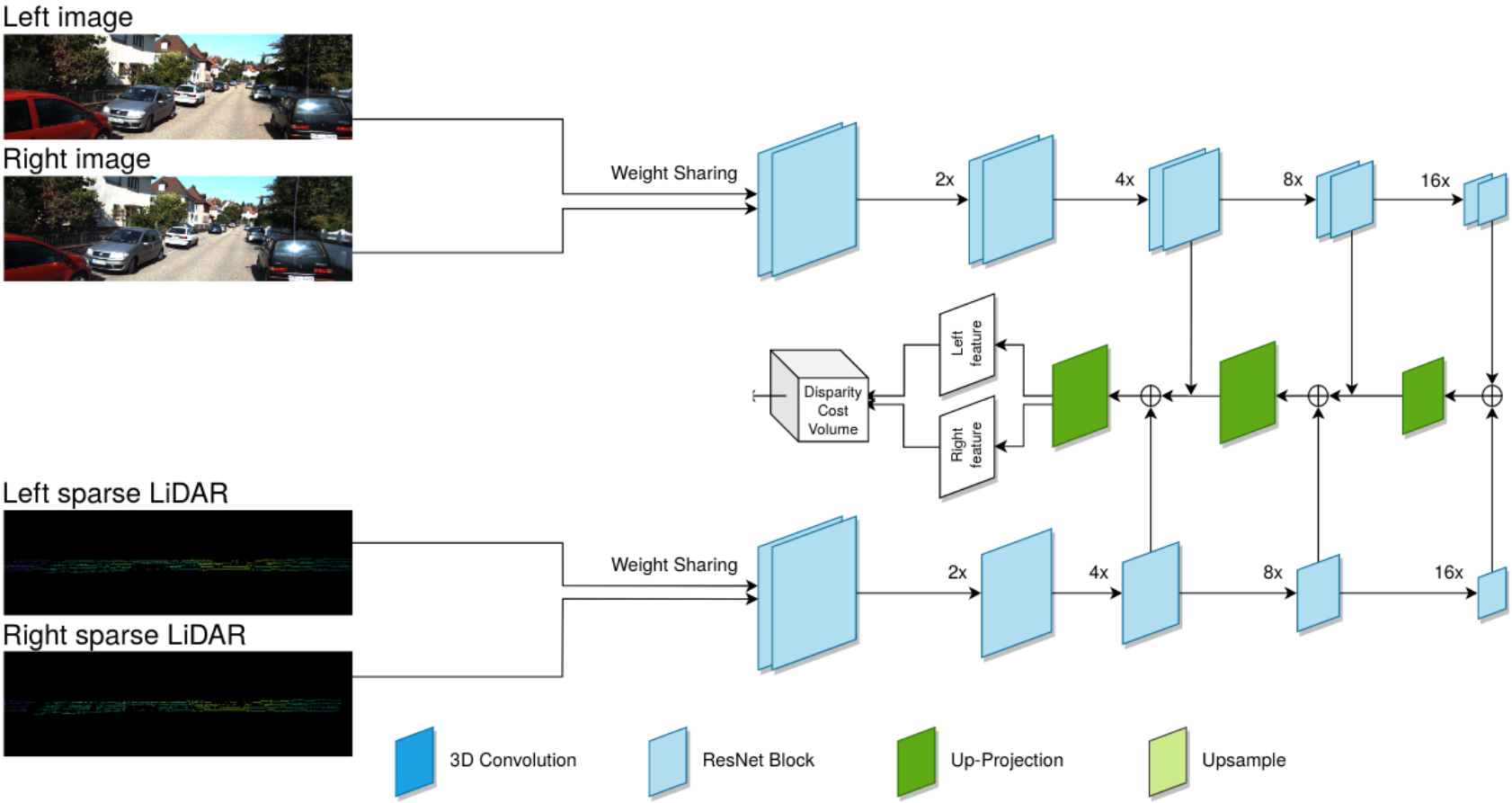
Method

Stereo Disparity Estimation [CVPR, 2018], concat LiDAR as 4th channel, multi-channel encoder-decoder DeepLiDAR [CVPR, 2019].



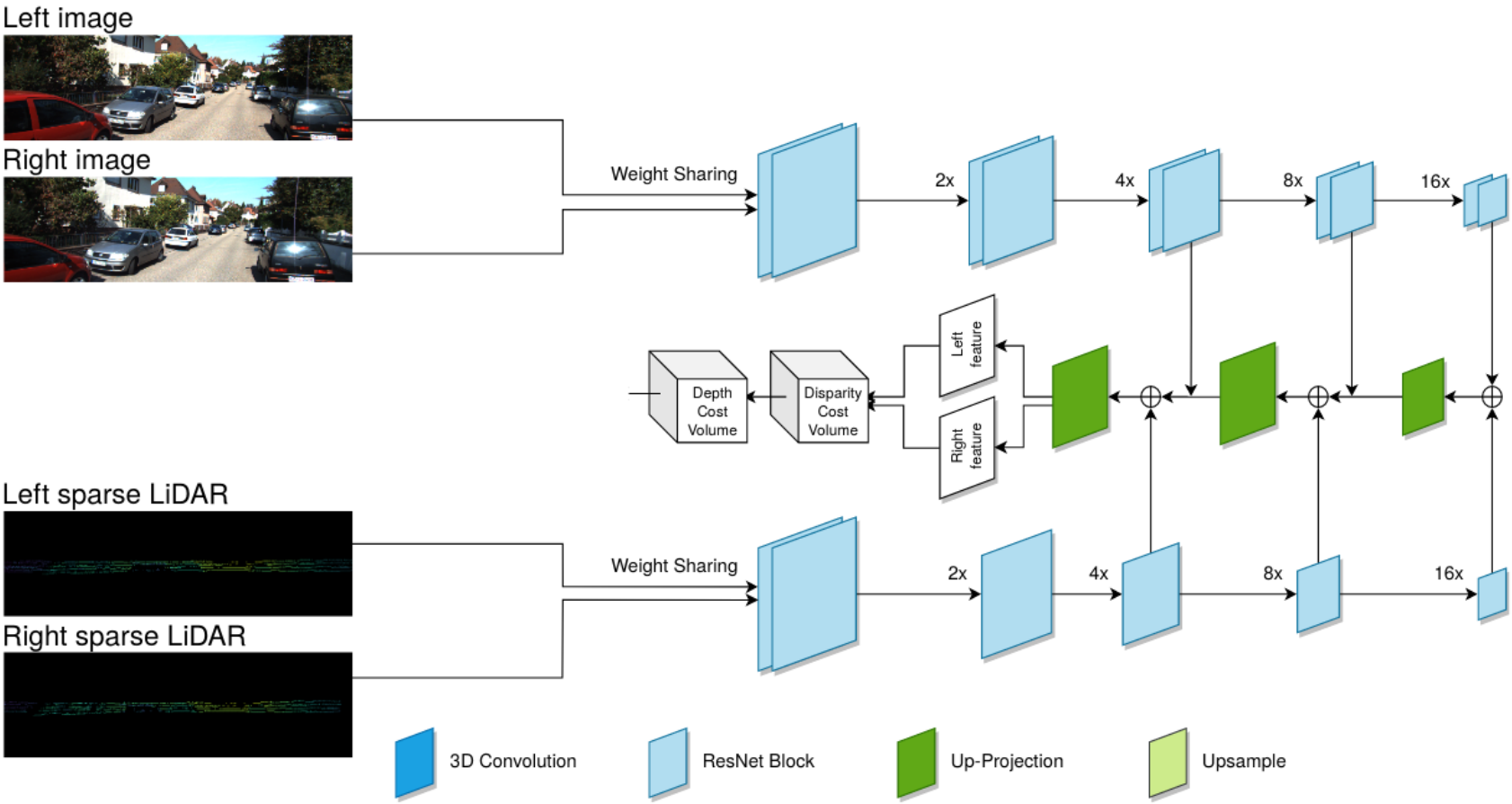
Method

Slide feature map from both sides and put into Disparity Cost Volume [CVPR, 2018]



Method

Depth Cost Volume [ICLR, 2020] (depth Z $\sim$ 1/disparity D), to directly optimize depth



Method

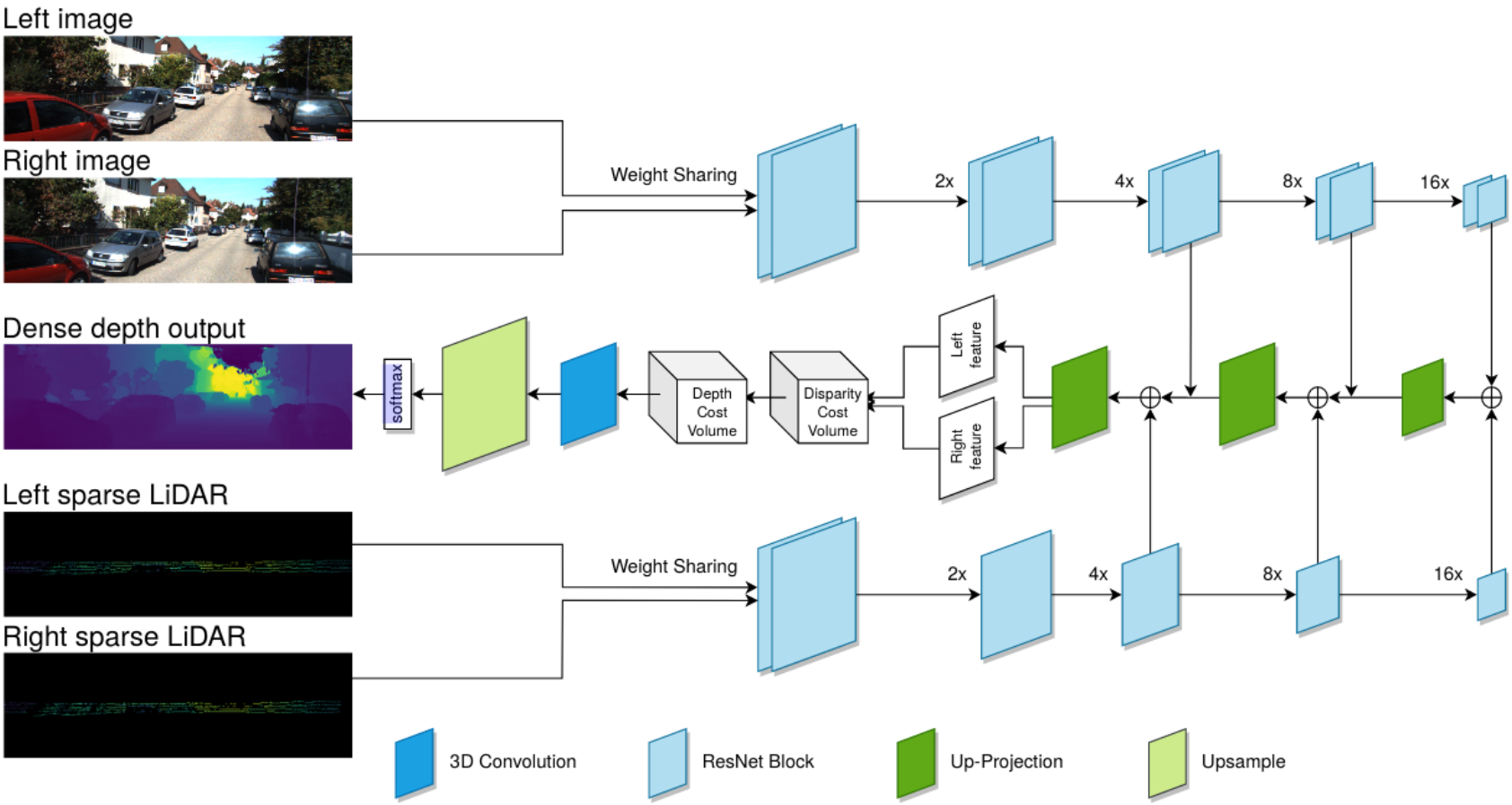


Figure: Our proposed network: SLS-Fusion

Overview

Results

Experiments

KITTI dataset [CVPR, 2012]

- 1 velodyne HDL-64E
- 2 grayscale, 2 color cameras (stereo)
- 1 (GPS/IMU)
- good weather
- 3D object detection: 7481 training & 7518 test samples

Training

- Depth estimation step: SceneFlow [CVPR, 2016], KITTI object detection
- 3D object detection step: PointRCNN [CVPR, 2019], KITTI object detection

Results

Results on Depth Estimation

Method	RMSE (mm)	MAE (mm)	iRMSE (1/km)	iMAE (1/km)
Baseline∗ [5]	1506.92	443.81	5.59	1.90
Ours∗	**845.21**	**307.36**	**2.72**	**1.28**

Figure: Dataset: KITTI Object Detection [CVPR, 2012], 3712 samples for training & 3769 samples (validation) for evaluating
Baseline∗: Pseudo-LiDAR++ [ICLR, 2020]
Ours∗: SLS-Fusion Network
Metric: RMSE, MAE, iRMSE, iMAE

Results

Results on 3D Object Detection

Method	Input	0.5 IoU			0.7 IoU		
		Easy	Moderate	Hard	Easy	Moderate	Hard
TLNet [8]	S	62.46 / 59.51	45.99 / 43.71	41.92 / 37.99	29.22 / 18.15	21.88 / 14.26	18.83 / 13.72
Stereo-RCNN [7]	S	87.13 / 85.84	74.11 / 66.28	58.93 / 57.24	68.50 / 54.11	48.30 / 36.69	41.47 / 31.07
Pseudo LiDAR [6]	S	88.4 / 88.0	76.6 / 73.7	69.0 / 67.8	73.4 / 62.3	56.0 / 44.9	52.7 / 41.6
DSGN [9]	S	-	-	-	83.24 / 73.2	63.91 / 54.27	57.83 / 47.71
CG-Stereo [10]	S	**97.04** / 90.58	88.58/ **87.01**	80.34/ 79.76	87.31 / 76.17	68.69 / 57.82	65.80 / 54.63
Baseline∗ [5]	S	89.8 / 89.7	83.8 / 78.6	77.5 / 75.1	82.0 / 67.9	64.0 / 50.1	57.3 / 45.3
Ours∗	L4+S	**90.13 / 89.9**	**83.89 / 78.81**	**78.03 / 76.14**	**82.38 / 68.08**	**65.42 / 50.81**	**57.81 / 46.07**
Baseline∗∗ [5]	L4+S	90.3 / 90.3	87.7 / **86.9**	**84.6 / 84.2**	**88.2** / 75.1	**76.9** / 63.8	73.4 / **57.4**
Ours∗∗	L4+S	**93.16 / 93.02**	**88.81** / 86.19	83.35 / 84.02	87.51 / **76.67**	76.88 / **63.90**	**73.55** / 56.78
PointRCNN [3]	L64	97.3 / 97.3	89.9 / 89.8	89.4 / 89.3	90.2 / 89.2	87.9 / 78.9	85.5 / 77.9

Figure: Dataset: KITTI Object Detection [CVPR, 2012], 3712 samples for training & 3769 samples (validation) for evaluating
Baseline: Pseudo-LiDAR++ [ICLR, 2020]
Ours: SLS-Fusion Network
Metric: AP BEV/ AP 3D object detection

Results

Results on 3D Object Detection

Method	Input	0.5 IoU			0.7 IoU		
		Easy	Moderate	Hard	Easy	**Moderate**	Hard
TLNet [8]	S	62.46 / 59.51	45.99 / 43.71	41.92 / 37.99	29.22 / 18.15	21.88 / 14.26	18.83 / 13.72
Stereo-RCNN [7]	S	87.13 / 85.84	74.11 / 66.28	58.93 / 57.24	68.50 / 54.11	48.30 / 36.69	41.47 / 31.07
Pseudo LiDAR [6]	S	88.4 / 88.0	76.6 / 73.7	69.0 / 67.8	73.4 / 62.3	56.0 / 44.9	52.7 / 41.6
DSGN [9]	S	-	-	-	83.24 / 73.2	63.91 / 54.27	57.83 / 47.71
CG-Stereo [10]	S	**97.04** / 90.58	88.58/ **87.01**	80.34/ 79.76	87.31 / 76.17	68.69 / 57.82	65.80 / 54.63
Baseline∗ [5]	S	89.8 / 89.7	83.8 / 78.6	77.5 / 75.1	82.0 / 67.9	64.0 / 50.1	57.3 / 45.3
Ours∗	L4+S	**90.13 / 89.9**	**83.89 / 78.81**	**78.03 / 76.14**	**82.38 / 68.08**	**65.42 / 50.81**	**57.81 / 46.07**
Baseline∗∗ [5]	L4+S	90.3 / 90.3	87.7 / **86.9**	**84.6 / 84.2**	**88.2** / 75.1	**76.9** / 63.8	73.4 / **57.4**
Ours∗∗	L4+S	**93.16 / 93.02**	**88.81** / 86.19	83.35 / 84.02	87.51 / **76.67**	76.88 / **63.90**	**73.55** / 56.78
PointRCNN [3]	L64	97.3 / 97.3	89.9 / 89.8	89.4 / 89.3	90.2 / 89.2	87.9 / 78.9	85.5 / 77.9

Figure: Dataset: KITTI Object Detection [CVPR, 2012], 3712 samples for training & 3769 samples (validation) for evaluating
Baseline: Pseudo-LiDAR++ [ICLR, 2020]
Ours: SLS-Fusion Network
Metric: AP BEV/ AP 3D object detection

Results

Results on 3D Object Detection

Method	Input	0.5 IoU			0.7 IoU		
		Easy	Moderate	Hard	Easy	**Moderate**	Hard
TLNet [8]	S	62.46 / 59.51	45.99 / 43.71	41.92 / 37.99	29.22 / 18.15	21.88 / 14.26	18.83 / 13.72
Stereo-RCNN [7]	S	87.13 / 85.84	74.11 / 66.28	58.93 / 57.24	68.50 / 54.11	48.30 / 36.69	41.47 / 31.07
Pseudo LiDAR [6]	S	88.4 / 88.0	76.6 / 73.7	69.0 / 67.8	73.4 / 62.3	56.0 / 44.9	52.7 / 41.6
DSGN [9]	S	-	-	-	83.24 / 73.2	63.91 / 54.27	57.83 / 47.71
CG-Stereo [10]	S	**97.04** / 90.58	88.58/ **87.01**	80.34/ 79.76	87.31 / 76.17	68.69 / 57.82	65.80 / 54.63
Baseline∗ [5]	S	89.8 / 89.7	83.8 / 78.6	77.5 / 75.1	82.0 / 67.9	64.0 / 50.1	57.3 / 45.3
Ours∗	L4+S	**90.13 / 89.9**	**83.89 / 78.81**	**78.03 / 76.14**	**82.38 / 68.08**	**65.42 / 50.81**	**57.81 / 46.07**
Baseline∗∗ [5]	L4+S	90.3 / 90.3	87.7 / **86.9**	**84.6 / 84.2**	**88.2** / 75.1	**76.9** / 63.8	73.4 / **57.4**
Ours∗∗	L4+S	**93.16 / 93.02**	**88.81** / 86.19	83.35 / 84.02	87.51 / **76.67**	76.88 / **63.90**	**73.55** / 56.78
PointRCNN [3]	L64	97.3 / 97.3	89.9 / 89.8	89.4 / 89.3	90.2 / 89.2	87.9 / 78.9	85.5 / 77.9

Figure: Dataset: KITTI Object Detection [CVPR, 2012], 3712 samples for training & 3769 samples (validation) for evaluating
Baseline: Pseudo-LiDAR++ [ICLR, 2020]
Ours: SLS-Fusion Network
Metric: AP BEV/ AP 3D object detection

Results

Visual results on Kitti (ground truth/ prediction: red/ green 3D bbox)

Overview

Summary

- A novel network by fusing stereo & LiDAR 4 beams (cheap sensors system) for depth estimation and also 3D object detection
- Accurate and cheap
- Release Code: https://github.com/maiminh1996/SLS-Fusion

Overview

Future Work

- How to efficiently use pseudo point cloud (very dense 4x LiDAR 64 beams)

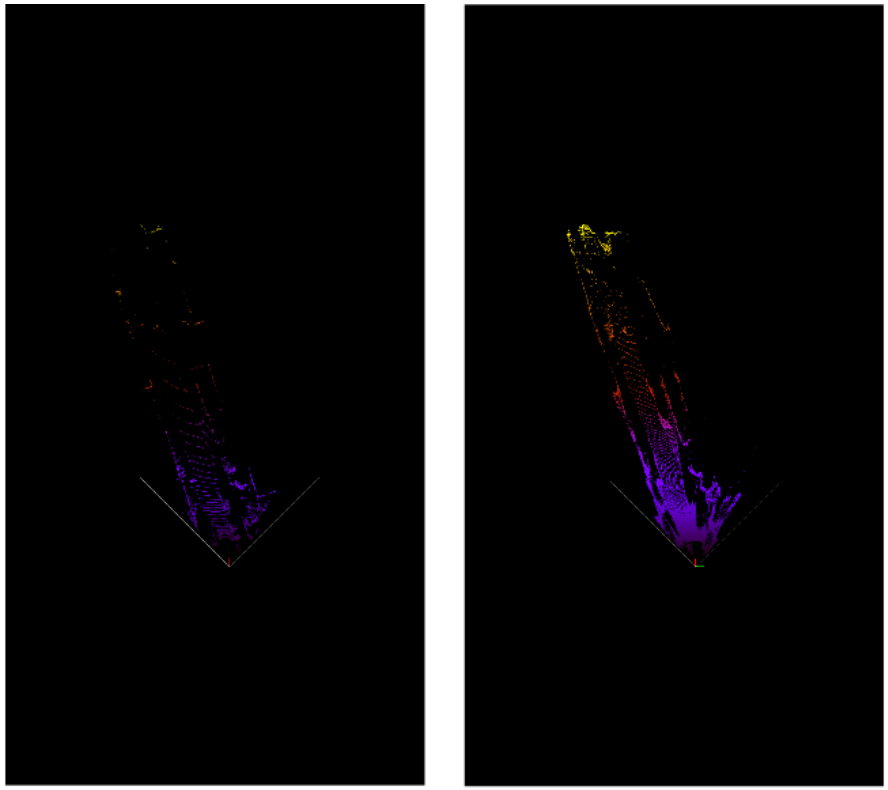


Figure: LiDAR 64 beams (left) and Pseudo point cloud generated by our SLS-Fusion method

- Sensors degraded on adverse weather

Thank you

References I

CVPR (2019)

Pseudo-lidar from visual depth estimation: Bridging the gap in 3d object detection for autonomous driving

Computer Vision and Pattern Recognition (CVPR)

CVPR (2019)

Learning the Depths of Moving People by Watching Frozen People

Computer Vision and Pattern Recognition (CVPR)

ICLR (2020)

Pseudo-lidar++: Accurate depth for 3d object detection in autonomous driving

International Conference on Learning Representations (ICLR)

CVPR (2018)

Pyramid stereo matching network

Computer Vision and Pattern Recognition (CVPR

References II

CVPR (2012)

Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite

Computer Vision and Pattern Recognition (CVPR)

CVPR (2019)

Pointrcnn: 3d object proposal generation and detection from point cloud

Computer Vision and Pattern Recognition (CVPR)

CVPR (2016)

A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation

Computer Vision and Pattern Recognition (CVPR)

CVPR (2019)

Deep surface normal guided depth prediction for outdoor scene from sparse lidar data and single color image

Computer Vision and Pattern Recognition (CVPR)

References III

ICCV (2015)

Multi-view convolutional neural networks for 3d shape recognition

International Conference on Computer Vision (ICCV)

CoRR (2018)

Complex-YOLO: Real-time 3D Object Detection on Point Clouds

Computing Research Repository (CoRR)

ICRA (2017)

Vote3deep: Fast object detection in 3d point clouds using efficient convolutional neural networks

Conferenceon Robotics and Automation (ICRA)

CoRR (2017)

VoxelNet: End-to-End Learning for PointCloud Based 3D Object Detection

Computing Research Repository (CoRR)

References IV

CVPR (2017)

PointNet: DeepLearning on Point Sets for 3d Classification and Segmentation

Computer Vision and Pattern Recognition (CVPR)

NIPS (2017)

PointNet++: Deep HierarchicalFeature Learning on Point Sets in a Metric Space

Neural Information Processing Systems