Multi-View 3D Object Detection Network for Autonomous Driving

Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, Tian Xia

CVPR 2017 (Spotlight)

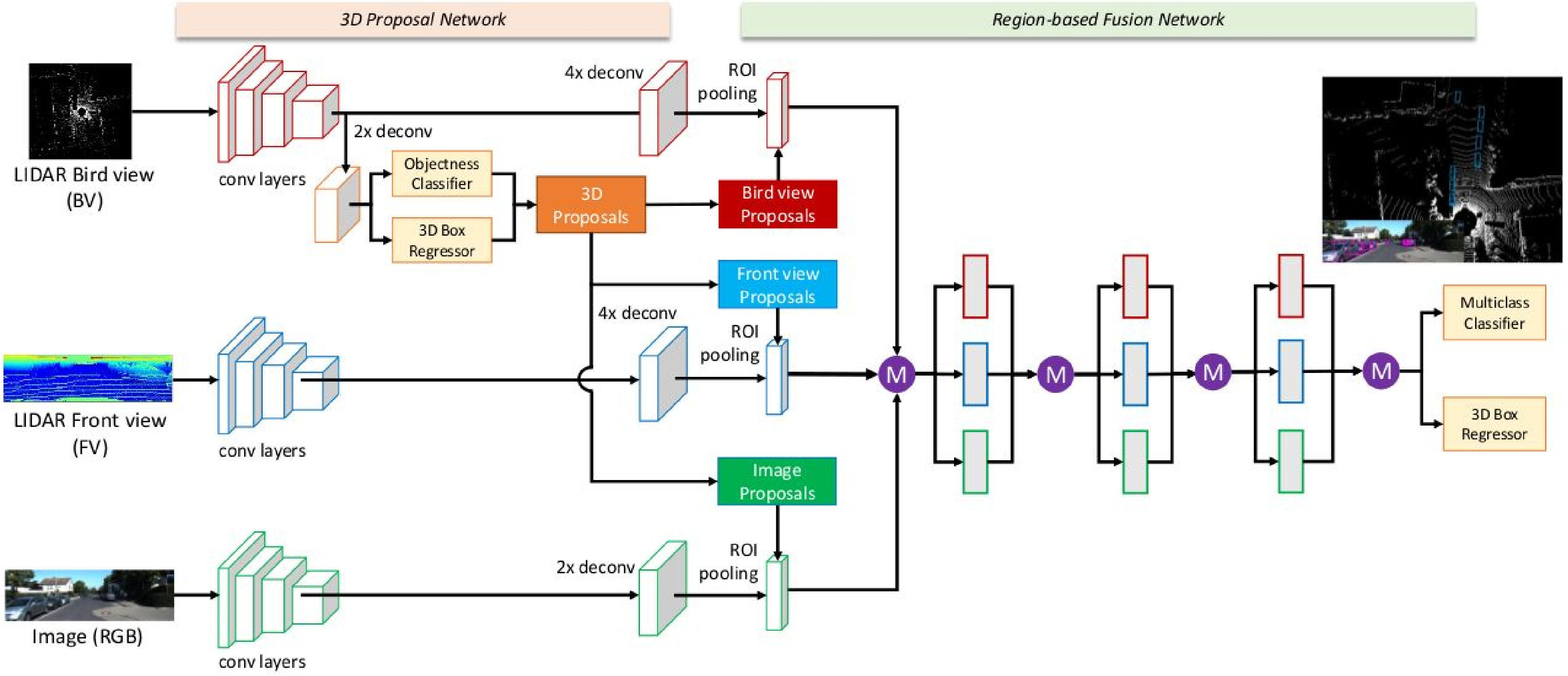
Presented By: Jason Ku

Overview

- Motivation
- Dataset
- Network Architecture
- Multi-View Inputs
- 3D Proposal Network
- Multi-View ROI Pooling
- Fusion Network
- Network Regularization
- Training
- Results
- Summary
- Improvements

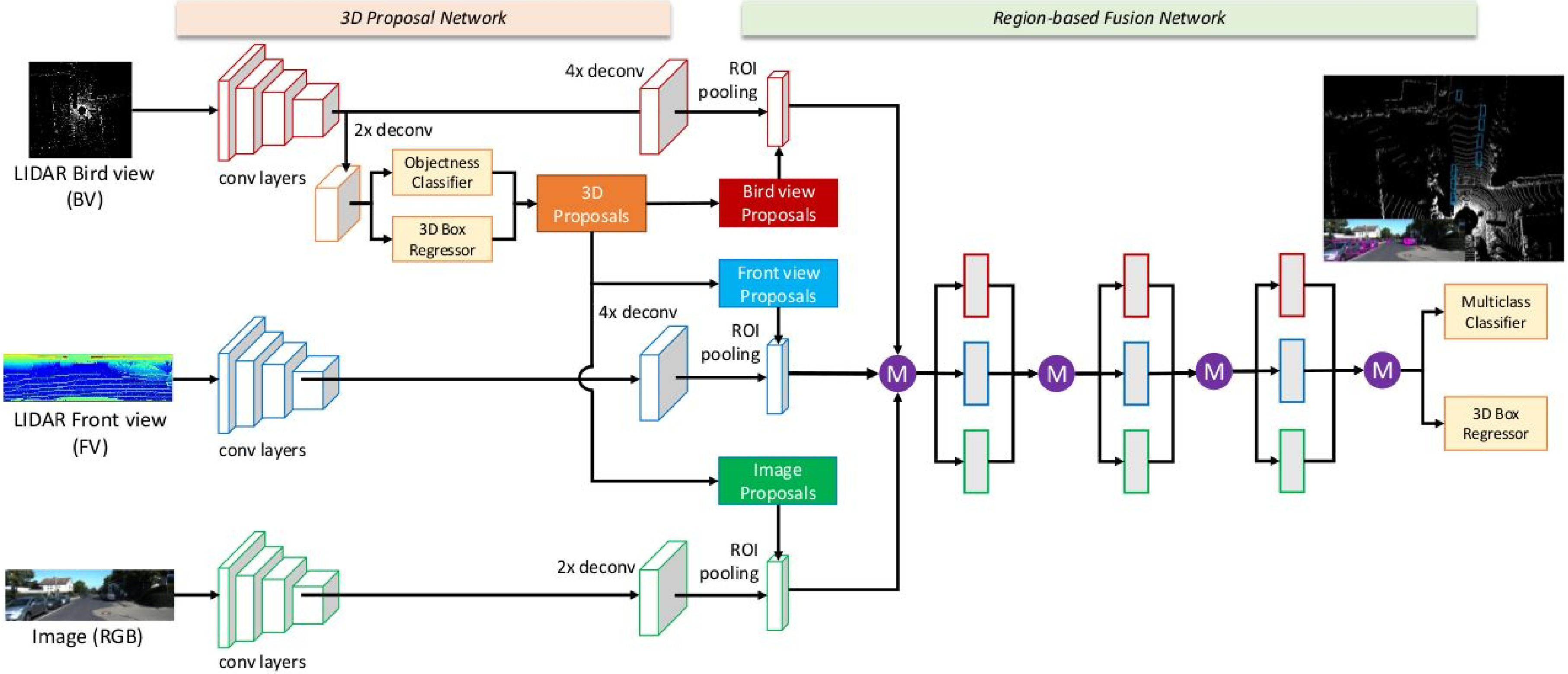
Motivation

- 3D detections are much more useful for autonomous driving
- LIDAR data contains accurate depth information
- Requires well designed model to take advantage of multiple views
- 2 stage architectures provide higher quality bounding boxes

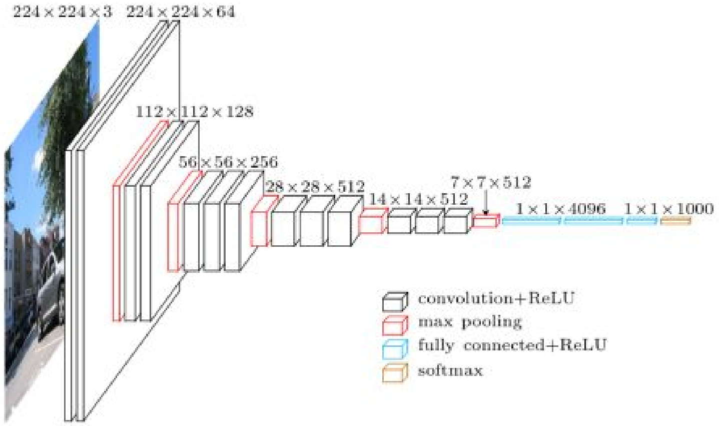


Dataset

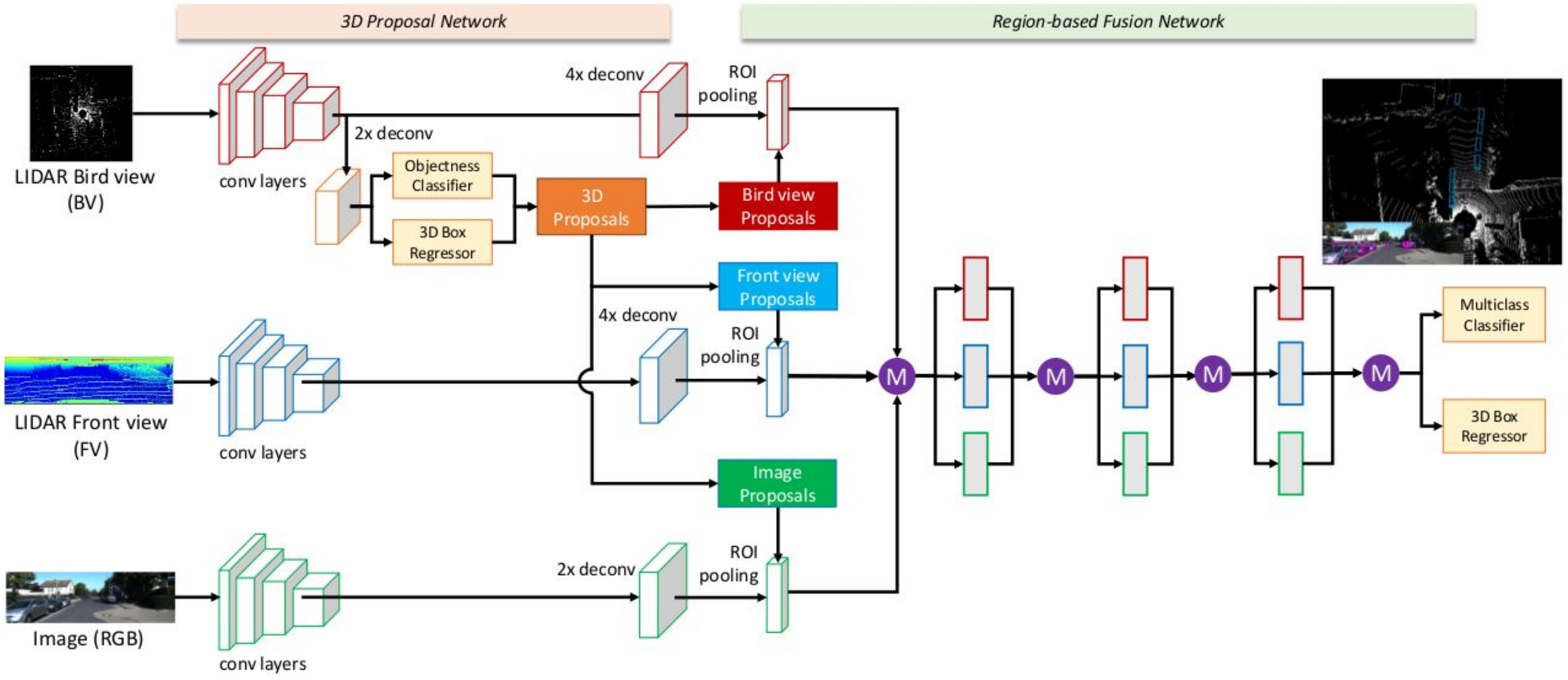
- KITTI images, only car instances
- Half split of training images for training and validation set (~3700 each)
- KITTI test server only evaluates 2D detections
- 3D detections evaluated on validation set



Network Architecture

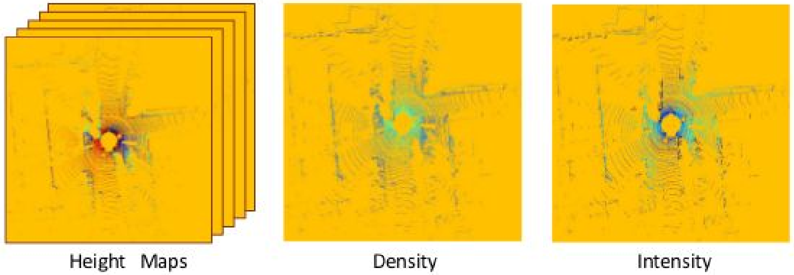


- VGG-16 base network for each view
  - Channels reduced by half
  - 4th max pooling operation is removed
  - Extra fully connected layer fc8
  - Weights initialized by sampling weights from VGG-16
- Even with 3 branches, only 75% number of parameters as full VGG-16

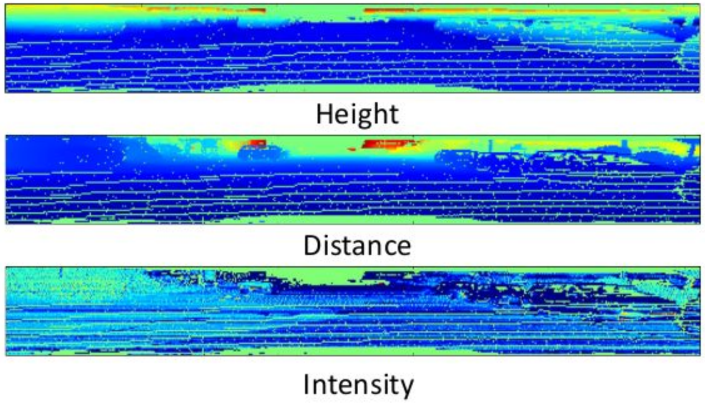


Multi-View Inputs

- Bird’s Eye View (BV)



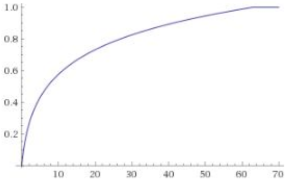
- Front View (FV)

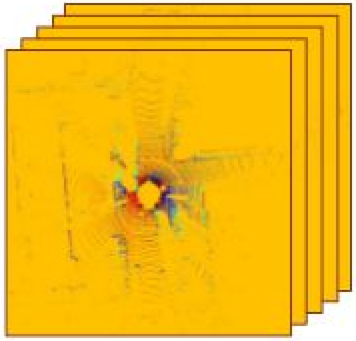


- Camera Image (RGB)
    - Upscaled to make shortest length 500 pixels

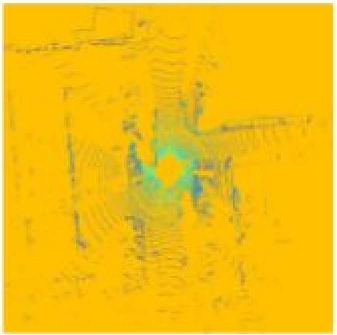
Bird’s-Eye View (BV)

- Discretized LIDAR point cloud with 0.1m resolution
- ~90° FOV images
- Range of [0, 70.4] (depth) x [-40, 40]
- 704 x 800 pixels
- Features in (M + 2) channels:
  - M Height Maps
    - Point cloud divided into M equal slices
    - Maximum height of points in cell
    - Probably to address tunnels, bridges, or trees
  - Density
    - Number of points in each cell
    - Normalized as $min(1.0, \frac{log(N+1)}{log(64)})$
    - N is the number of points in the cell
  - Intensity
    - LIDAR reflectance of point with maximum height in cell

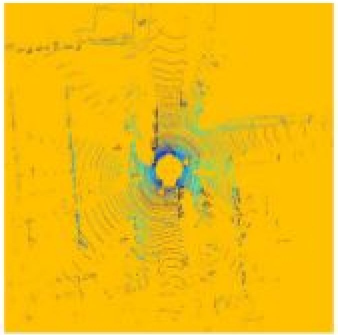




Height Maps



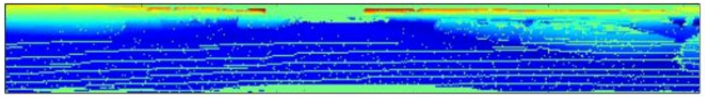
Density



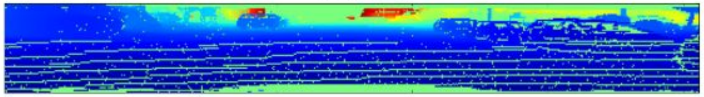
Intensity

Front View (FV)

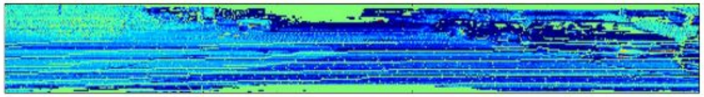
- Projects LIDAR point cloud to cylinder plane
- Denser map than projection to 2D point map
- Height, Distance, Intensity
- 64 beam Velodyne => 64 x 512 pixels



Height



Distance



Intensity

LIDAR point cloud to cylinder plane conversion:

Given a 3D point $p = (x, y, z)$, its coordinates in the front view $p_{fv} = (r, c)$

can be computed using

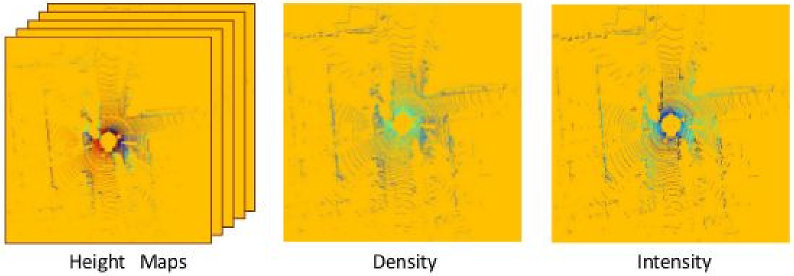
$$c = \lfloor \mathrm{atan2}(y, x)/\Delta\theta]\rfloor$$

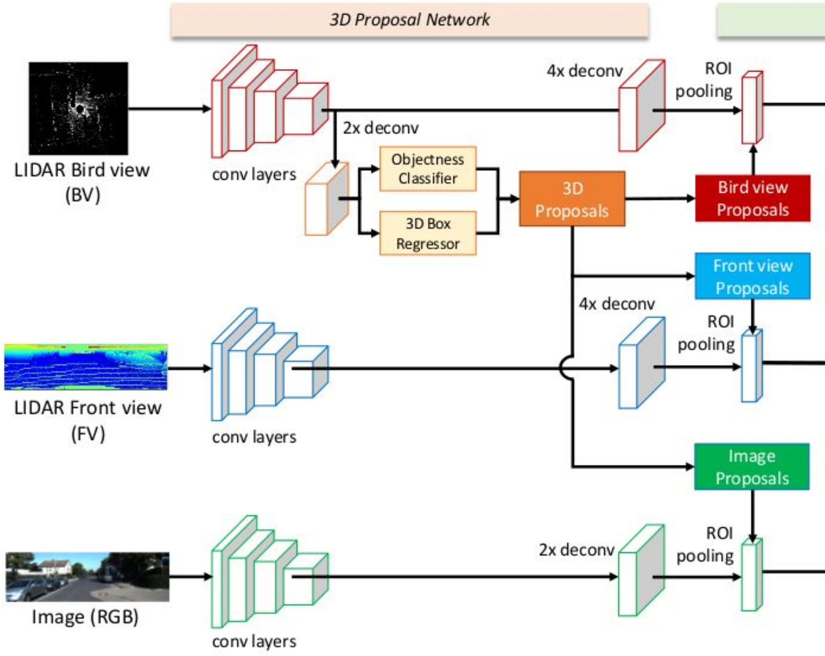
$$r = \lfloor \mathrm{atan2}(z, \sqrt{x^2 + y^2})/\Delta\phi\rfloor$$

where $\Delta\theta$ and $\Delta\phi$ are horizontal and vertical resolution of laser beams

3D Proposal Network

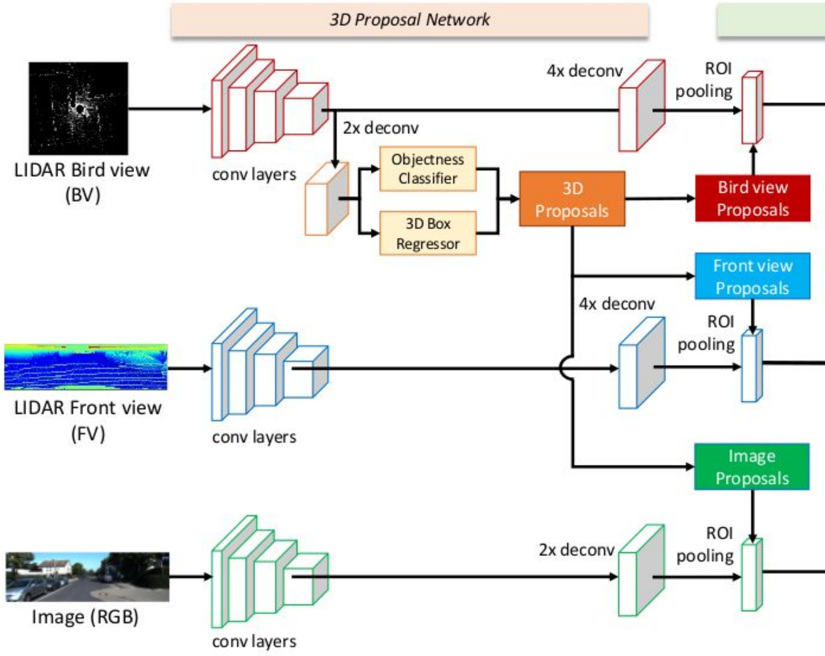
- Bird’s eye view input
  - Preserves physical size
  - Objects occupy different space
  - Provides better predictions since objects are grounded, and encodes depth information
- Feature upsampling for small objects
  - 2x bilinear feature map upsampling
  - Proposal network gets 4x downsampled input (176 x 200 px)

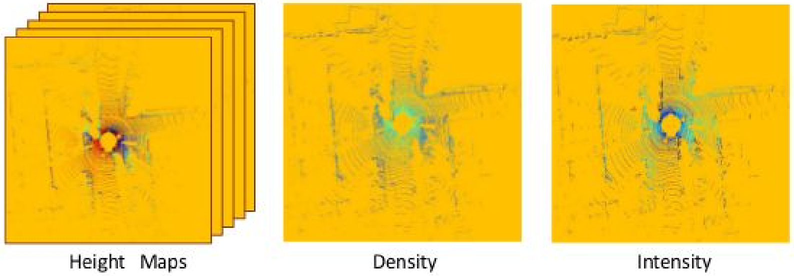




3D Proposal Network

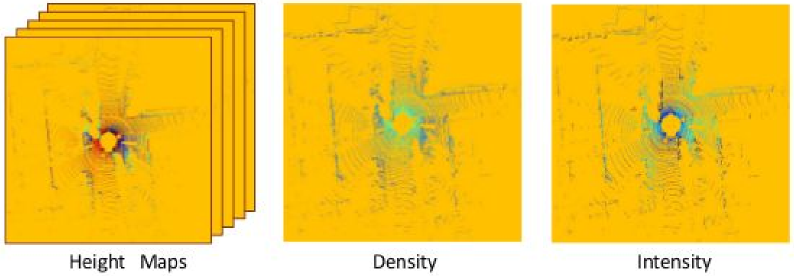
- 3D Anchors
  - 3D prior boxes created by clustering ground truth object sizes
  - Represented with center and sizes
  - (l, w) = {(3.9, 1.6), (1.0, 0.6)}, h = 1.56m
  - Orientations {0°, 90°}, not regressed
  - Close to orientations of most road scene objects
  - 4 boxes
- Proposal Filtering
  - Remove background and empty proposals
  - 0.7 IoU NMS in BV, based on objectness score
  - Top 2000 proposals for training
  - Top 300 for testing

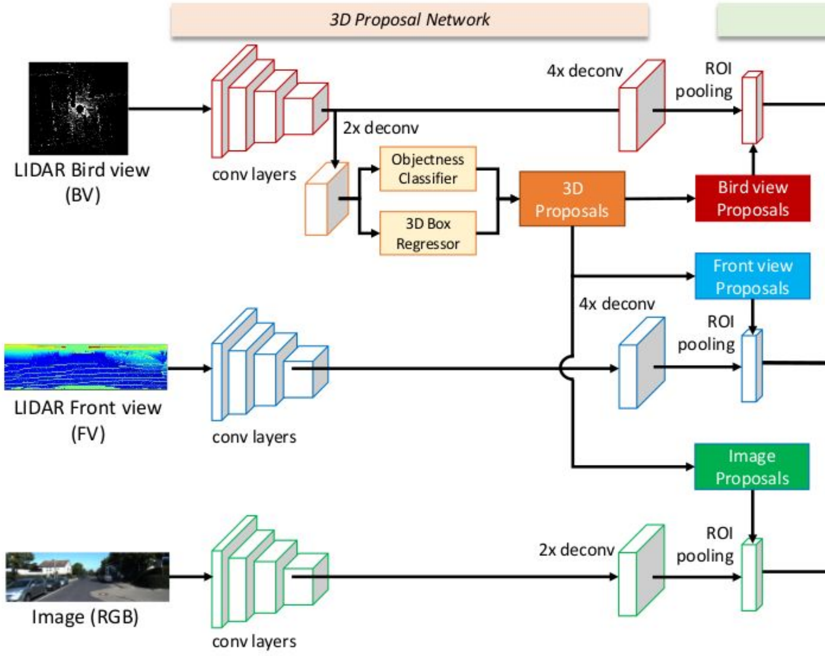




3D Proposal Bounding Box Regression

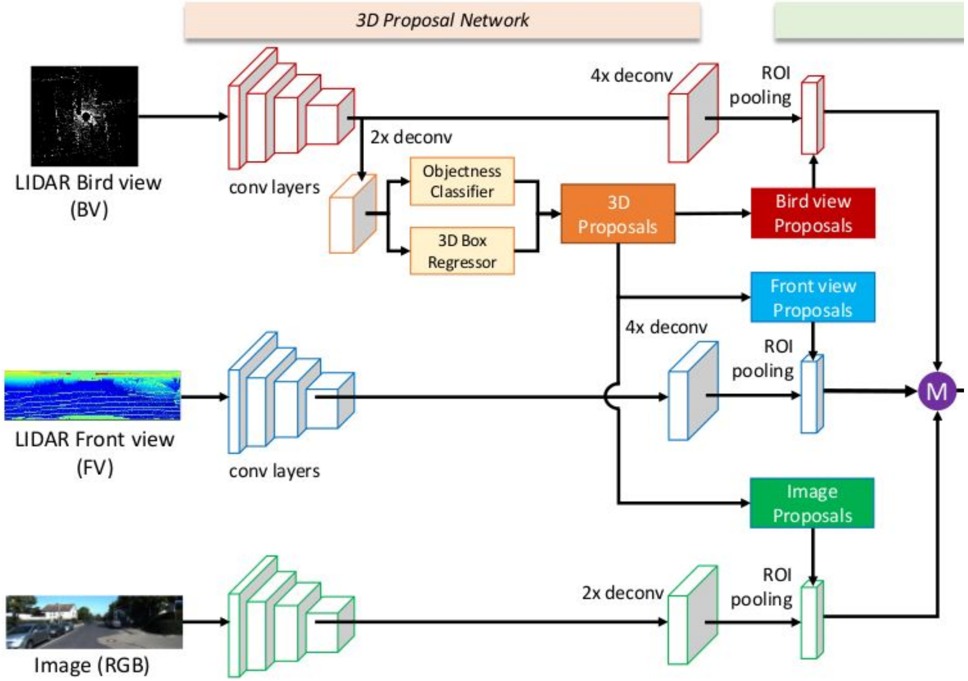
- Parameterized as t = (∆x, ∆y, ∆z, ∆l, ∆w, ∆h)
- (∆x, ∆y, ∆z) are the center offsets normalized by anchor size
- (∆l, ∆w, ∆h) are computed as $\Delta s = log\frac{s_{\text{GT}}}{s_{\text{anchor}}}, s \in \{l, w, h\}$
- Multi-task loss
  - Class-entropy (log loss) for objectness
  - Smooth L1 (distance) for 3D box regression





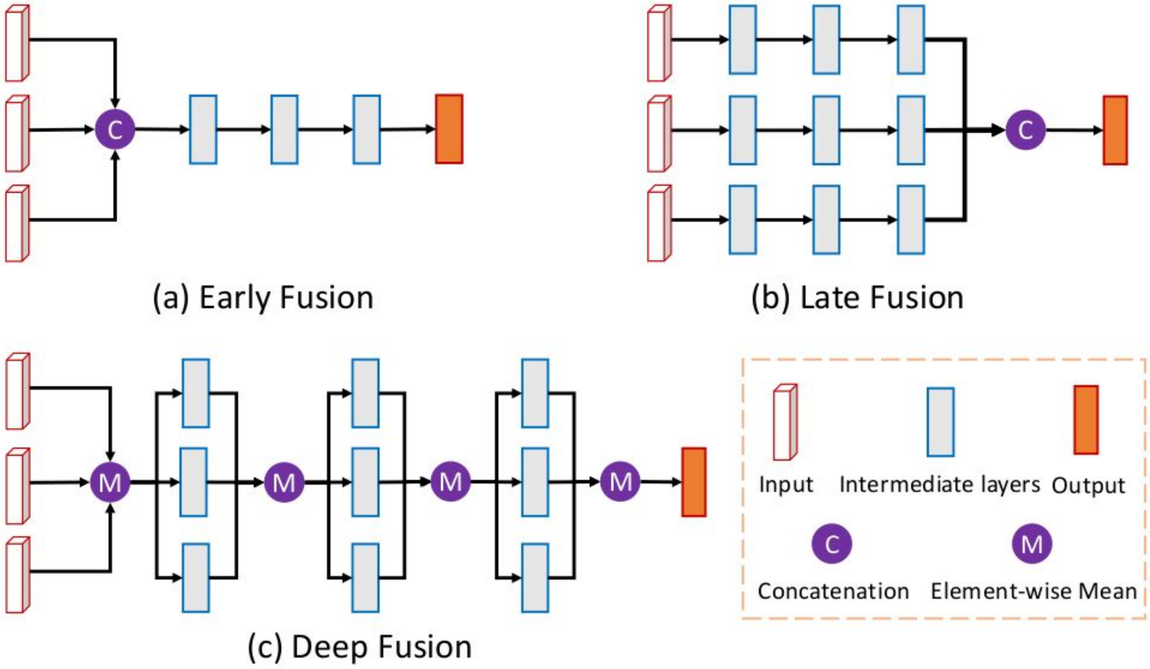
Multi-View ROI Pooling

- 3D box proposals projected into each view
- 4x/4x/2x upsampled feature vector
- Region of Interest (ROI) pooling to create same length feature vectors

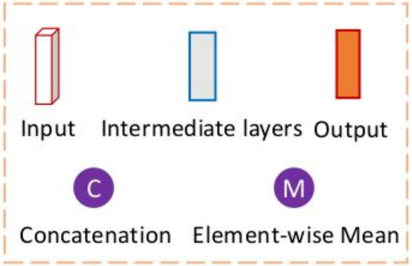


Fusion Network

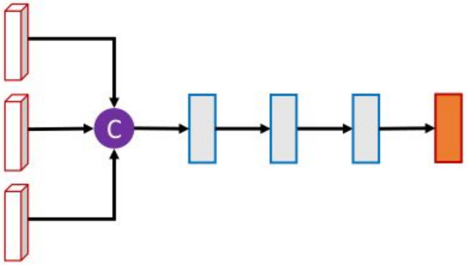
- Combines information from different feature vectors



Early and Late Fusion



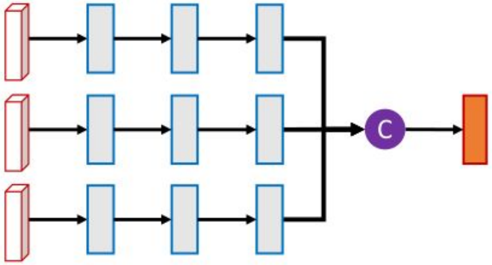
- Early Fusion
  - Features combined at the input stage
  - For L layers, $f_L = \mathbf{H}_L(\mathbf{H}_{L-1}(\cdots \mathbf{H}_1(f_{BV} \oplus f_{FV} \oplus f_{RGB})))$
  - Where $\{\mathbf{H}_l, l = 1, \cdots, L\}$ are feature transformation functions, and $\oplus$ is a join operation (concatenation)



(a) Early Fusion

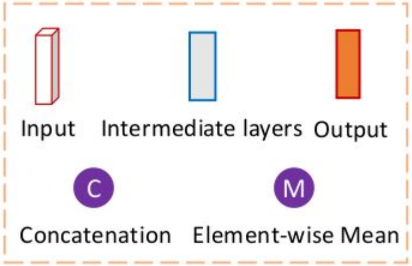
- Late Fusion
  - Separate subnetworks learn features independently
  - Output combined at the prediction stage

$$\begin{aligned} f_L = &(\mathbf{H}_L^{BV}(\cdots \mathbf{H}_1^{BV}(f_{BV}))) \oplus \\ &(\mathbf{H}_L^{FV}(\cdots \mathbf{H}_1^{FV}(f_{FV}))) \oplus \\ &(\mathbf{H}_L^{RGB}(\cdots \mathbf{H}_1^{RGB}(f_{RGB}))) \end{aligned}$$



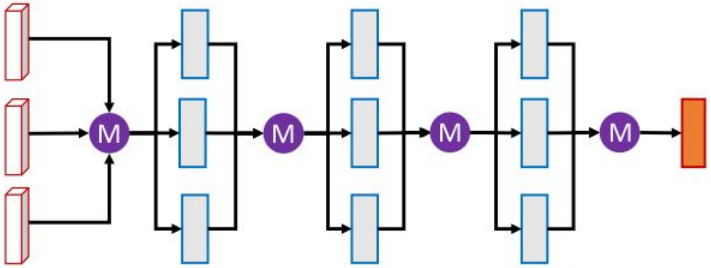
(b) Late Fusion

Deep Fusion



- Element wise mean for join operation
- More interaction among features
- More flexible when combined with drop-path training

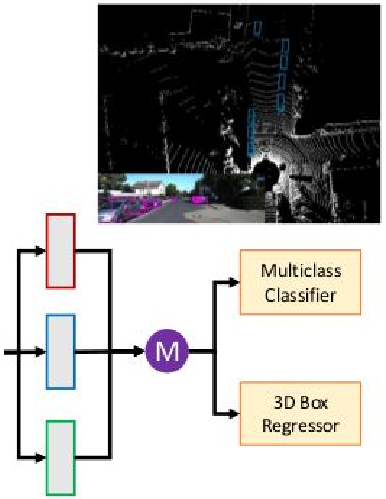
$$
\begin{aligned}
f_0 =& f_{BV} \oplus f_{FV} \oplus f_{RGB} \\
f_l =& \mathbf{H}_l^{BV}(f_{l-1}) \oplus \mathbf{H}_l^{FV}(f_{l-1}) \oplus \mathbf{H}_l^{RGB}(f_{l-1}), \\
& \forall l = 1, \cdots, L
\end{aligned}
$$



(c) Deep Fusion

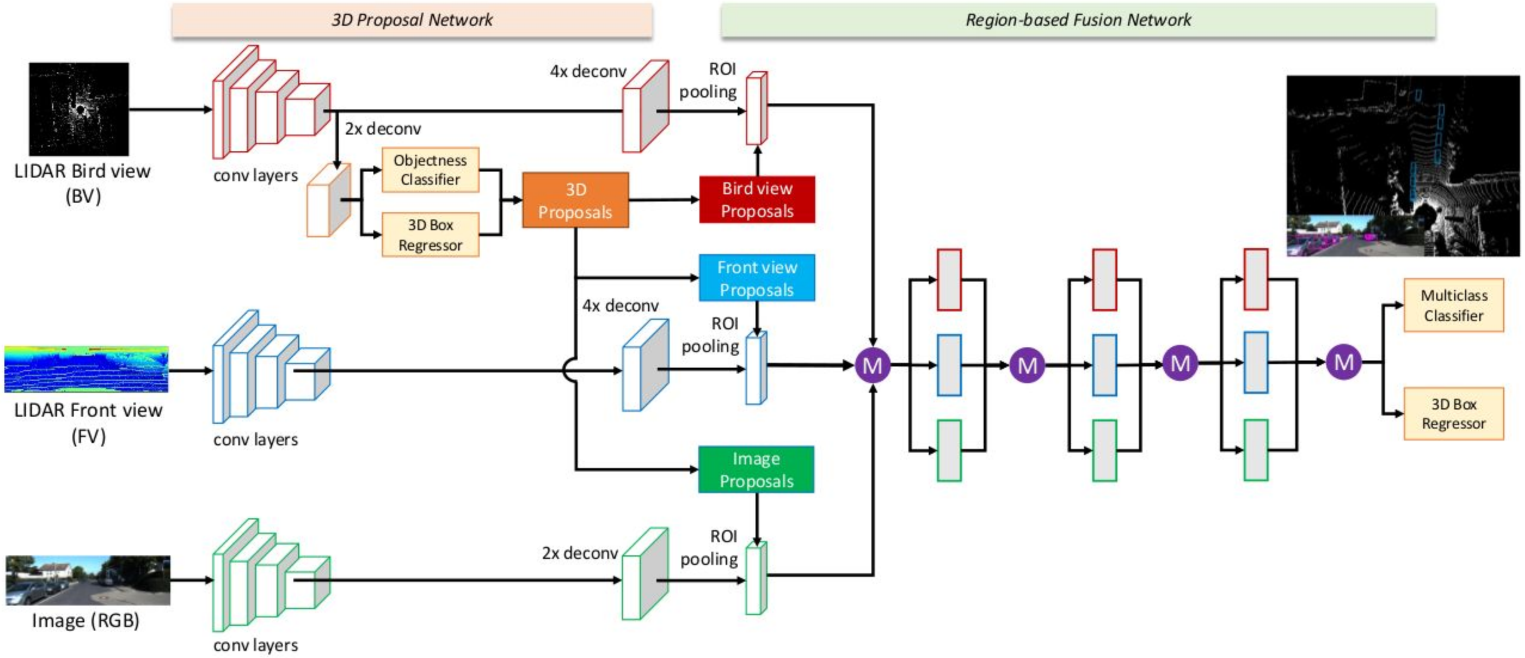
Oriented 3D Box Regression

- Uses “fusion” features of the multi-view network
- 8 corner representation $(\Delta x_0, \cdots, \Delta x_7, \Delta y_0, \cdots, \Delta y_7, \Delta z_0, \cdots, \Delta z_7)$
  - Normalized by diagonal length of proposal box
  - 24D vector is redundant, but works better than centres and sizes approach
  - Orientation computed from 3D box corners
- Multi-task loss
  - Cross-entropy for classification
  - Smooth L1 loss for 3D bounding box
- During inference, NMS on 3D boxes in BV with IoU < 0.05
  - Want no overlapping boxes



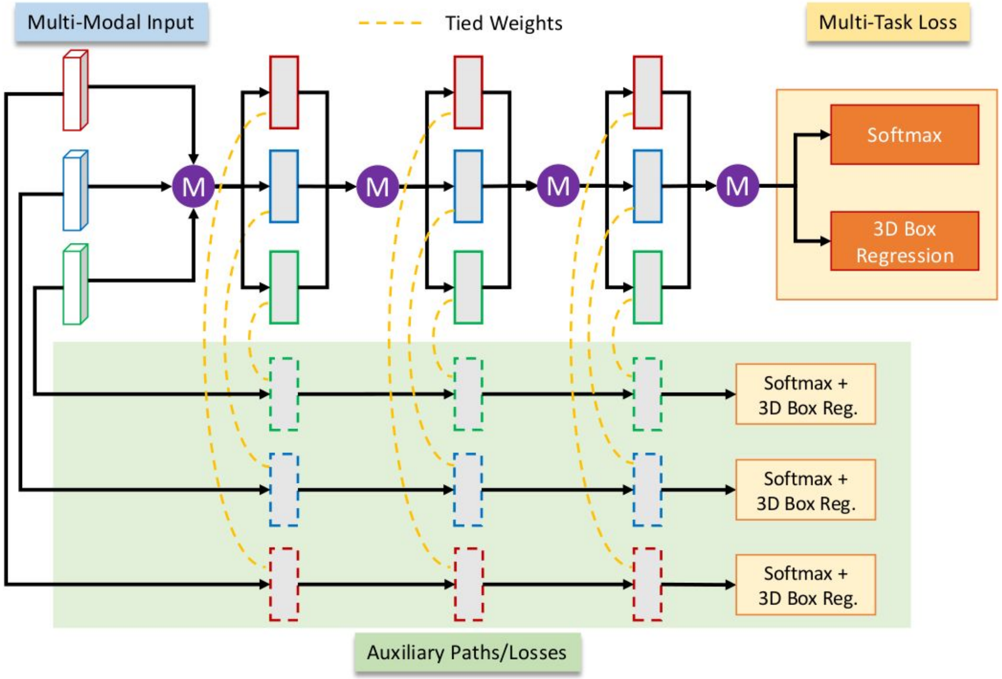
Network Regularization - Drop Path Training

- Randomly choose global or local drop-path with 50% probability
  - Global
    - Select single view from 3 views with equal probability
  - Local
    - Path input to each join node are dropped with 50% probability
    - At least 1 path is kept



Network Regularization - Auxiliary Losses

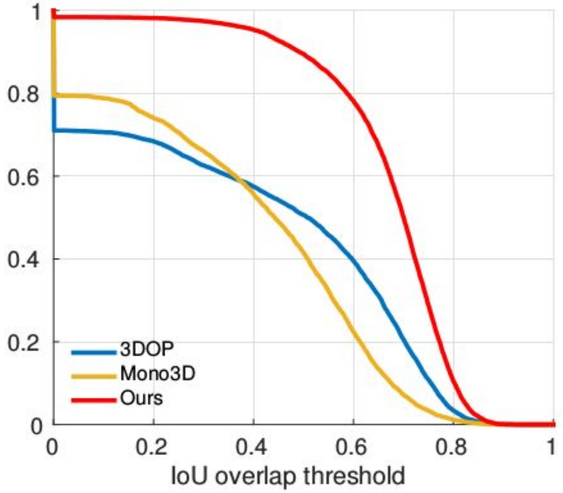
- Additional paths and losses
- Same number of layers as main network
- Parameter sharing with main network
- Strengthens the representation capability of each view
- All losses weighted equally
- Not used during inference



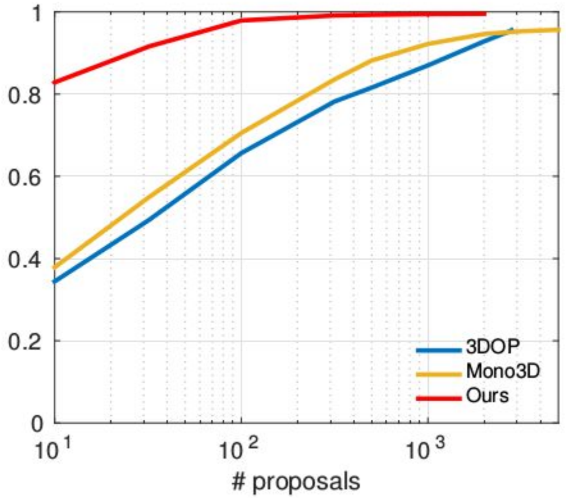
Training

- Trained end-to-end on training split (~3700 images)
- Trained with SGD
  - Learning rate of 0.001 for 100K iterations
  - 0.0001 for another 20K iterations
- 3D detections evaluated on validation set only

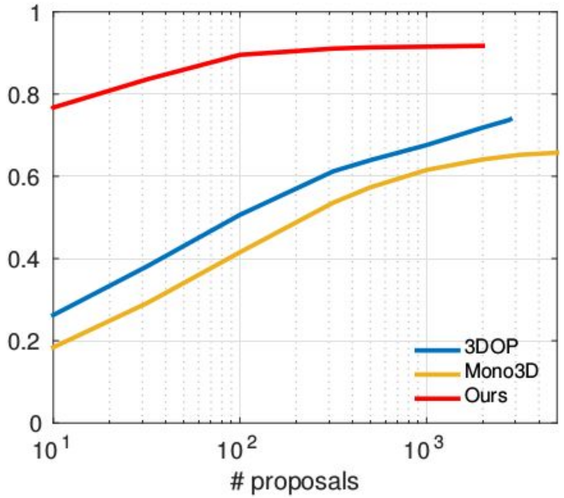
Results - 3D Proposal Recall



Recall vs IoU using 300 proposals



Recall vs # Proposals at 0.25 IoU

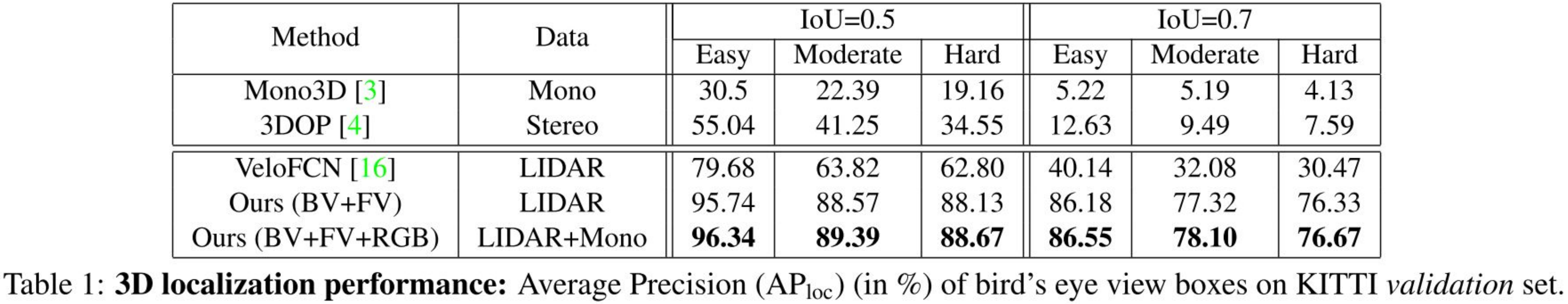


Recall vs # Proposals at 0.5 IoU

- Moderate KITTI Data
- With 300 proposals
  - 99.1% recall at 0.25 IoU
  - 91% recall at 0.5 IoU

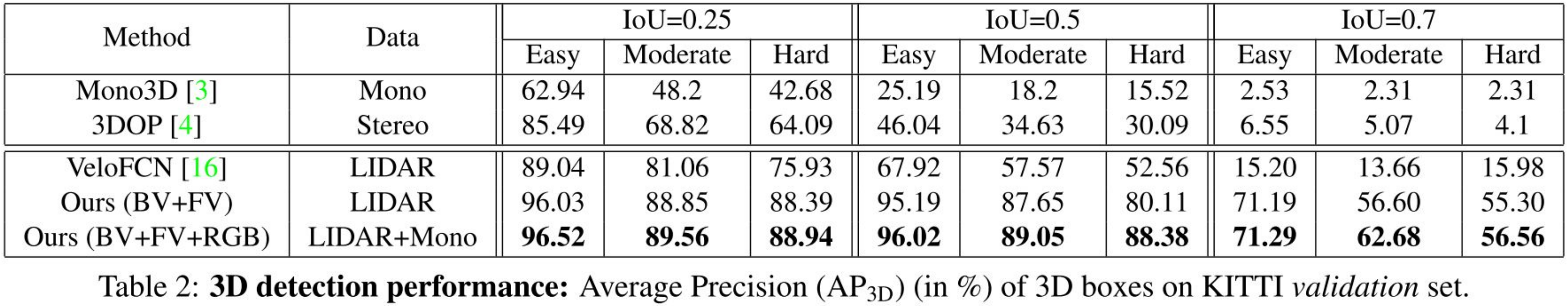
Results - 3D Detections

Inference time for one image: 0.36s on Titan X GPU



Method	Data	IoU=0.5			IoU=0.7		
		Easy	Moderate	Hard	Easy	Moderate	Hard
Mono3D [3]	Mono	30.5	22.39	19.16	5.22	5.19	4.13
3DOP [4]	Stereo	55.04	41.25	34.55	12.63	9.49	7.59
VeloFCN [16]	LIDAR	79.68	63.82	62.80	40.14	32.08	30.47
Ours (BV+FV)	LIDAR	95.74	88.57	88.13	86.18	77.32	76.33
Ours (BV+FV+RGB)	LIDAR+Mono	**96.34**	**89.39**	**88.67**	**86.55**	**78.10**	**76.67**

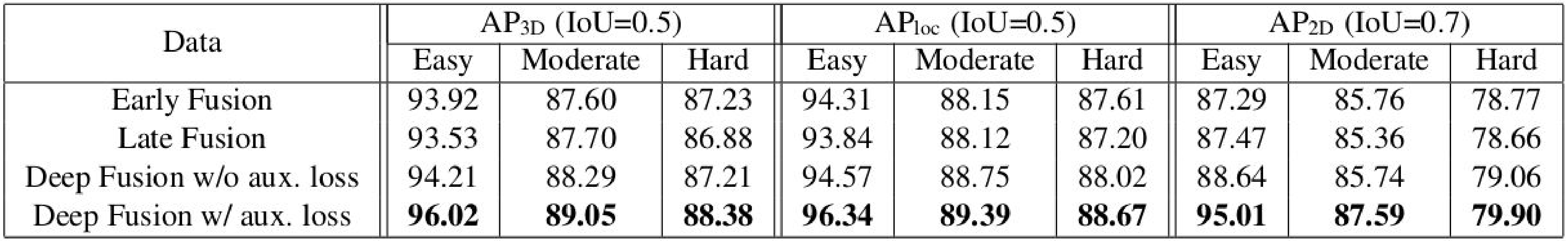
Table 1: **3D localization performance:** Average Precision (AP_{loc}) (in %) of bird's eye view boxes on KITTI *validation* set.



Method	Data	IoU=0.25			IoU=0.5			IoU=0.7		
		Easy	Moderate	Hard	Easy	Moderate	Hard	Easy	Moderate	Hard
Mono3D [3]	Mono	62.94	48.2	42.68	25.19	18.2	15.52	2.53	2.31	2.31
3DOP [4]	Stereo	85.49	68.82	64.09	46.04	34.63	30.09	6.55	5.07	4.1
VeloFCN [16]	LIDAR	89.04	81.06	75.93	67.92	57.57	52.56	15.20	13.66	15.98
Ours (BV+FV)	LIDAR	96.03	88.85	88.39	95.19	87.65	80.11	71.19	56.60	55.30
Ours (BV+FV+RGB)	LIDAR+Mono	**96.52**	**89.56**	**88.94**	**96.02**	**89.05**	**88.38**	**71.29**	**62.68**	**56.56**

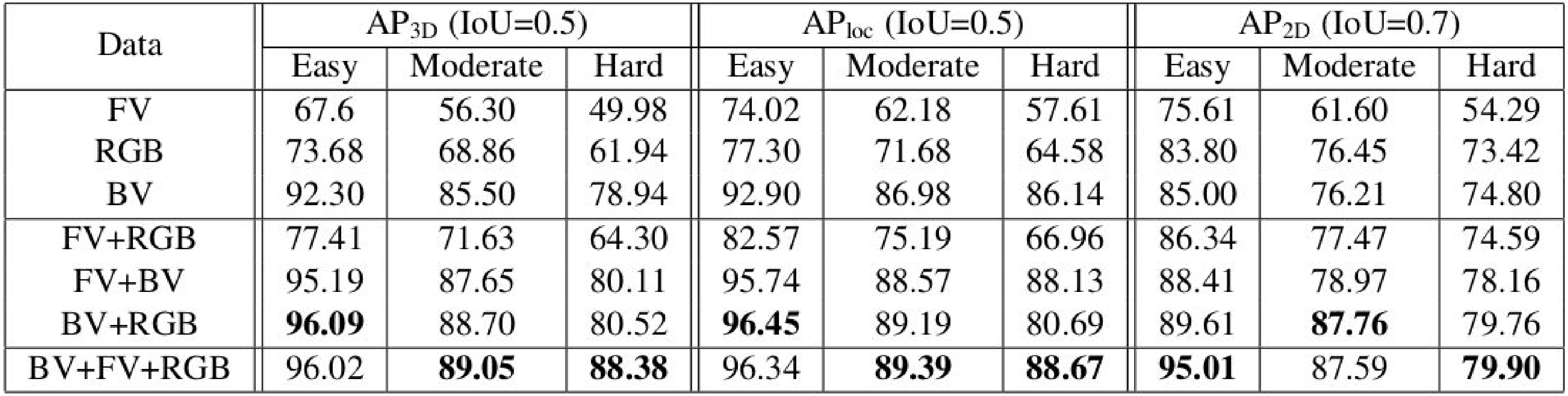
Table 2: **3D detection performance:** Average Precision (AP_{3D}) (in %) of 3D boxes on KITTI *validation* set.

Results - Feature Fusion, Multi-View Features



Data	AP_{3D} (IoU=0.5)			AP_{loc} (IoU=0.5)			AP_{2D} (IoU=0.7)		
	Easy	Moderate	Hard	Easy	Moderate	Hard	Easy	Moderate	Hard
Early Fusion	93.92	87.60	87.23	94.31	88.15	87.61	87.29	85.76	78.77
Late Fusion	93.53	87.70	86.88	93.84	88.12	87.20	87.47	85.36	78.66
Deep Fusion w/o aux. loss	94.21	88.29	87.21	94.57	88.75	88.02	88.64	85.74	79.06
Deep Fusion w/ aux. loss	**96.02**	**89.05**	**88.38**	**96.34**	**89.39**	**88.67**	**95.01**	**87.59**	**79.90**

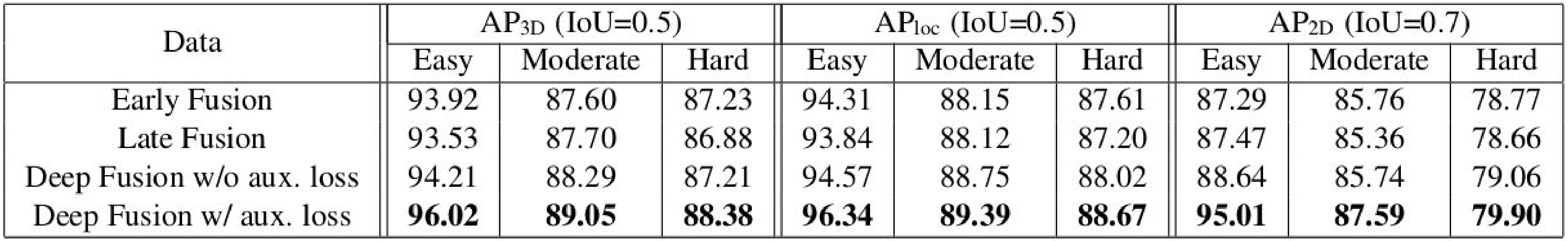
Table 3: **Comparison of different fusion approaches:** Peformance are evaluated on KITTI *validation* set.



Data	AP_{3D} (IoU=0.5)			AP_{loc} (IoU=0.5)			AP_{2D} (IoU=0.7)		
	Easy	Moderate	Hard	Easy	Moderate	Hard	Easy	Moderate	Hard
FV	67.6	56.30	49.98	74.02	62.18	57.61	75.61	61.60	54.29
RGB	73.68	68.86	61.94	77.30	71.68	64.58	83.80	76.45	73.42
BV	92.30	85.50	78.94	92.90	86.98	86.14	85.00	76.21	74.80
FV+RGB	77.41	71.63	64.30	82.57	75.19	66.96	86.34	77.47	74.59
FV+BV	95.19	87.65	80.11	95.74	88.57	88.13	88.41	78.97	78.16
BV+RGB	**96.09**	88.70	80.52	**96.45**	89.19	80.69	89.61	**87.76**	79.76
BV+FV+RGB	96.02	**89.05**	**88.38**	96.34	**89.39**	**88.67**	**95.01**	87.59	**79.90**

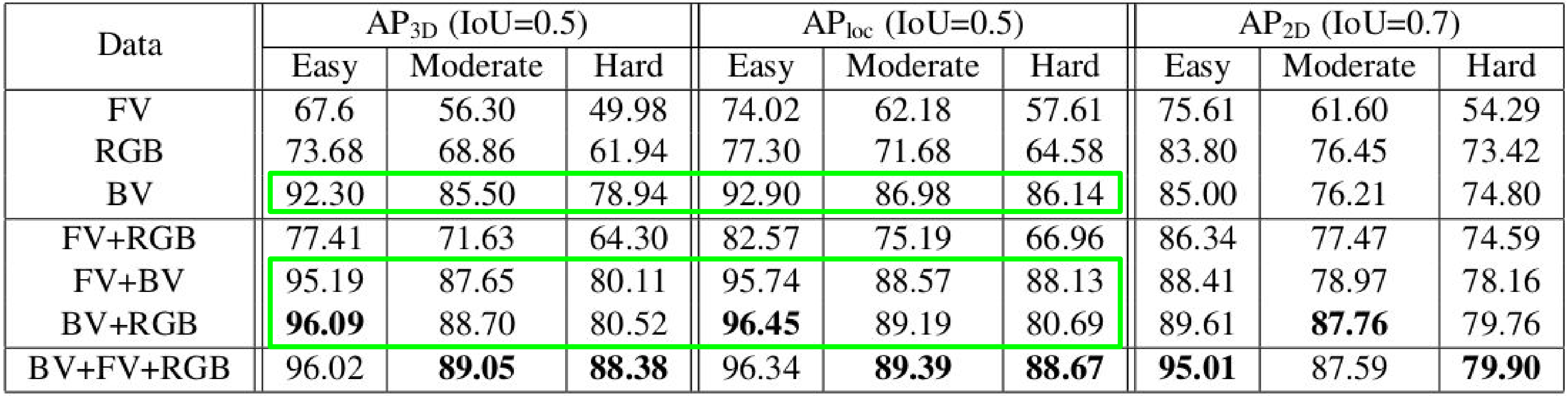
Table 4: **An ablation study of multi-view features**: Peformance are evaluated on KITTI *validation* set.

Results - Feature Fusion, Multi-View Features



Data	AP_{3D} (IoU=0.5)			AP_{loc} (IoU=0.5)			AP_{2D} (IoU=0.7)		
	Easy	Moderate	Hard	Easy	Moderate	Hard	Easy	Moderate	Hard
Early Fusion	93.92	87.60	87.23	94.31	88.15	87.61	87.29	85.76	78.77
Late Fusion	93.53	87.70	86.88	93.84	88.12	87.20	87.47	85.36	78.66
Deep Fusion w/o aux. loss	94.21	88.29	87.21	94.57	88.75	88.02	88.64	85.74	79.06
Deep Fusion w/ aux. loss	**96.02**	**89.05**	**88.38**	**96.34**	**89.39**	**88.67**	**95.01**	**87.59**	**79.90**

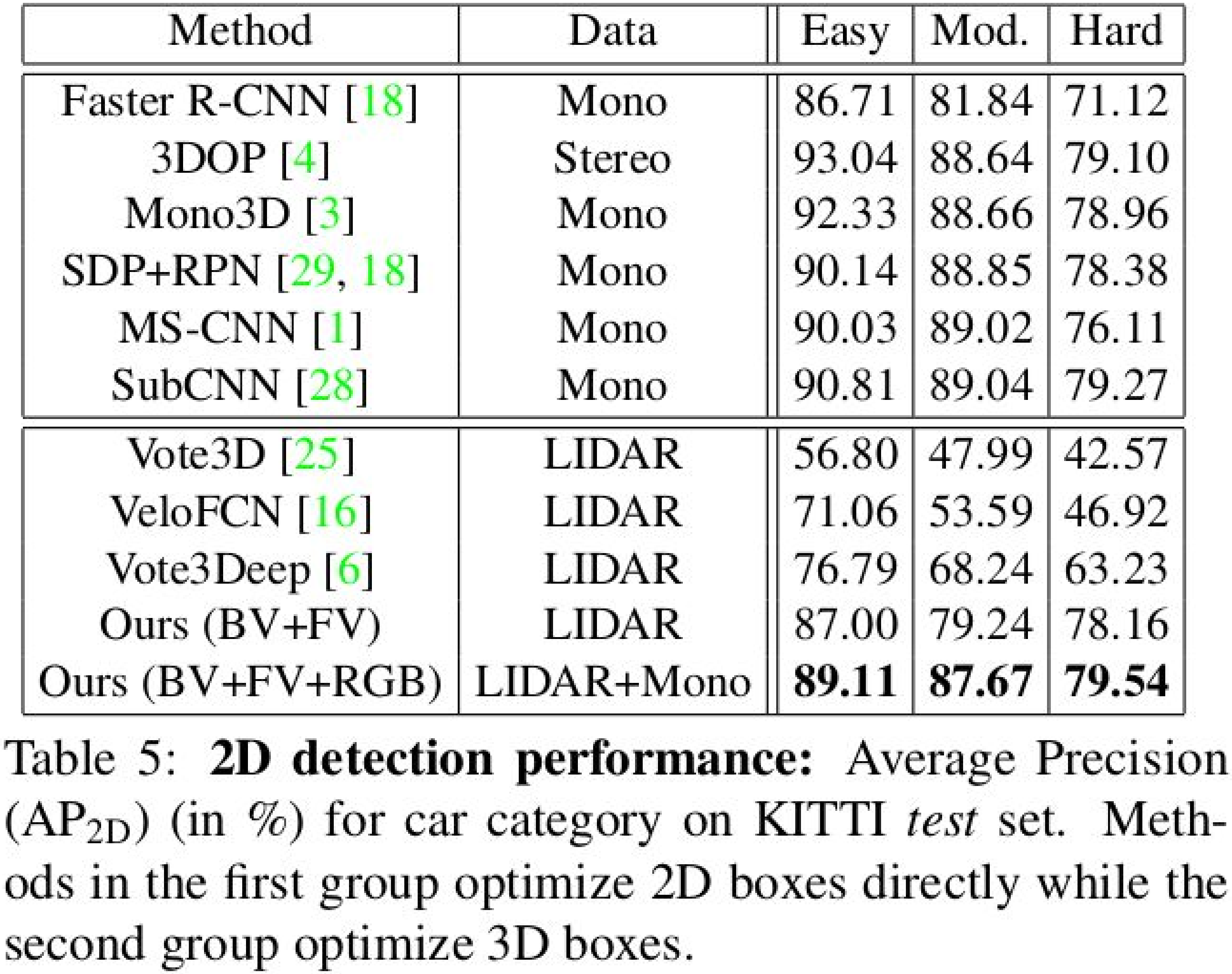
Table 3: **Comparison of different fusion approaches:** Peformance are evaluated on KITTI *validation* set.



Data	AP_{3D} (IoU=0.5)			AP_{loc} (IoU=0.5)			AP_{2D} (IoU=0.7)		
	Easy	Moderate	Hard	Easy	Moderate	Hard	Easy	Moderate	Hard
FV	67.6	56.30	49.98	74.02	62.18	57.61	75.61	61.60	54.29
RGB	73.68	68.86	61.94	77.30	71.68	64.58	83.80	76.45	73.42
BV	92.30	85.50	78.94	92.90	86.98	86.14	85.00	76.21	74.80
FV+RGB	77.41	71.63	64.30	82.57	75.19	66.96	86.34	77.47	74.59
FV+BV	95.19	87.65	80.11	95.74	88.57	88.13	88.41	78.97	78.16
BV+RGB	**96.09**	88.70	80.52	**96.45**	89.19	80.69	89.61	**87.76**	79.76
BV+FV+RGB	96.02	**89.05**	**88.38**	96.34	**89.39**	**88.67**	**95.01**	87.59	**79.90**

Table 4: **An ablation study of multi-view features**: Peformance are evaluated on KITTI *validation* set.

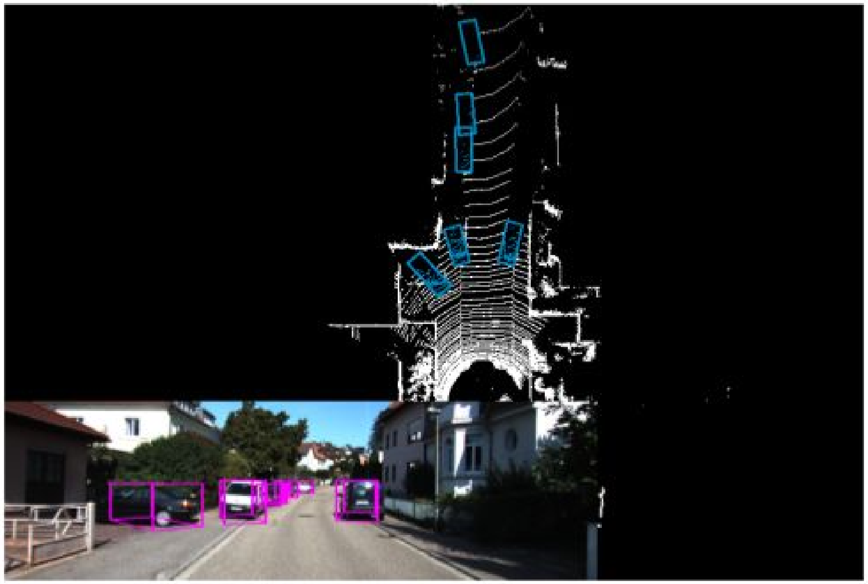
Results - 2D Detections

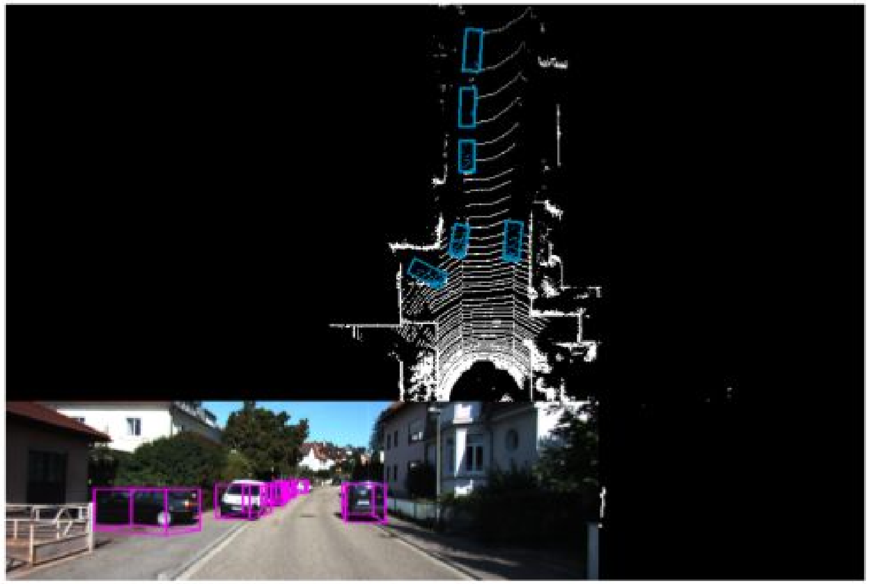


Method	Data	Easy	Mod.	Hard
Faster R-CNN [18]	Mono	86.71	81.84	71.12
3DOP [4]	Stereo	93.04	88.64	79.10
Mono3D [3]	Mono	92.33	88.66	78.96
SDP+RPN [29, 18]	Mono	90.14	88.85	78.38
MS-CNN [1]	Mono	90.03	89.02	76.11
SubCNN [28]	Mono	90.81	89.04	79.27
Vote3D [25]	LIDAR	56.80	47.99	42.57
VeloFCN [16]	LIDAR	71.06	53.59	46.92
Vote3Deep [6]	LIDAR	76.79	68.24	63.23
Ours (BV+FV)	LIDAR	87.00	79.24	78.16
Ours (BV+FV+RGB)	LIDAR+Mono	**89.11**	**87.67**	**79.54**

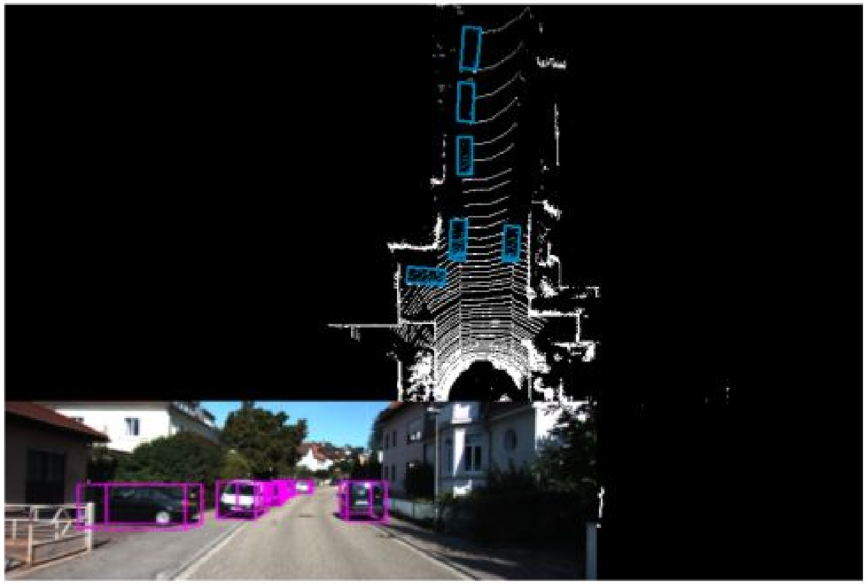
Table 5: **2D detection performance:** Average Precision (AP_{2D}) (in %) for car category on KITTI *test* set. Methods in the first group optimize 2D boxes directly while the second group optimize 3D boxes.

Qualitative Results

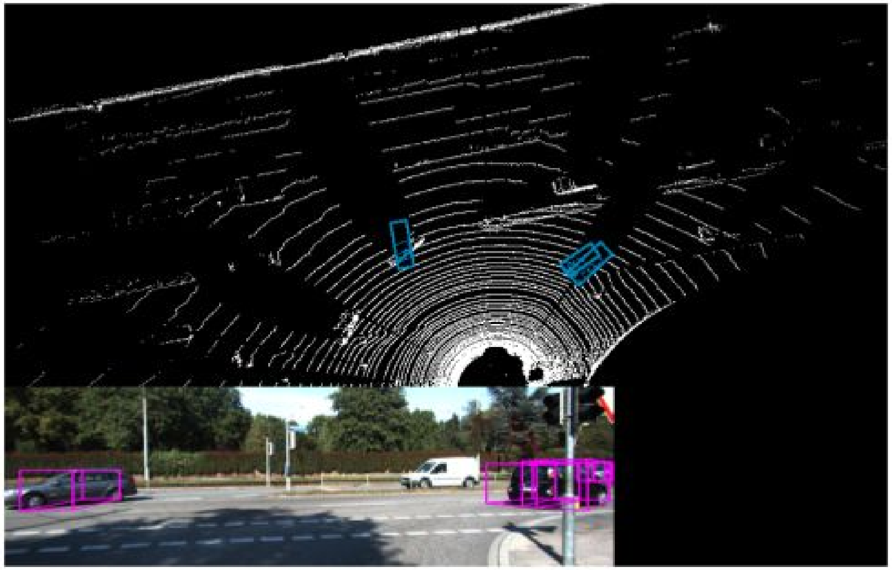


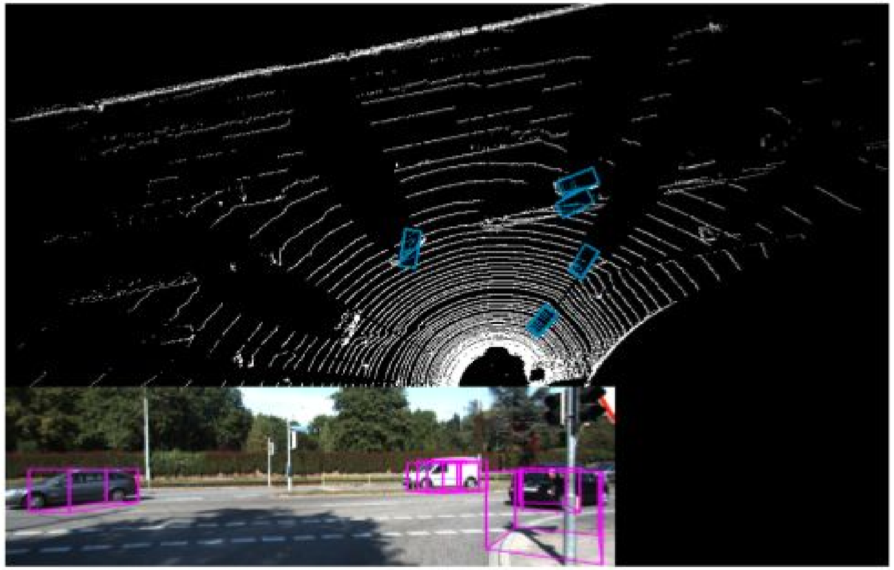


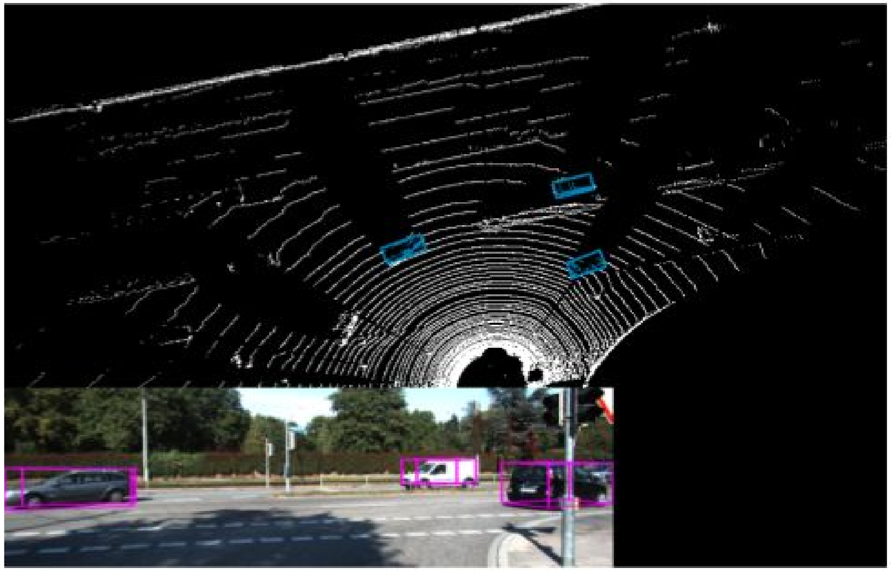
- Top Left: 3DOP
- Top Right: VeloFCN
- Right: Multi-View



Qualitative Results







- Top Left: 3DOP
- Top Right: VeloFCN
- Right: Multi-View

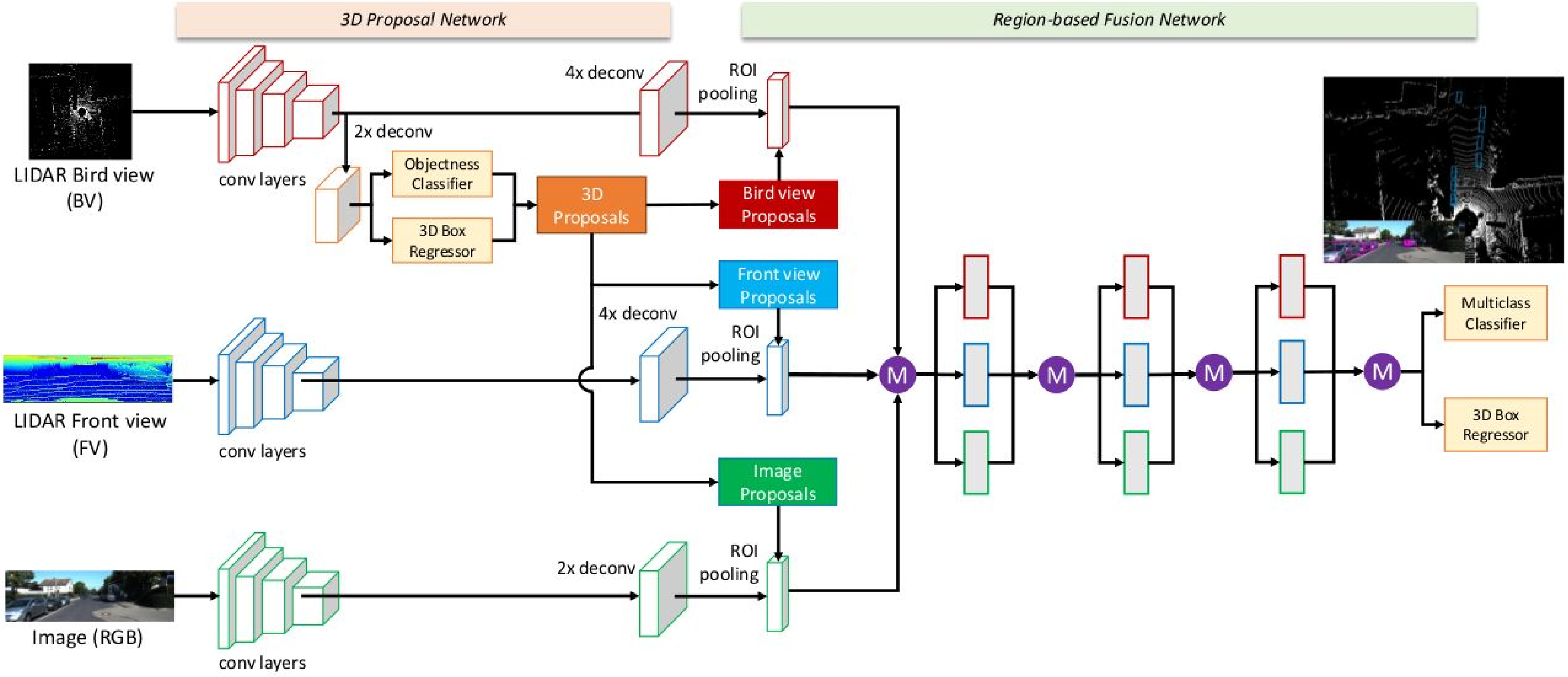
Summary

- Multi-view input representations
- Region-based deep fusion network
- Improves LIDAR and image-based methods
- Outperforms other methods by ~25% and 30% AP for 3D detections
- 2D detections are also competitive

Shortcomings / Improvements

- LIDAR vs Stereo Data
- Inference time 0.36s almost fast enough
  - Pre-processed input representations
- Code not released
- 3D detections only tested on validation set
- No detections for pedestrians or cyclists
  - Points may be too sparse
  - Data augmentation required for more instances in KITTI

Questions?


3D Proposal Network
Region-based Fusion Network
LIDAR Bird view (BV)
conv layers
2x deconv
4x deconv
ROI pooling
Objectness Classifier
3D Box Regressor
3D Proposals
Bird view Proposals
Front view Proposals
Image Proposals
LIDAR Front view (FV)
conv layers
4x deconv
ROI pooling
Image (RGB)
conv layers
2x deconv
ROI pooling
M
M
M
M
Multiclass Classifier
3D Box Regressor