Strategies and Methods for Sensor Fusion

Robert Laganière
Professor, University of Ottawa
CEO, Sensor Cortek Inc

Perception and sensors

- Human perception is the faculty of capturing the environment using senses and mind
- Machine perception is the faculty of capturing the environment using sensors and processors
  - Each sensor captures information in its own way
  - Processor(s) integrate (fuse) these incoming source of data to produce perceptual information
  - Good sensor fusion should make use of both:
    - The complementarity of the sensor data
    - The redundancy of the sensor data

Few popular sensors

- Camera
  - A passive sensor that captures visible light emitted and reflected by the environment
- Thermal camera
  - A passive sensor that detects the heat emitted by objects in the environment
- Lidar
  - An active sensor that emits pulsed laser to measure range
- Radar
  - An active sensor that transmits and receives frequency modulated waveforms to detect moving targets

Camera:

- **Pros**
  - Low power, inexpensive
  - Best for classification/ recognition
  - Can be infrared
  - No interference (multiple cameras)
  - High resolution
  - AI research very advanced
- **Cons**
  - Dependent on lighting and visibility
  - Affected by shadows/reflections
  - Gets dirty easily
  - No direct 3D (without stereo)

Thermal:

- **Pros**
  - Day/night visibility
  - Good under most weather and air conditions
  - Sees through thin material
  - Accurate temperature measurement
  - Offers some privacy protection

- **Cons**
  - Expensive (lens)
  - Affected by emissivity and reflection of objects
  - Cannot read texture and text
  - Can be difficult to interpret under erratic temperature conditions

Lidar:

- **Pros**
  - Day/night capture
  - Direct 3D information
  - Excellent accuracy
  - Can be long range

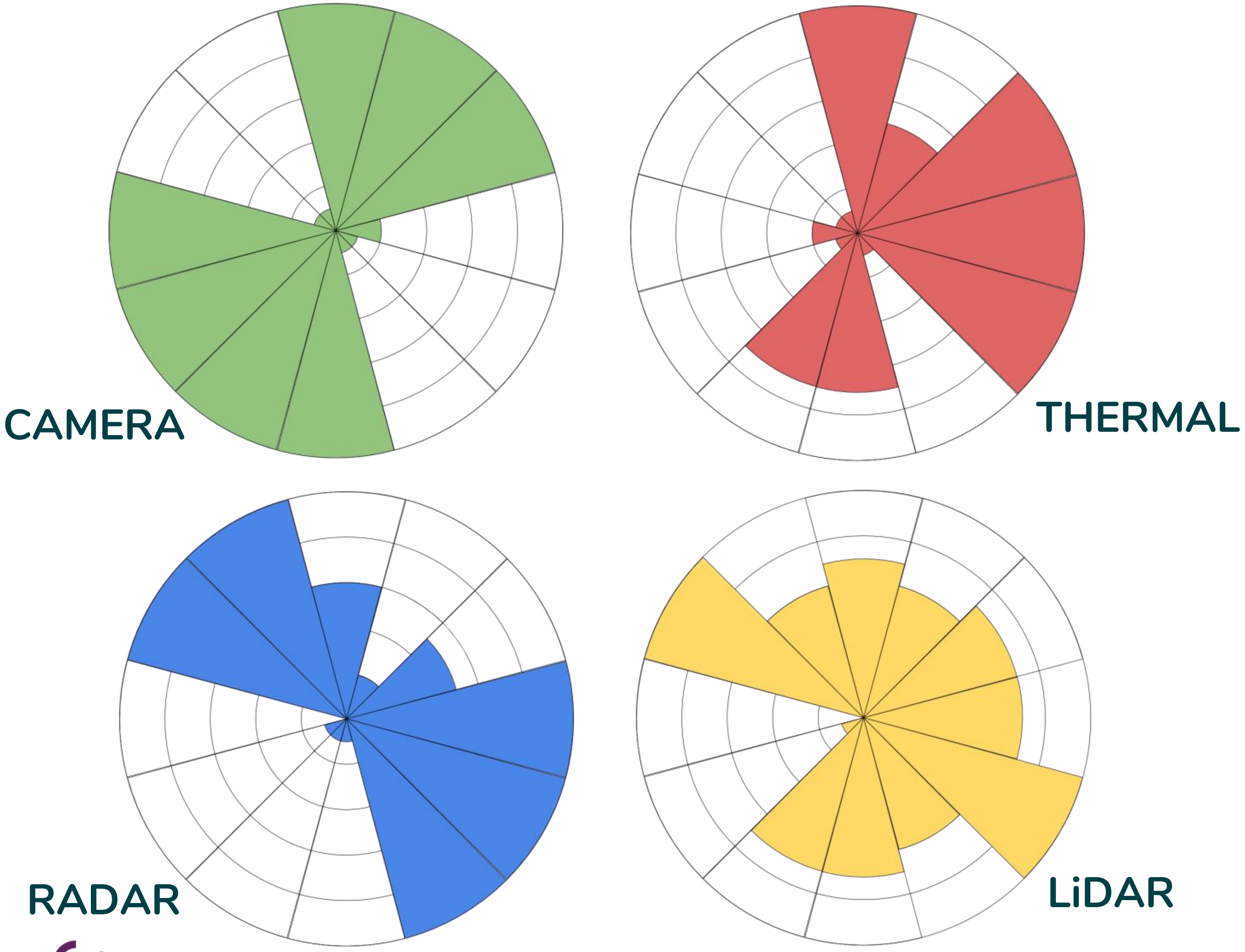
- **Cons**
  - Expensive
  - Produces sparse data
  - Captures shape, not appearance
  - Becomes noisy under fog, rain and snow
  - Generally includes mechanical parts
  - Subject to interference

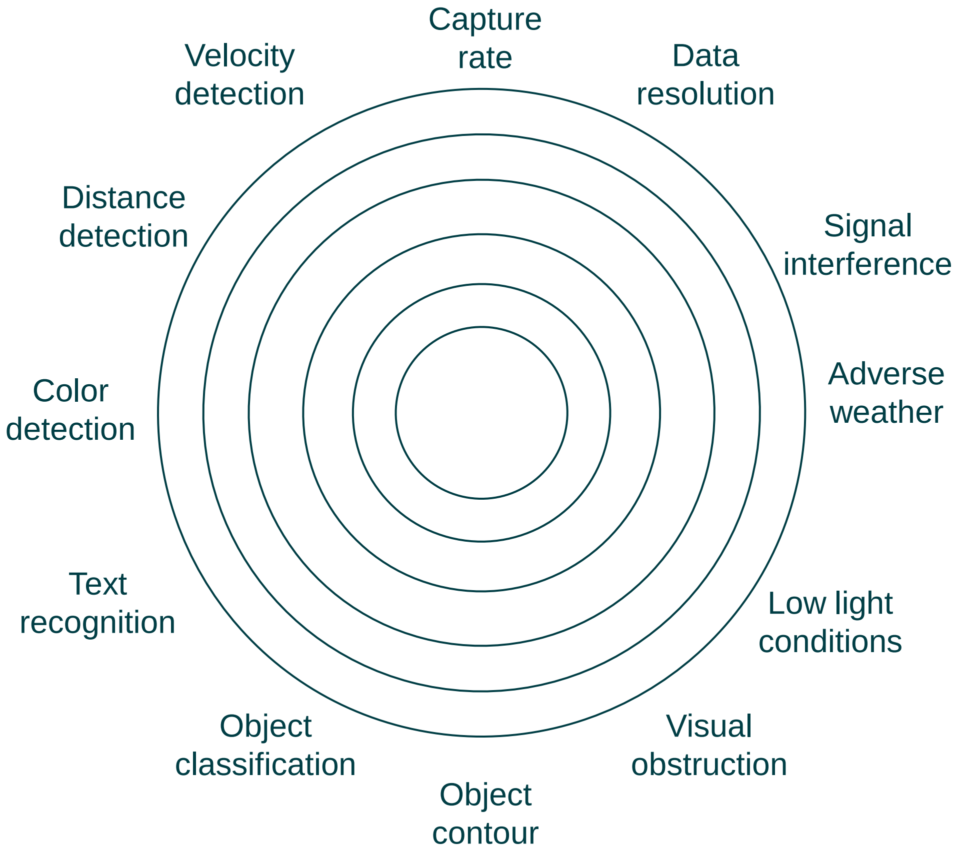
Radar:

- **Pros**
  - Captures direction, distance and speed
  - Inexpensive
  - Reliable solid-state technology
  - Day/night capture
  - Good immunity to weather conditions

- **Cons**
  - Poor angular resolution
  - Can’t detect small objects
  - Noisy
  - Limited classification ability
  - Subject to interference (e.g. background metallic objects)

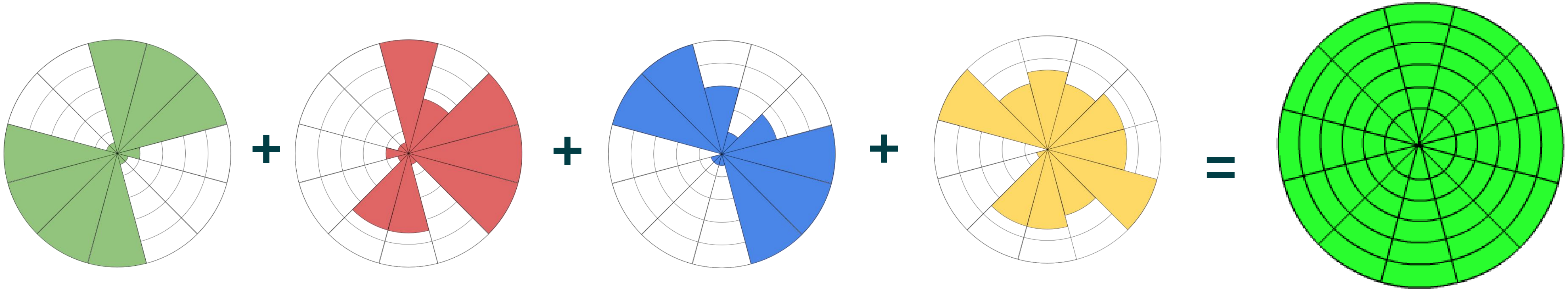
No one is perfect…





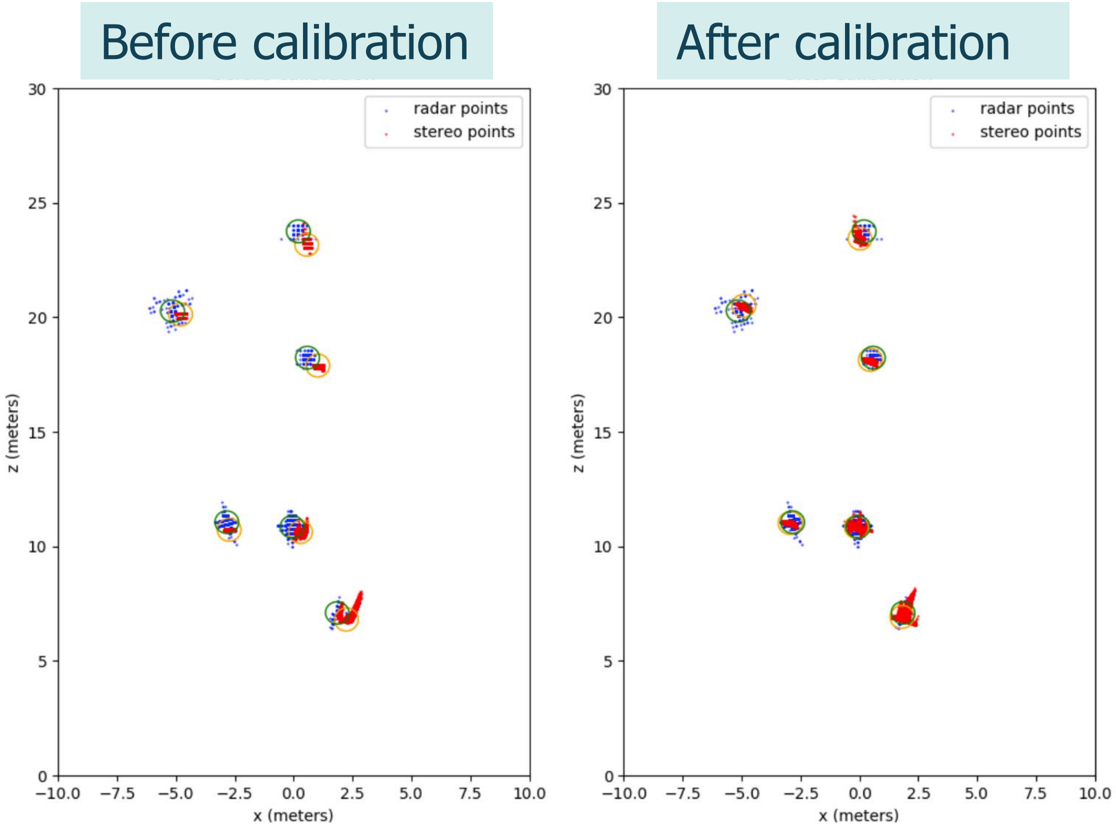
SENSOR CORTEK

Solution: sensor fusion



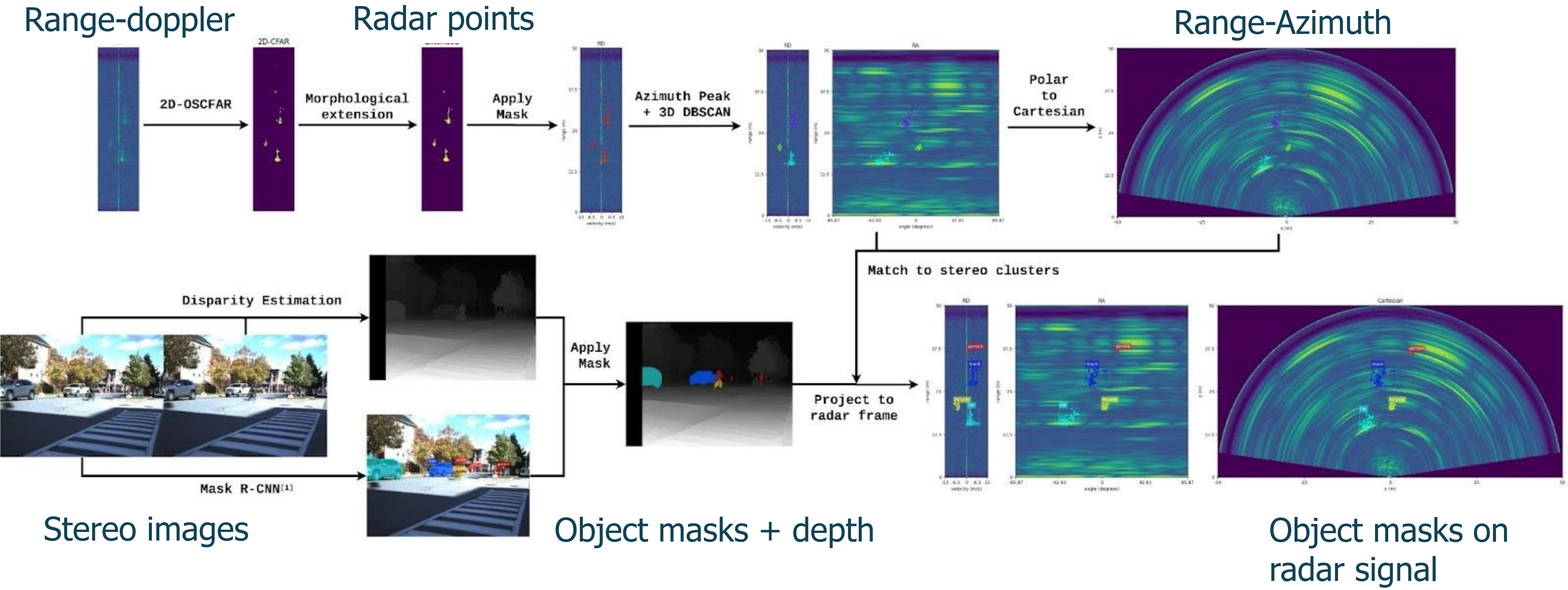
Sensor fusion : prerequisite

- Sensors must be calibrated and registered one with respect to the others
- Sensors must be synchronized or must use a common time reference
- To produce training data, multimodal sensor data must be jointly annotated



$$P_{stereo}^{radar} = \underset{P_{stereo}^{radar}}{\arg\min} \left[\left(X_{radar} - P_{stereo}^{radar} \cdot X_{stereo} \right) + \left(\Omega_{radar} - P_{stereo}^{radar} \cdot \Omega_{stereo} \right) \right]^2$$

Example: radar / stereo auto-annotation



Multi-sensor fusion strategies

Early fusion

Fuse sensor data and then perform inference using a network

Late fusion

Perform inference from each sensor data and then merge the predictions

Mid-level fusion

Fuse intermediate representations from sensor data and then train a predictor

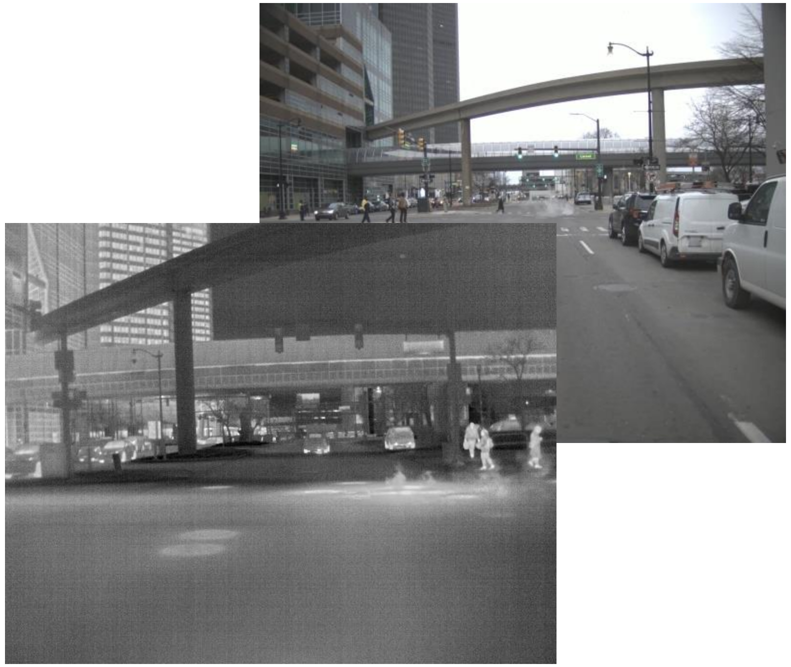
Sequential fusion

Use sensor data inference in sequence to refine predictions

See FrustrumNet paper in references

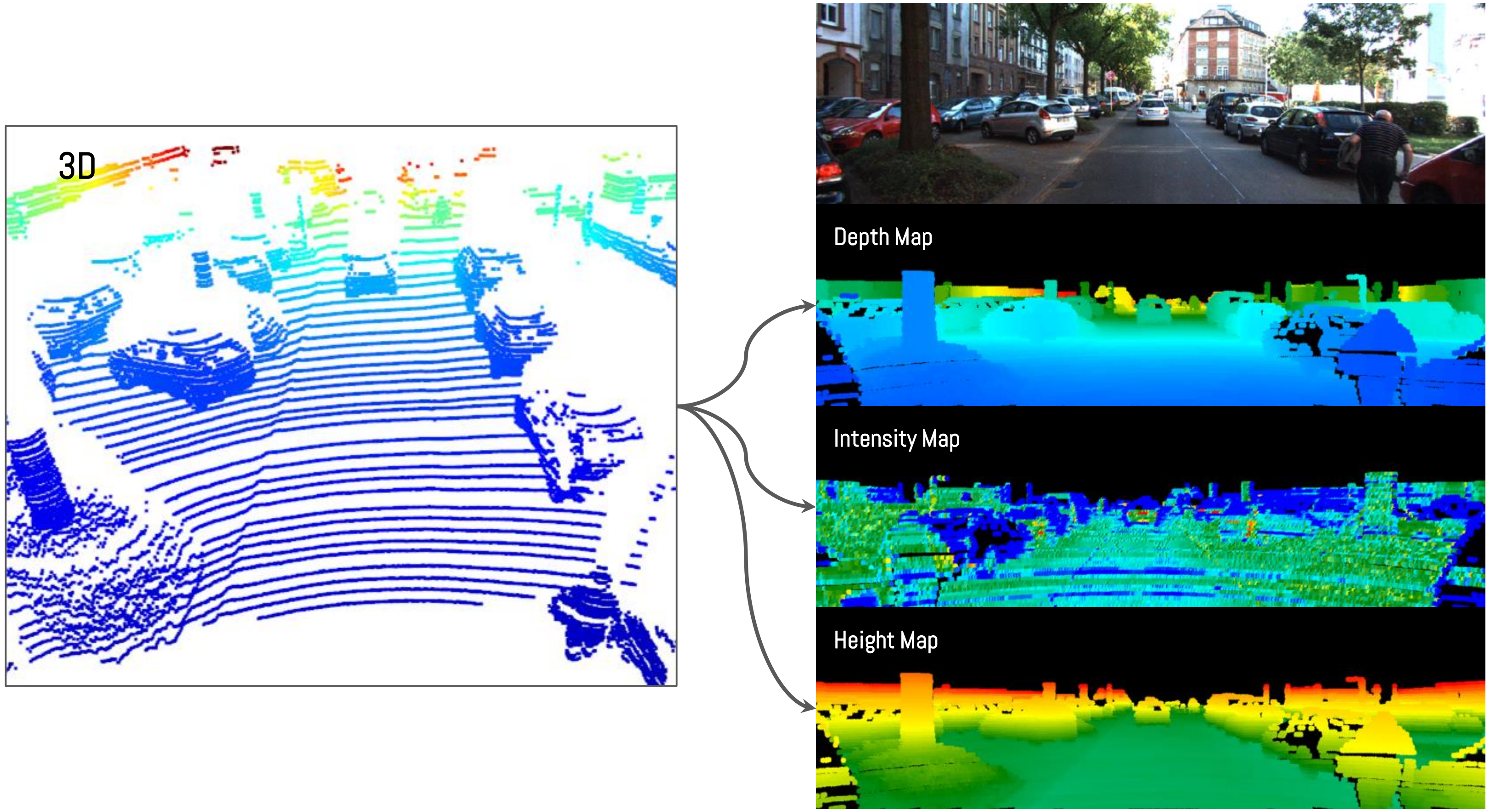
Early fusion

- Fuse sensor data by creating a common tensor representation
  - Which operator should be used for fusion?
  - Does not exploit the specific characteristics of each individual sensor
- Ideally, sensor data should be similar in nature
  - If not, compatible representations should be built
  - Information could be lost when the sensors do not have the same (temporal and spatial) resolution



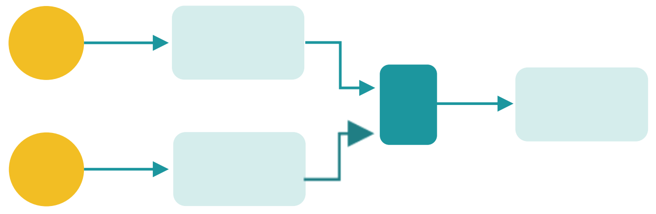
Early fusion: sensor data representation

- Example: merging a camera frame with a Lidar 3D point cloud



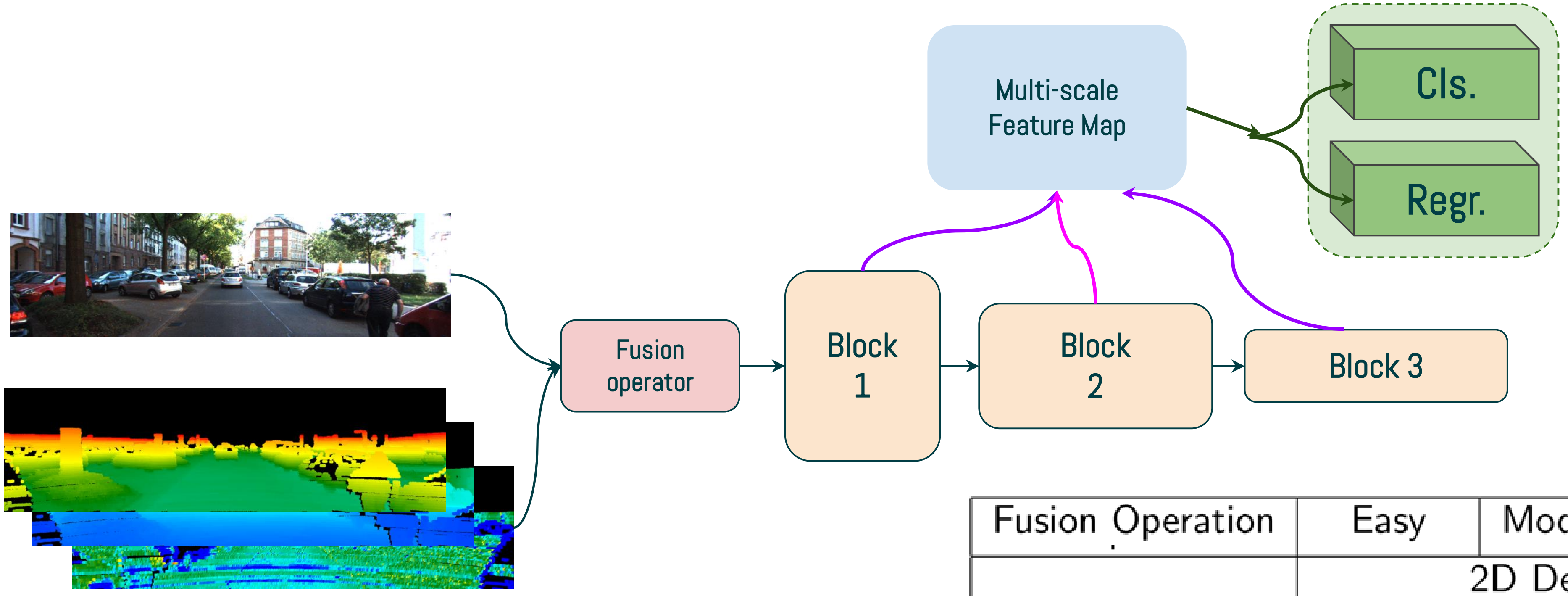
Merging sensor data: fusion operators

- Multi-modal sensor data (or feature maps) must be merged within a neural network architecture
  - This applies to all sensor fusion strategies
- Main fusion operators
  - Concatenation
  - Arithmetic (addition, multiplication)
  - Order-statistic (max, median)
  - Neural subnetwork
  - Learnable fusion



Early fusion: example

- Camera + Fontal view Lidar fusion network



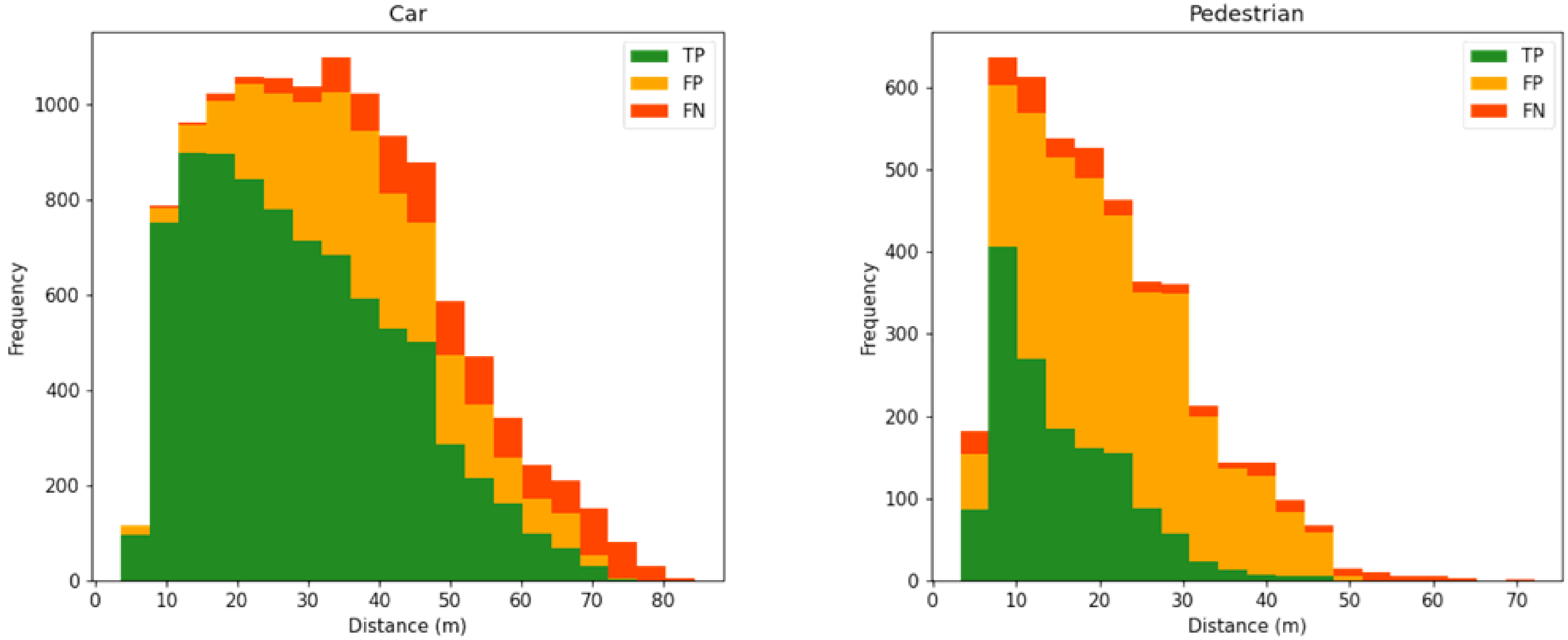
Fusion Operation	Easy	Moderate	Hard
	2D Detection		
Addition	71.12%	69.14%	68.93%
Multiplication	**84.32%**	**79.71%**	**77.37%**

Late fusion

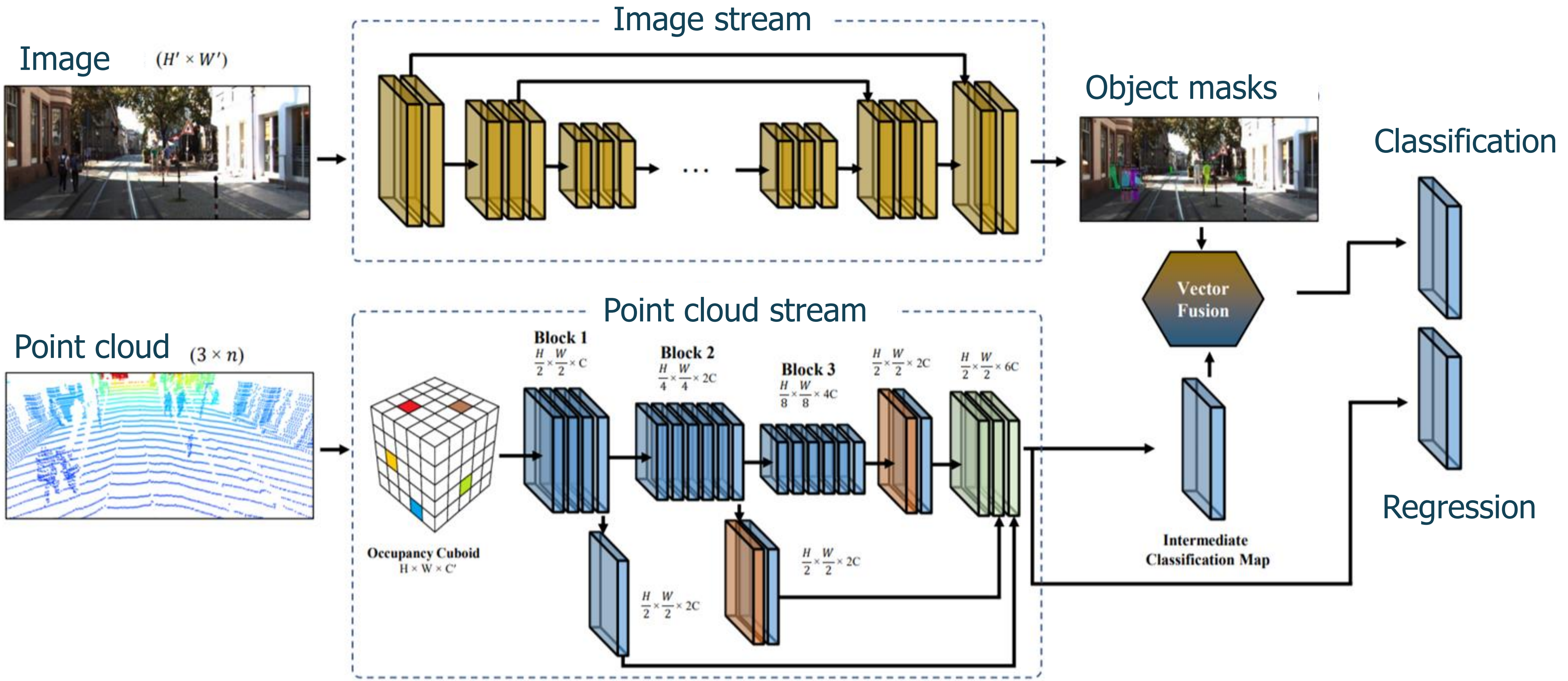
- Each sensor is processed independently
- The two resulting feature maps are then combined into one
- A classifier produces a prediction from this hybrid map

Late fusion: example

- Late fusion networks are often used to increase precision
- Example: car and pedestrian detection

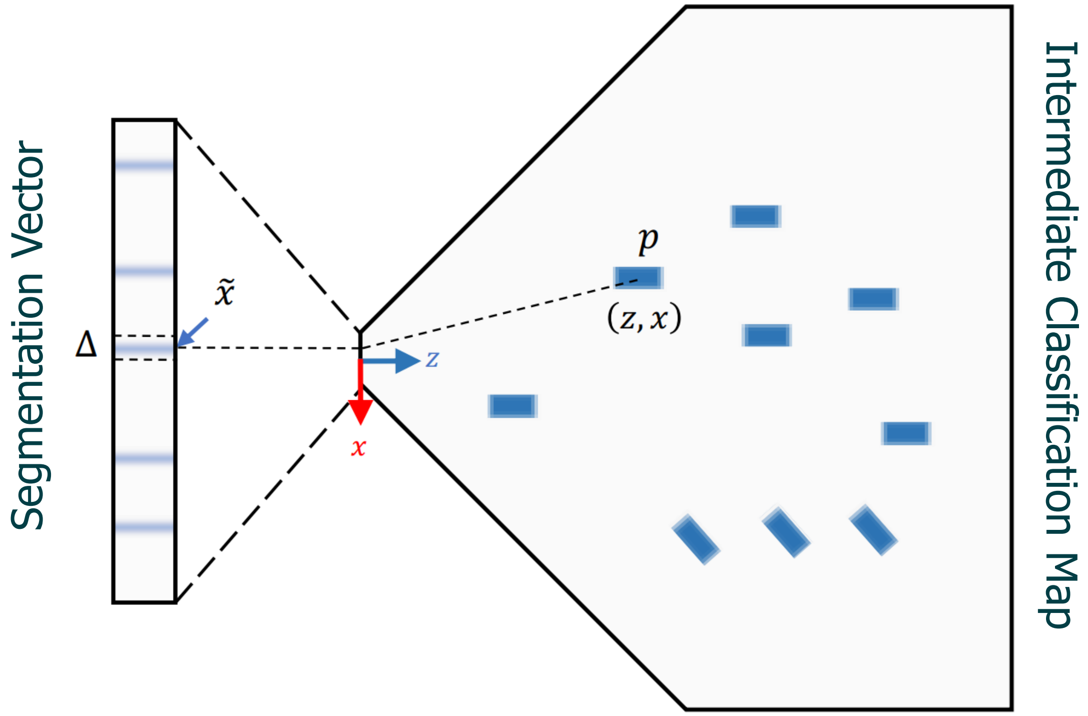


Late fusion: Camera frame + Fontal view Lidar



Late fusion: vector fusion operator

- Segmentation vector is max of instance segmentation mask along y-axis
- Lidar bird's eye view (BEV) intermediate classification map reprojected onto segmentation vector
- Positive density is the integration of BEV projection over object size interval Δ

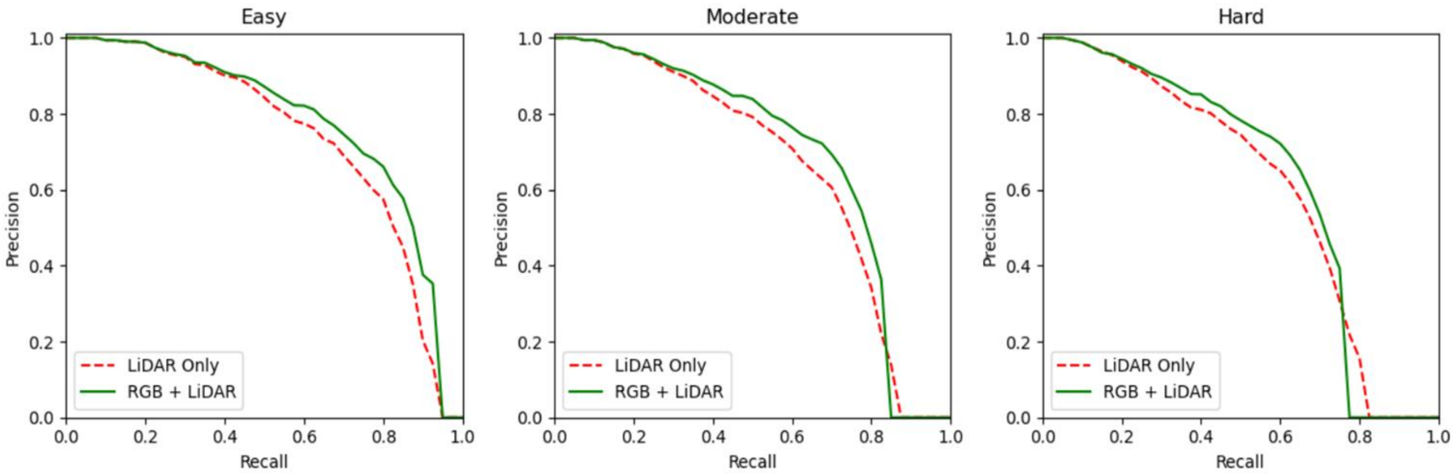


Late fusion: results

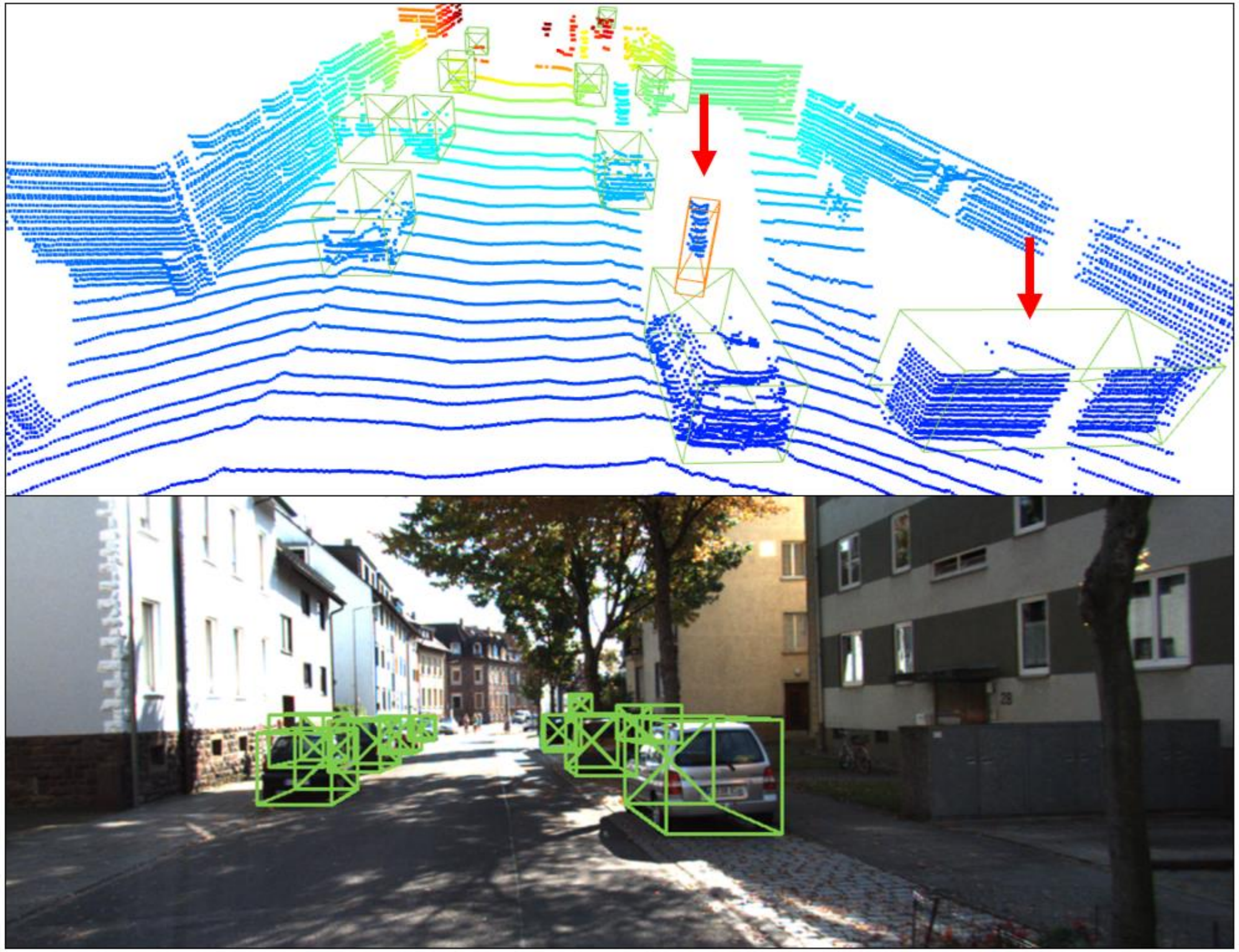
- Accuracy of Bird's Eye View predictions (BEV) and 3D Bounding Boxes predictions

	Modality	BEV E	BEV M	BEV H	3D E	3D M	3D H
Car	LiDAR Only	94.63	88.10	85.49	87.35	75.47	71.97
	RGB + LiDAR	92.69	88.14	85.73	87.41	75.50	70.91
	Delta	-1.94	+0.05	+0.24	+0.06	+0.03	-1.06
Pedestrian	LiDAR Only	72.89	65.06	59.52	63.97	56.50	49.86
	RGB + LiDAR	76.74	68.23	61.17	67.06	59.02	52.08
	Delta	+3.85	+3.17	+1.65	+3.09	+2.52	+2.22

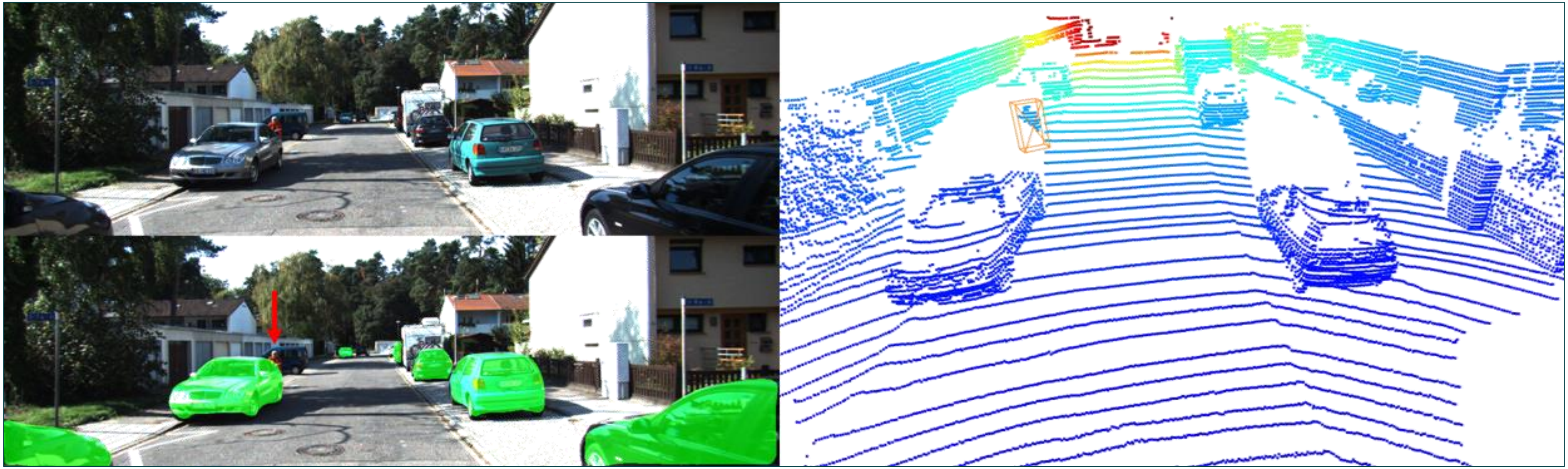
E: easy testset
M: moderate testset
H: hard testset



Late fusion: sample result



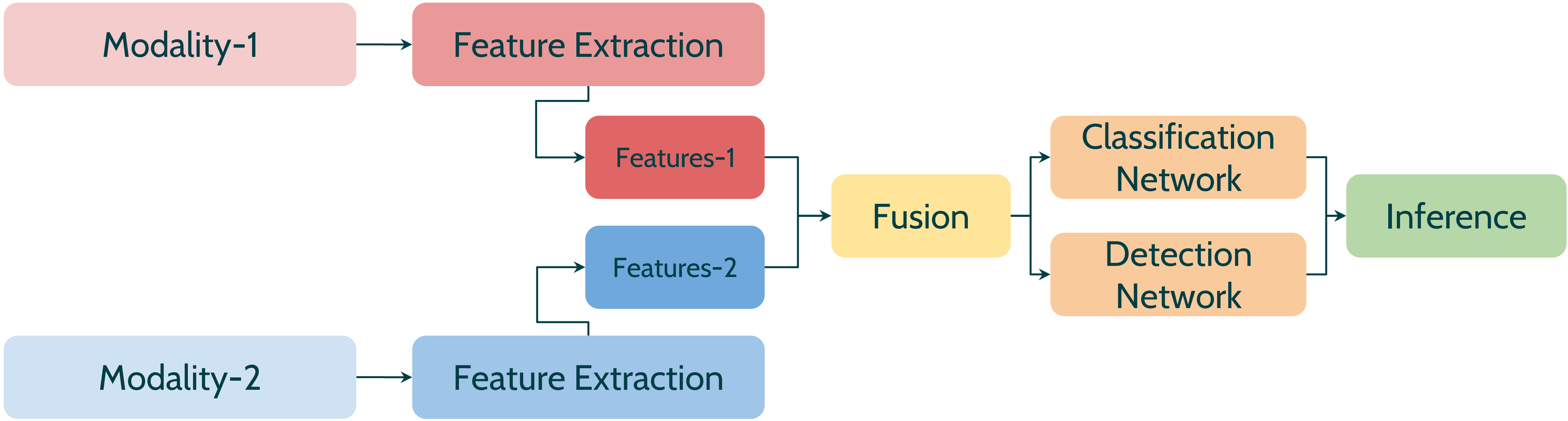
Late fusion: sample result (failure case)



Mid-level fusion

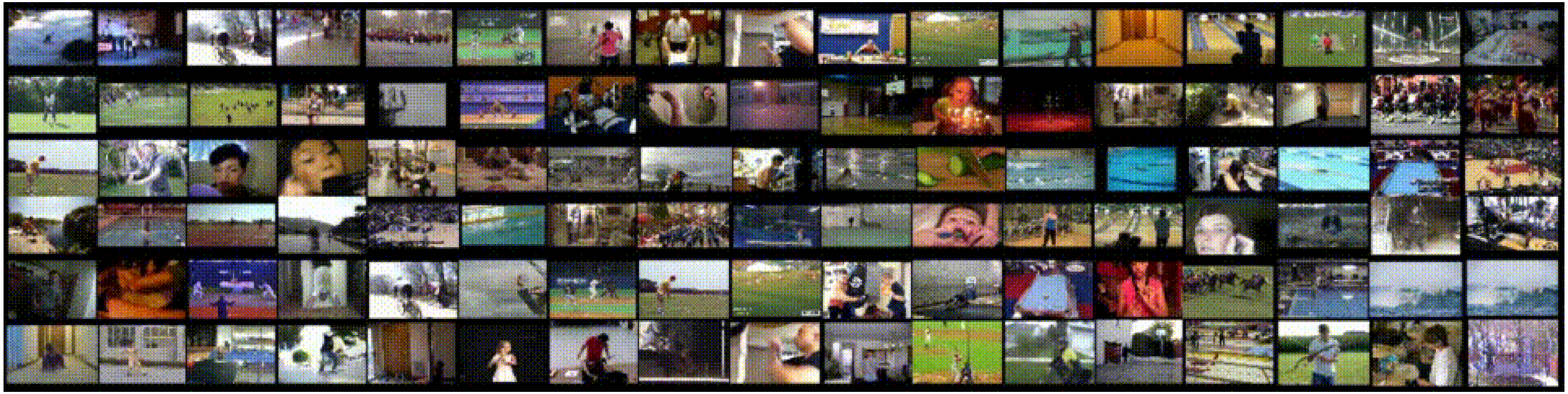
- Independent feature maps are generated from each sensor
- These two branches are combined and then a new CNN branch generates prediction
- Because of this additional branch, more complex feature fusion mechanism can be used
- But mid-level fusion model are generally more difficult to train!
  - Lots of parameters
  - Back-propagation in two directions

Mid-level fusion



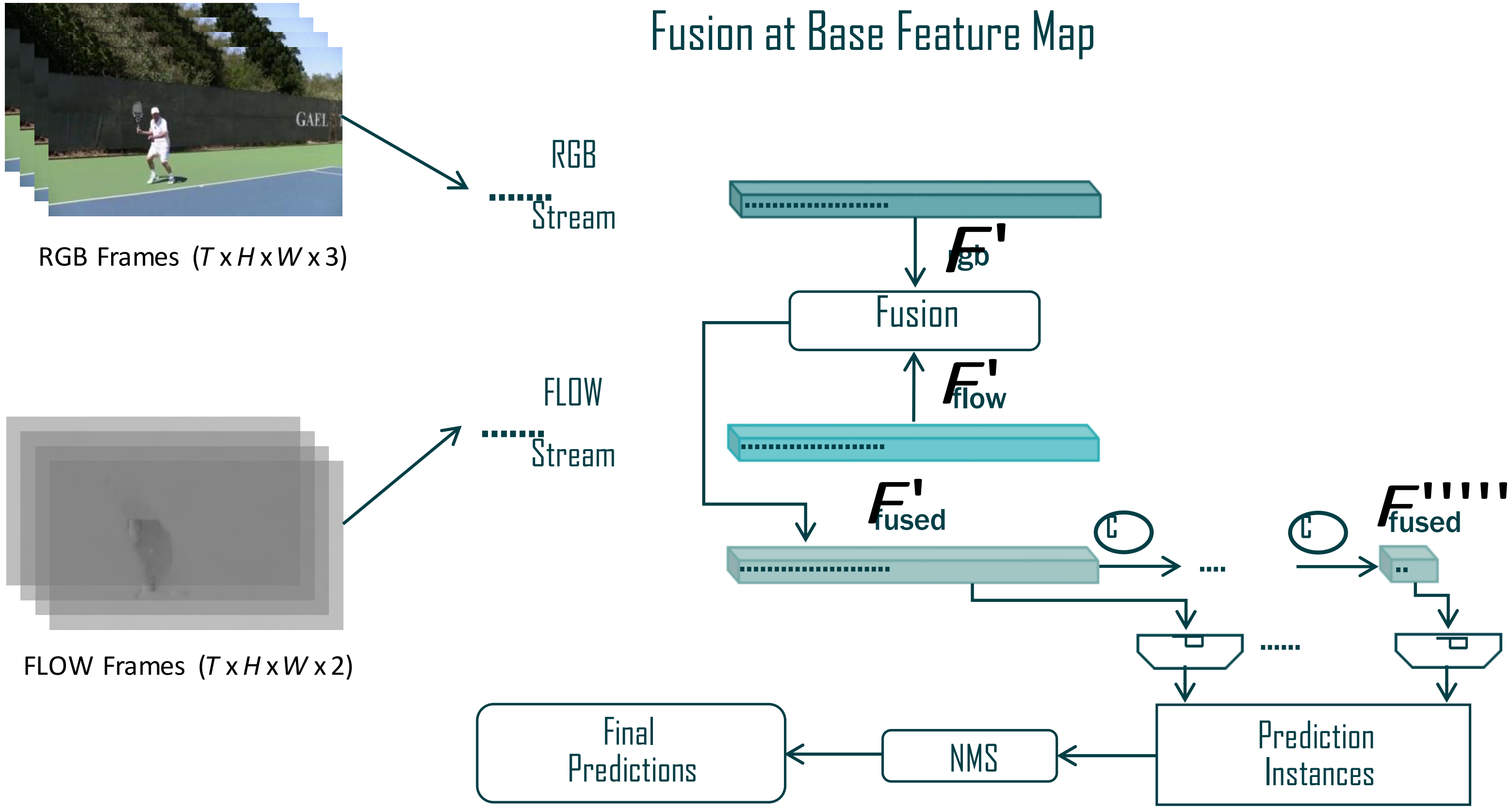
Mid-level fusion: temporal activity recognition

- Given a temporally untrimmed long video sequence, the goal is to classify and temporally localize each activity happening in the video.

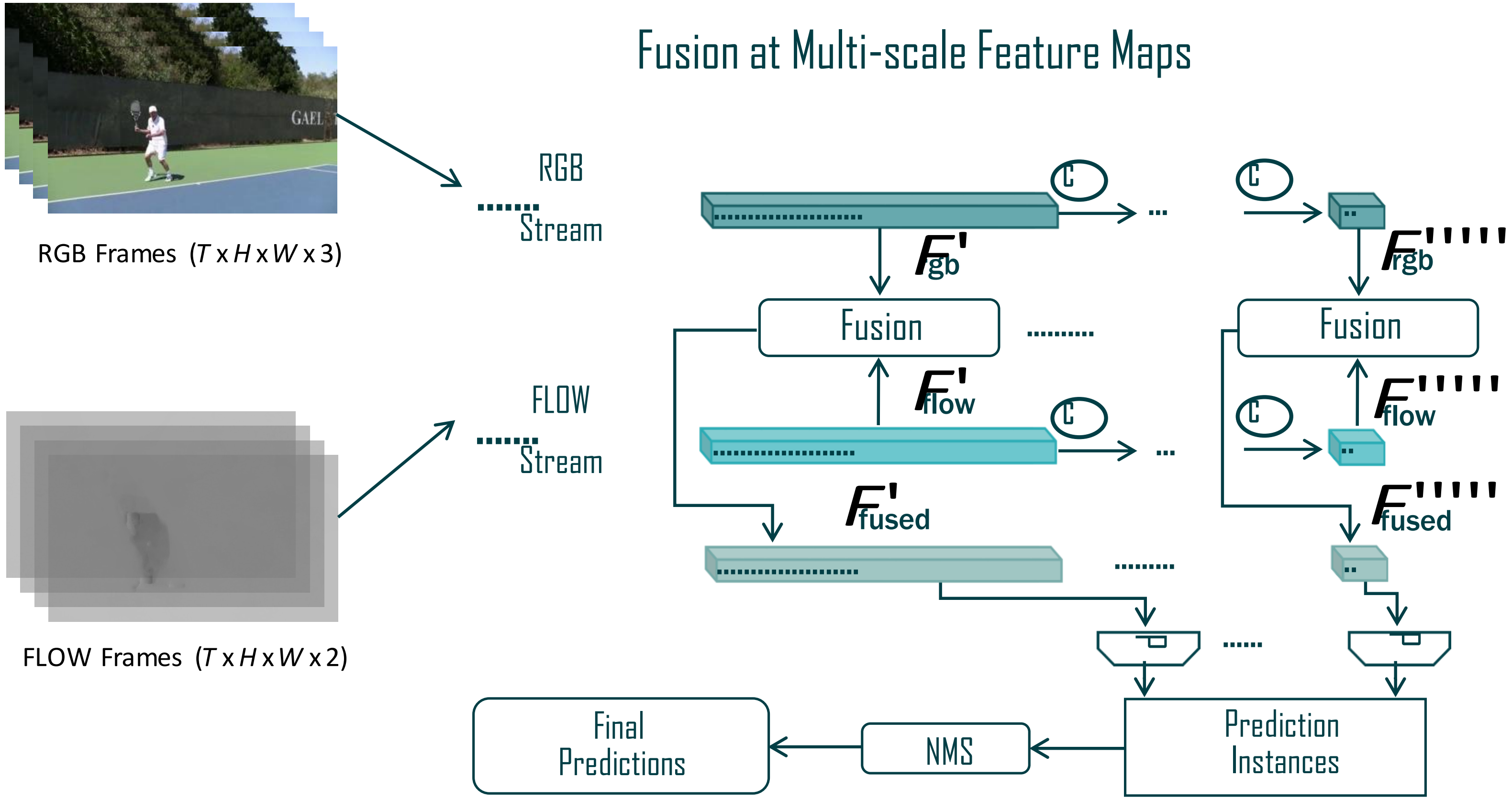


The THUMOS Dataset

Mid-level fusion at base feature map



Mid-level fusion at multi-scale



Merging feature maps: learnable fusion

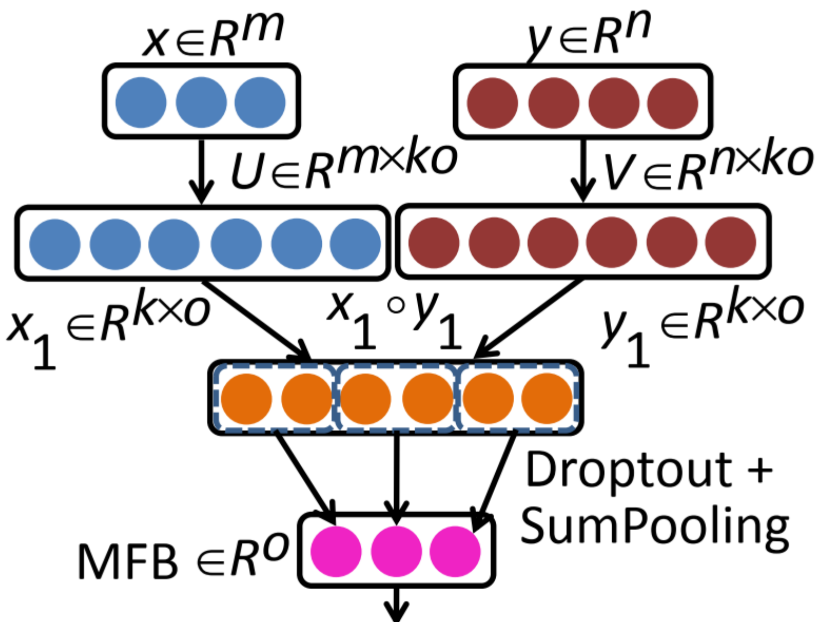
- Based on bilinear operation:

  $$\mathbf{y} = \mathbf{a}^T\mathbf{W}\mathbf{b} + \mathbf{k}$$

- Computational complexity reduced using Multi-modal Low-rank Bilinear Pooling (MLB):

  $$\mathbf{W} = \mathbf{U}\mathbf{V}^T$$

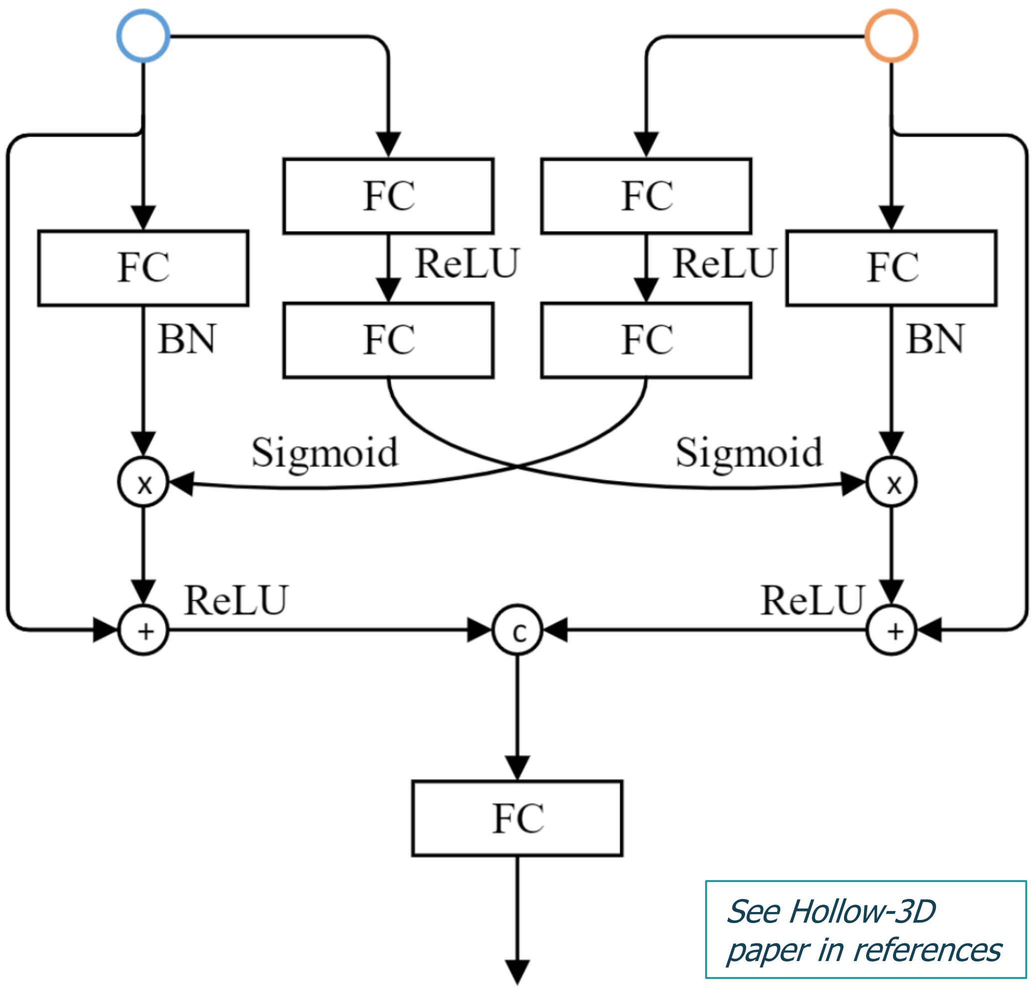
- And improved based on Multi-modal Factorized Bilinear Pooling (MFB)
- Most general fusion operator
  - The network basically learns how to best merge data
- Enable high interaction between input modalities



MFB

Merging sensor data: fusion subnetwork

- Introduces a light-weight gating mechanism for feature selection
- The fusion network benefits from the efficient interaction between sensor modalities
- Information from one branch guides discrimination in the other branch
  - This is an attention mechanism



See Hollow-3D paper in references

Mid-level fusion: activity recognition accuracy

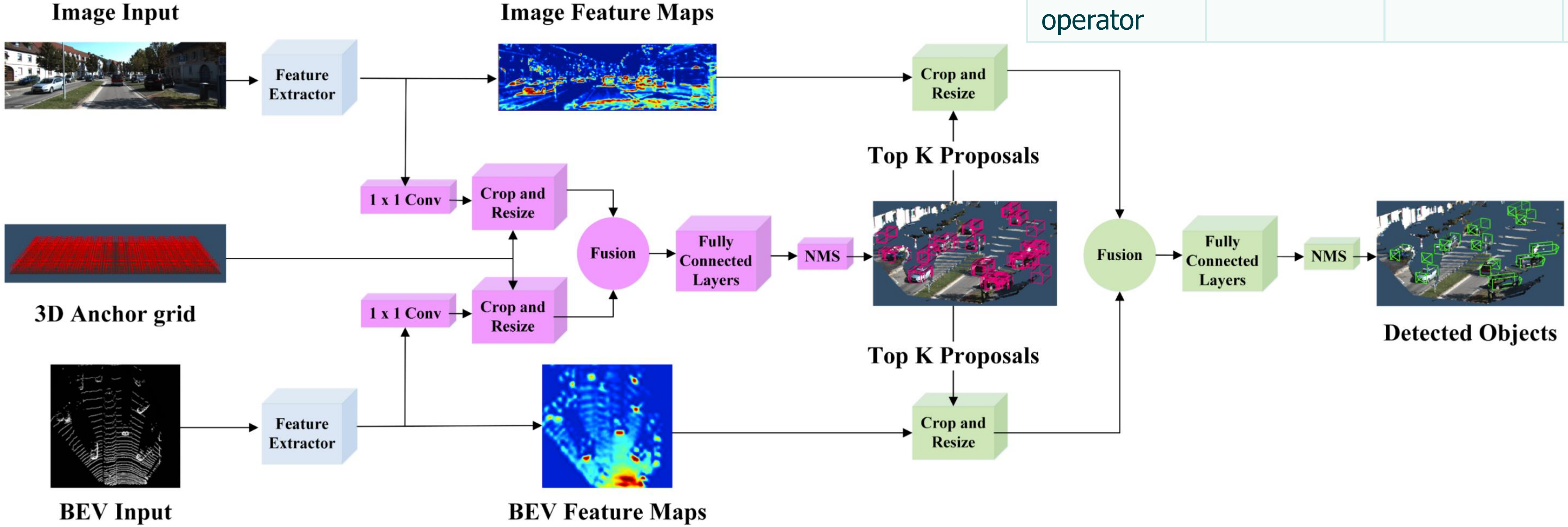
How to Fuse	***Where*** to Fuse		
	Mid-level		*Late*
	$\mathcal{F}'$	$\mathcal{F}^{MS}$	
Averaging			46.42
Sum	47.08	48.48	
Max	46.09	47.53	
Convolution	46.86	47.45	
MLB	46.65	47.77	
MFB	38.29	49.85	
MFB_new	37.47	**50.88**	

Base vs Multi-Scale Fusion

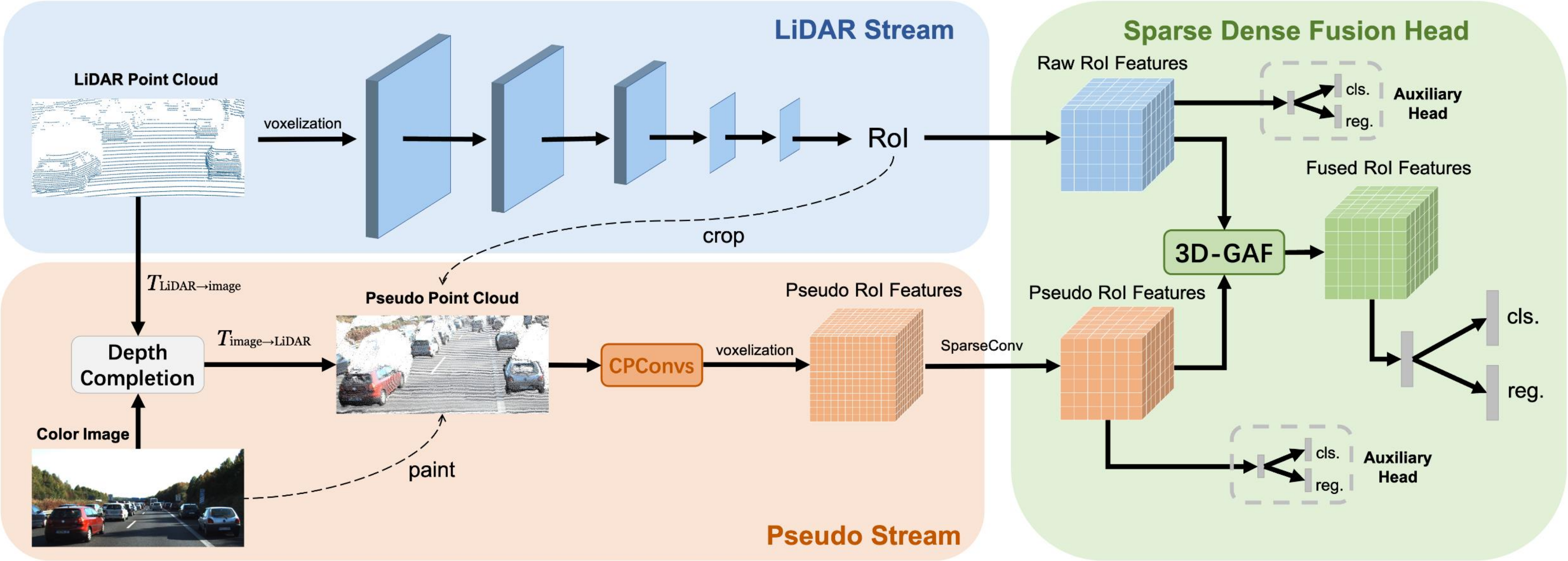
Another example: the AVOD architecture

- AVOD uses arithmetic mean for image/LiDAR fusion
- AVOD is a Region Proposal Network that includes 2 fusion steps
- Using MFB for fusion improves the results

Car detection	Easy	Moderate	Hard
Mean fusion operator	88.7	79.3	78.3
MFB fusion operator	89.7	80.2	79.1



One more (recent) example: Sparse Fuse Dense



- 3D Grid-wise Attentive Fusion
  - Sub-network fusion operator
- #1 on KITTI 3D car detection leader board

Method	Modality	BEV			
		mAP	Easy	Mod.	Hard
Voxel-RCNN [4]	LiDAR	89.94	94.85	88.83	86.13
SA-SSD [10]	LiDAR	90.67	95.03	91.03	85.96
SE-SSD [50]	LiDAR	91.41	**95.68**	91.84	86.72
EPNet [20]	LiDAR+RGB	88.79	94.22	88.47	83.69
3D-CVF [45]	LiDAR+RGB	88.51	93.52	89.56	82.45
CLOCs PVCas [25]	LiDAR+RGB	89.81	93.05	89.80	86.57
SFD (ours)	LiDAR+RGB	**91.44**	95.64	**91.85**	**86.83**

Conclusion

- Sensor fusion exploits the complementary characteristics of each sensor
  - Sensor fusion becomes particularly significant under adverse driving conditions
- Early fusion
  - In detection networks, often used to increase recall (the number of detected objects)
  - Relatively easy to train
- Late fusion
  - In detection networks, often used to increase precision
  - Multiple networks to be trained
- Mid-level fusion
  - Potentially optimal performances
  - Particularly adapted to heterogenous sensors
  - Could be very difficult to train

For more information

Resources…

Radar/Stereo dataset

https://www.site.uottawa.ca/research/viva/projects/raddet/index.html

THUMOS Dataset

http://crcv.ucf.edu/THUMOS14/home.html

2022 Embedded Vision Summit

See us at the Synopsys booth –
Embedded radar demo

References

- Qi C.R. et al. (2018) Frustum PointNets for 3D Object Detection From RGB-D Data, CVPR18.
- Rahman Md A, Laganiere R. (2020) Mid-level fusion for end-to-end temporal activity detection in untrimmed videos, BMVC
- Pfeuffer A., Dietmayer K. (2018). Optimal Sensor Data Fusion Architecture for Object Detection in Adverse Weather Conditions, Int. Conf. on Information Fusion.
- Deng J. et al. (2021) From Multi-View to Hollow-3D, IEEE trans. Circuits and Systems for Video Tech.
- Ku J. et al. (2018) AVOD Joint 3D Proposal Generation and Object Detection from View Aggregation, IROS.
- Wu X. et al. (2022) Sparse Fuse Dense, arXiv.