Modeling Traffic Scenes for Intelligent Vehicles using CNN-based Detection and Orientation Estimation

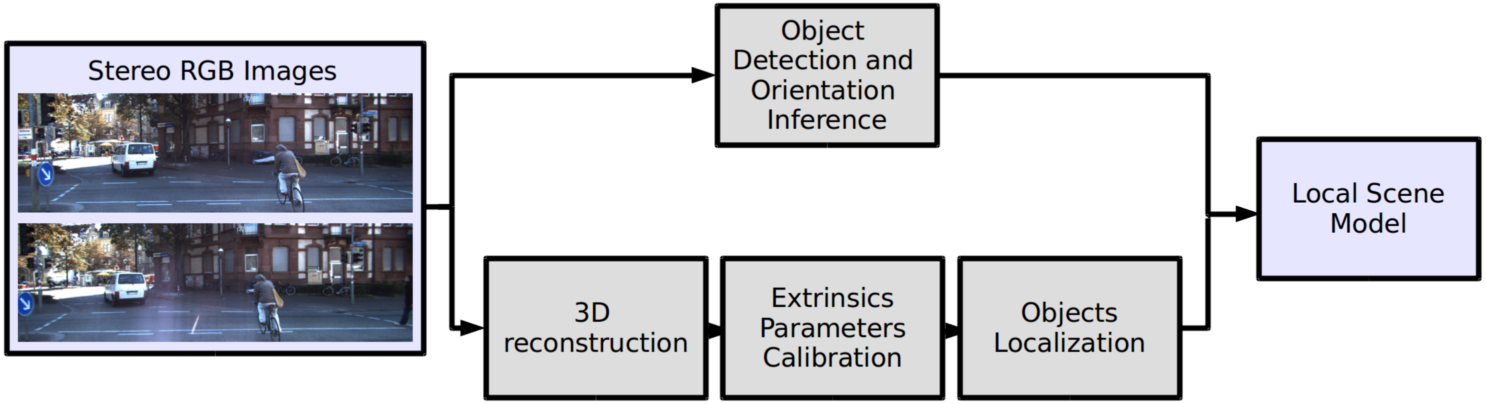
Carlos Guindel, David Martín and José María Armingol

Intelligent Systems Laboratory (LSI) · Universidad Carlos III de Madrid

Sevilla · 23 November 2017

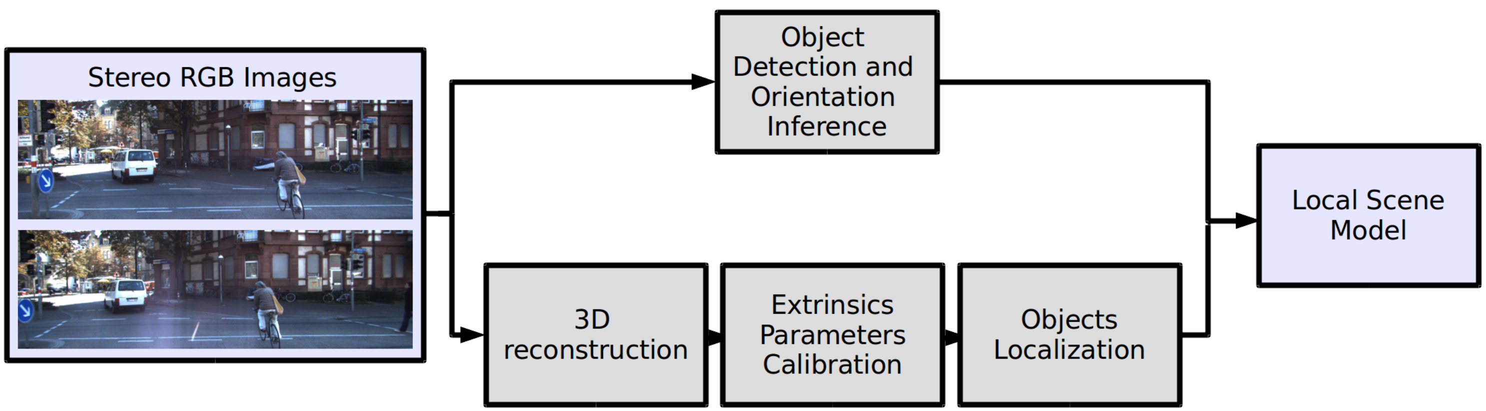
Agenda

1. Introduction
2. Obstacle detection
3. Scene modeling
4. Results
5. Conclusion



Agenda

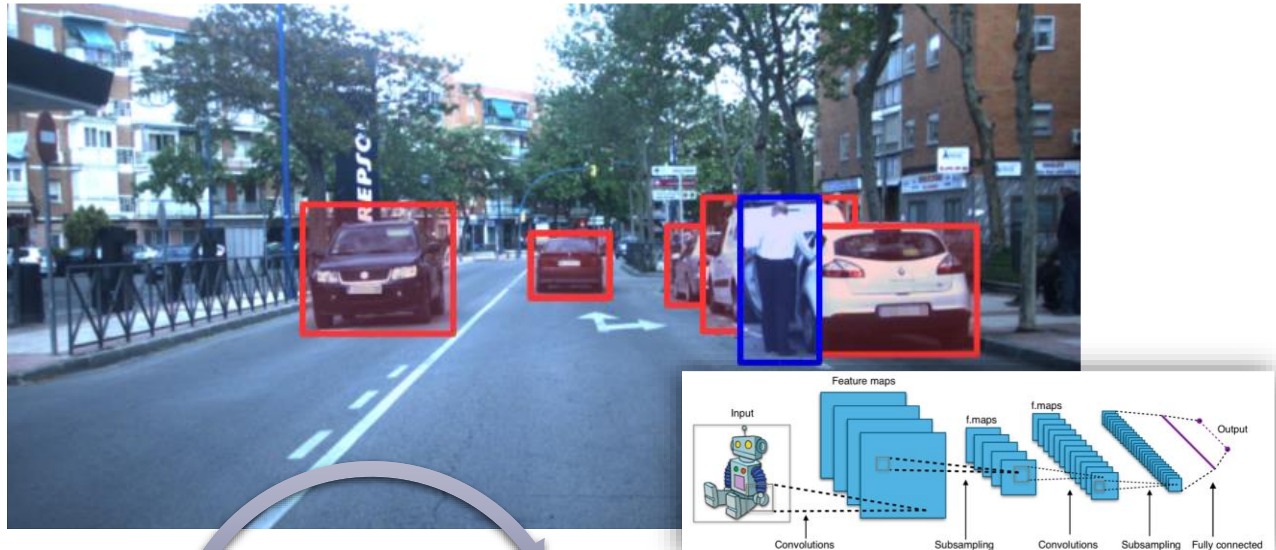
1. Introduction
2. Obstacle detection
3. Scene modeling
4. Results
5. Conclusion



Introduction

Automated vehicles

- Highly dynamic, semi-structured environments
- They have to handle complex situations



Obstacle detection

- A basic requirement for driving tasks

Classification

- An accurate estimation of the class is essential

Vision-based approaches

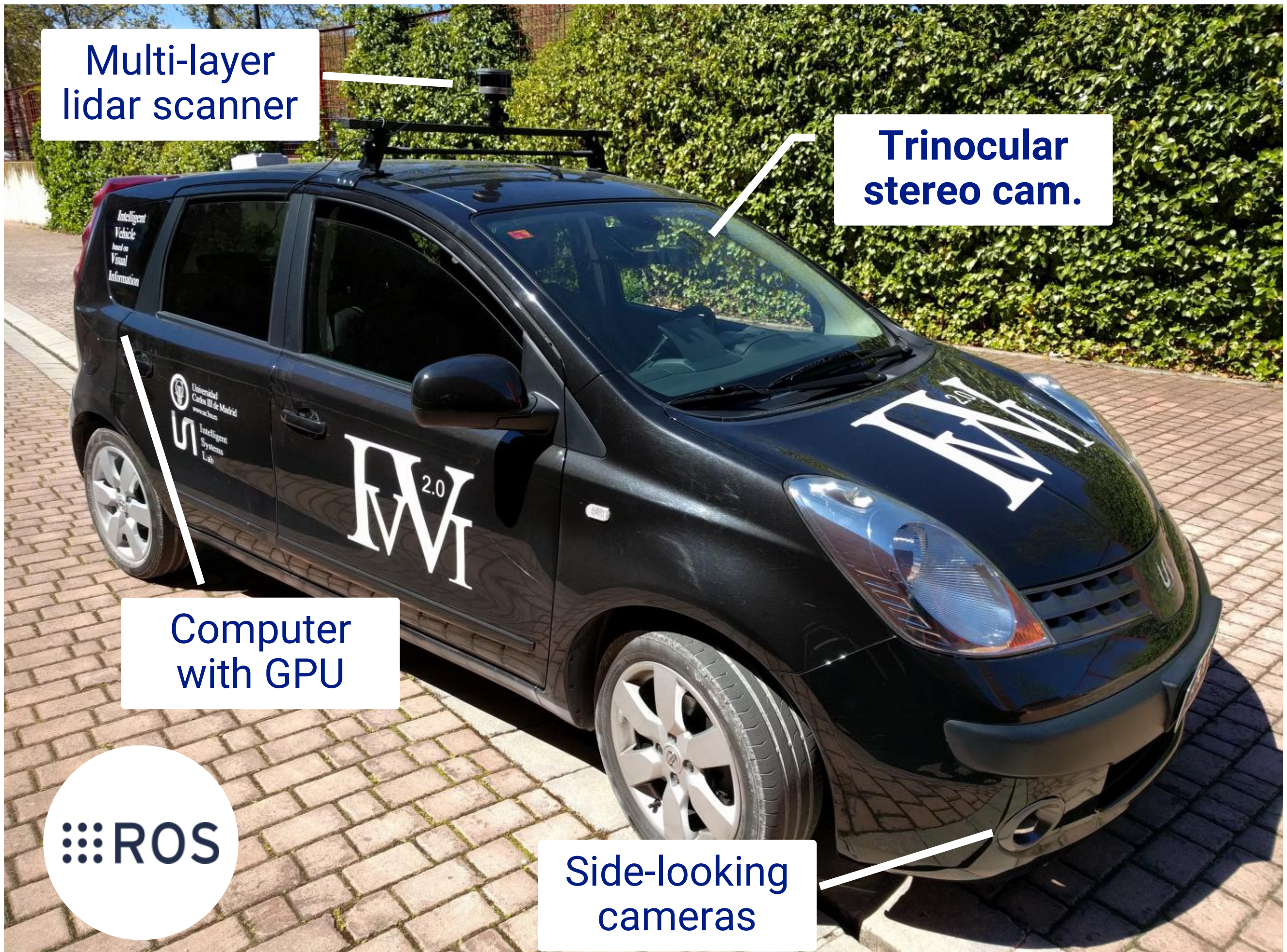
- Close-to-market assemblies
- Rich data source

Convolutional Neural Networks

- Feature learning
- The new paradigm in computer vision

IVVI 2.0 project

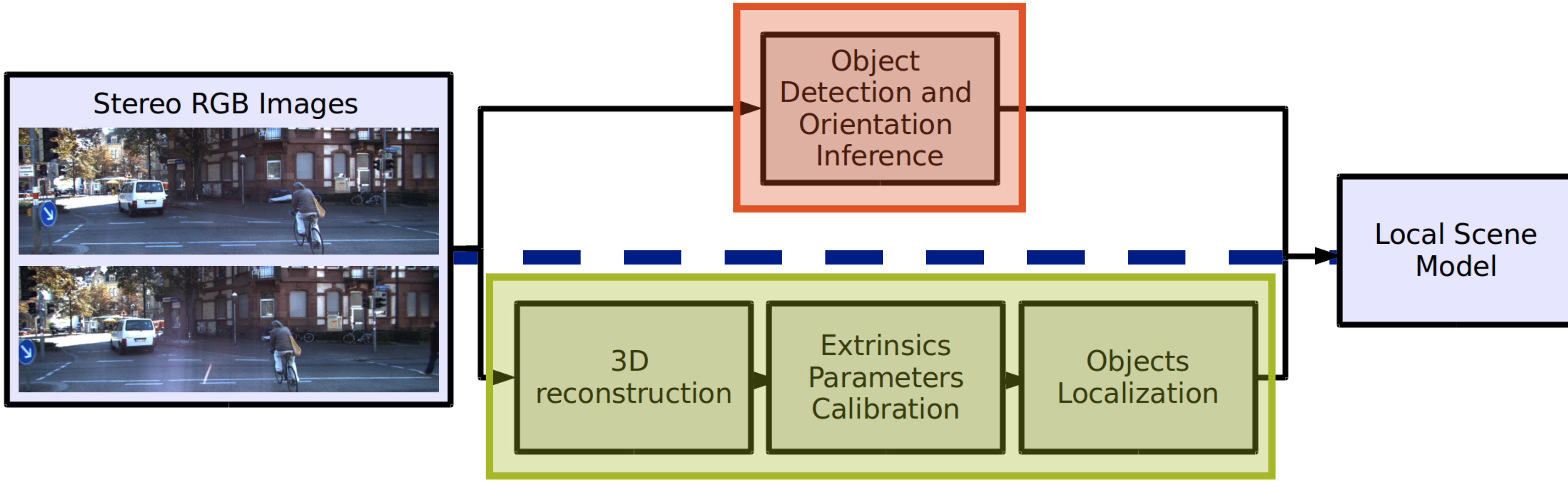
INTELLIGENT VEHICLE BASED ON VISUAL INFORMATION 2.0



+info:
uc3m.es/islab

System overview

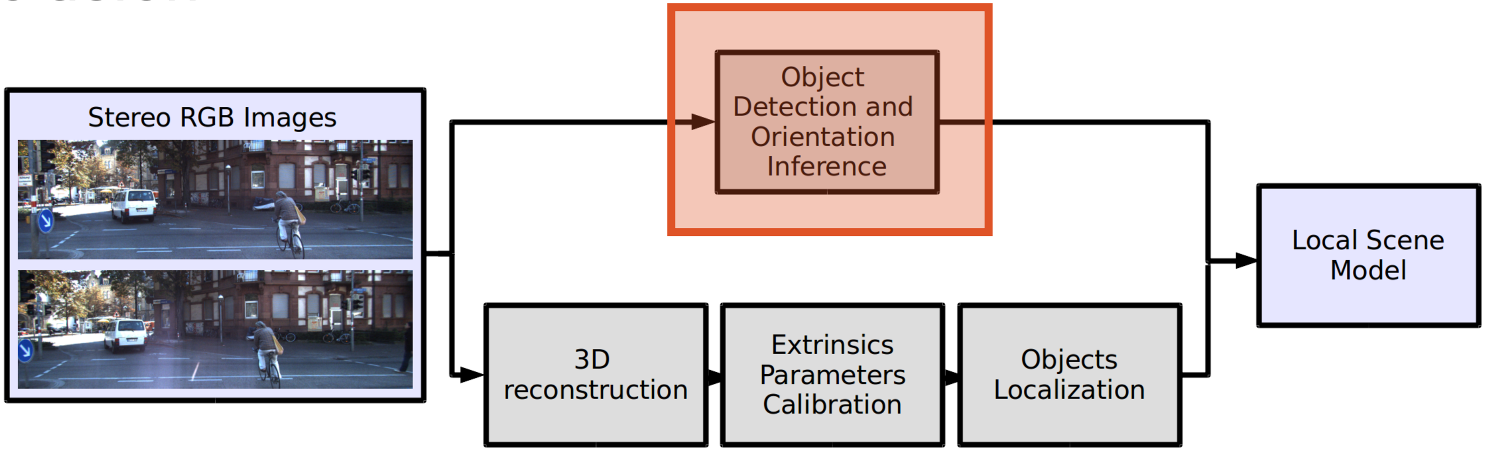
- Two main branches intended to run in parallel
- Obstacle detection
  - Features are extracted exclusively from the left stereo image



- Scene modeling
  - Stereo-based 3D reconstruction & flat-ground assumption

Agenda

1. Introduction
2. **Obstacle detection**
3. Scene modeling
4. Results
5. Conclusion



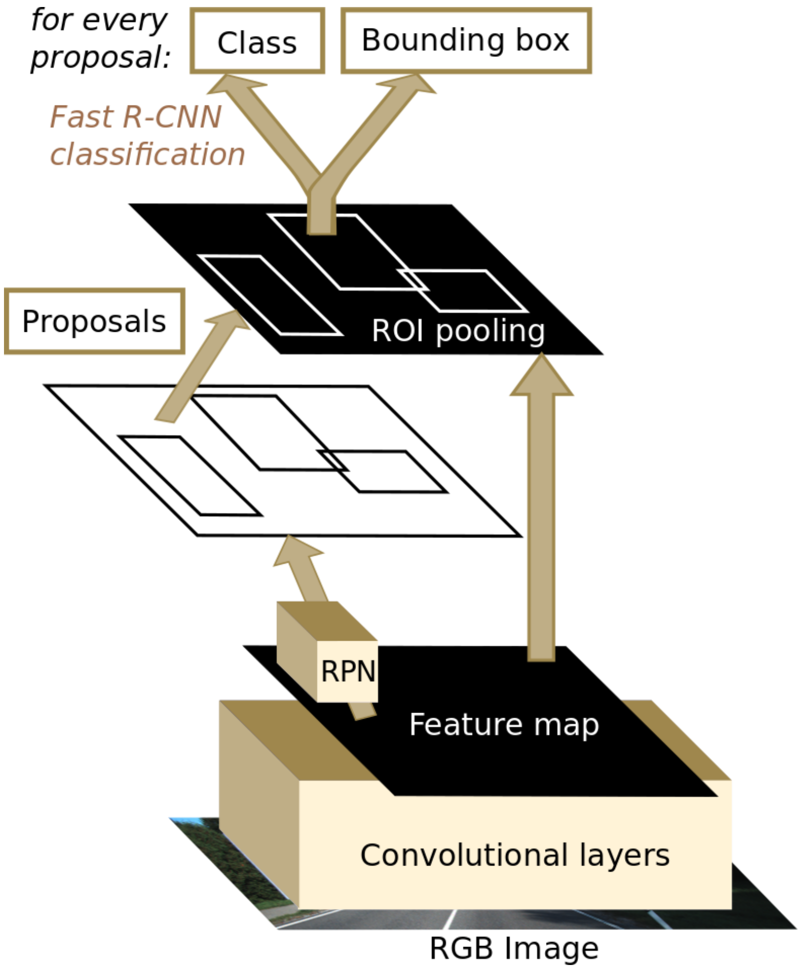
Faster R-CNN framework

Parameters are learned through a **multi-task loss**

Conv. features in these regions are pooled for **classification**

A **RPN** generates proposals wrt. a fixed set of anchors

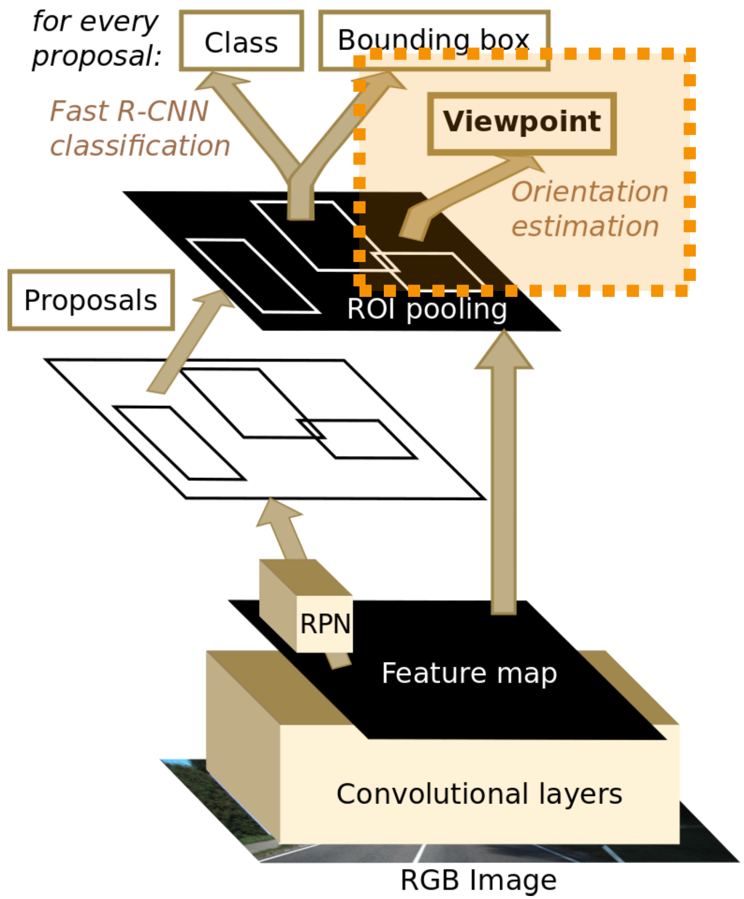
Convolutional features computed **only once** per image



S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 6, pp. 1137–1149, 2016.

Viewpoint estimation

- Faster R-CNN framework was modified to introduce **viewpoint inference**

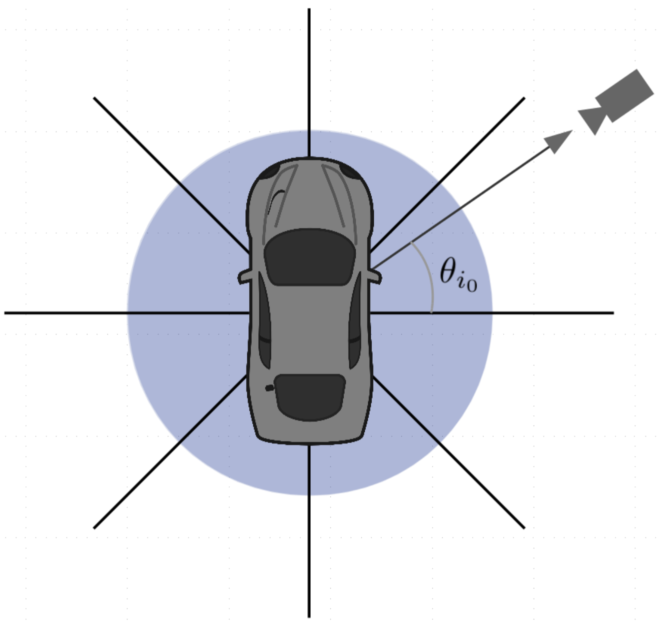


C. Guindel, D. Martin, and J. M. Armingol, "Joint object detection and viewpoint estimation using CNN features," in Proc. of the IEEE International Conference on Vehicular Electronics and Safety (ICVES), 2017, pp. 145–150.

Discrete viewpoint inference

N_b **angle bins** $\Theta_i \dots \Theta_{N_b}$

$N_b = 8$



- Every object is assigned a bin

Training: $\theta_{i_0} \rightarrow \Theta_i$

$$\Theta_i = \left\{ \theta \in [0, 2\pi) \;\middle|\; \frac{2\pi}{N_b} \cdot i \leq \theta < \frac{2\pi}{N_b} \cdot (i+1) \right\}$$

- Inference gives a categorial distribution

Inference output: $\mathrm{r} \in \Delta^{N_b - 1}$

$$\Delta^N = \left\{ x \in \mathbb{R}^{N+1} \;\middle|\; \sum_{i=1}^{N+1} x_i = 1 \;\wedge\; \forall i : x_i \geq 0 \right\}$$

r

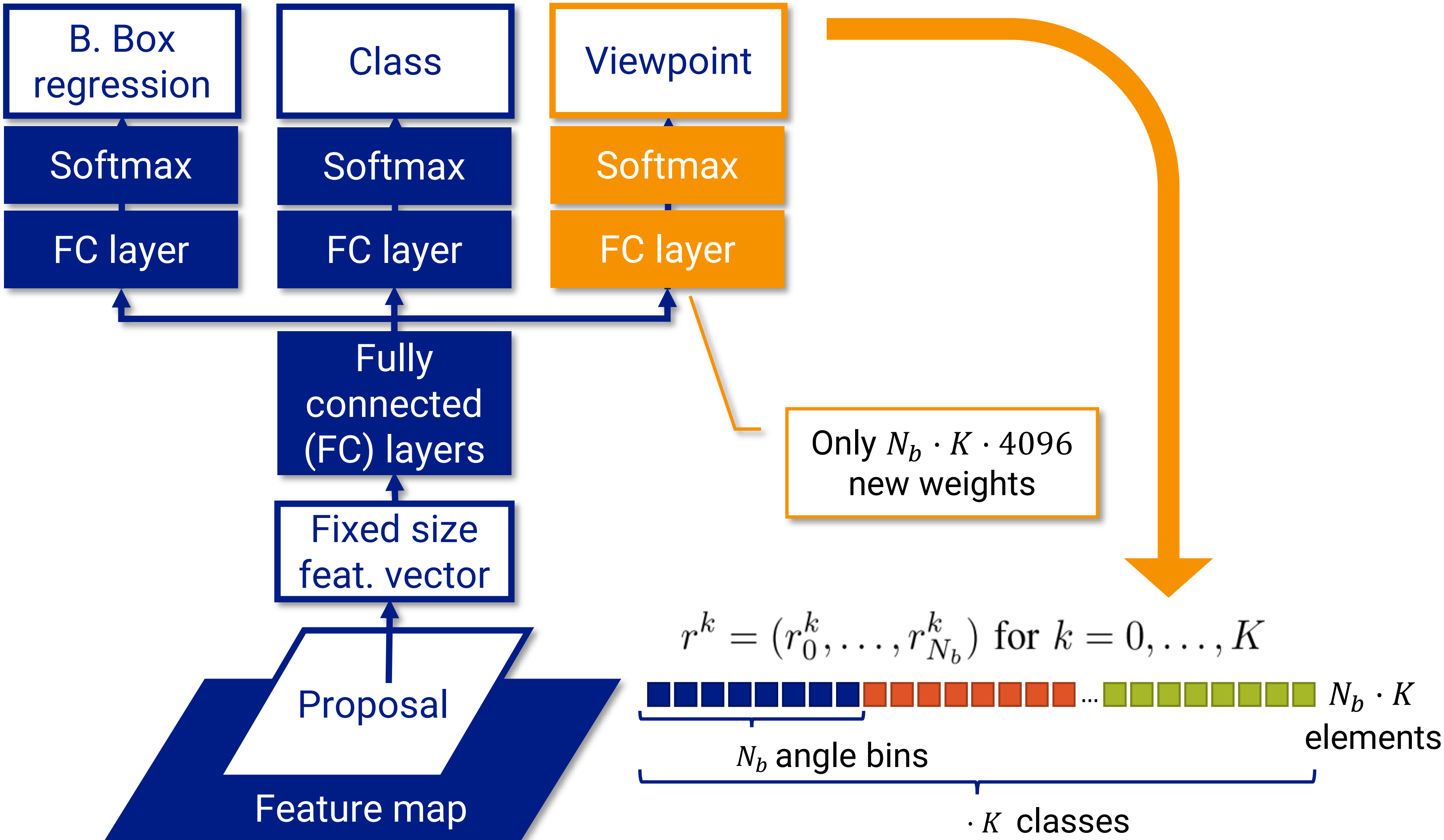
$$i^* = \arg\max_i (x_i)$$

Elements of r

Final estimation: $\Theta_{i^*} \rightarrow \hat{\theta}$

$$\hat{\theta} = \frac{\pi(2i^* + 1)}{N_b}$$

Joint detection and viewpoint estimation



Loss function and training

- Unweighted muli-task loss with **five** components

Faster R-CNN loss

Logistic loss
for RPN objectness

+

Smooth-L1 loss
for RPN b.box regression

+

Logistic loss
for class

+

Smooth-L1 loss
for b.box regression

+

Logistic loss
for viewpoint estimation

$$r^k = (r_0^k, \ldots, r_{N_b}^k) \text{ for } k = 0, \ldots, K$$

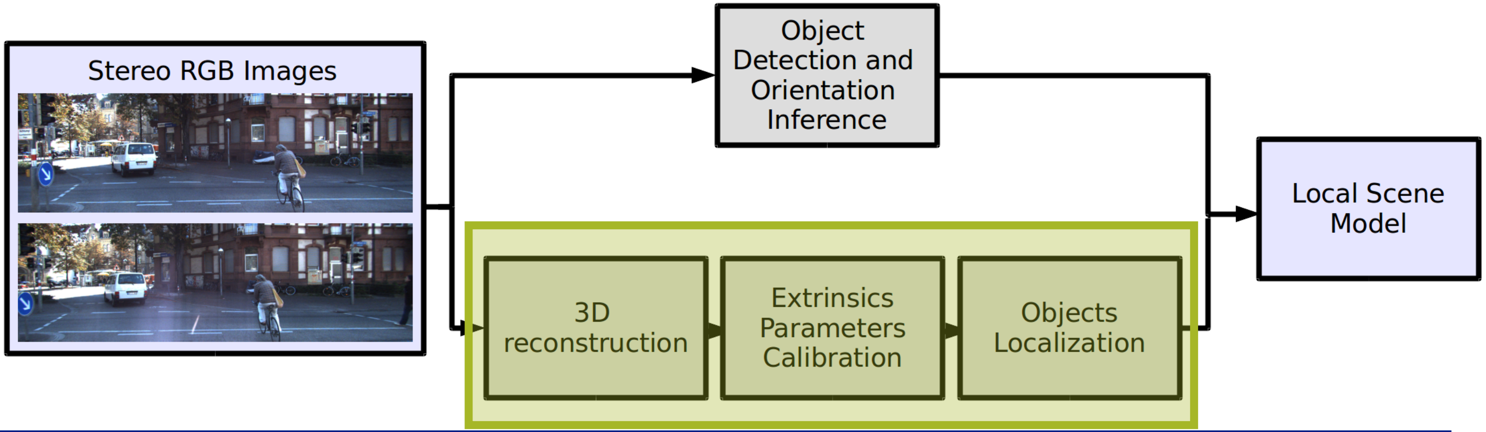
N_b angle bins of the ground truth class

$$L_v = \frac{1}{N_B} \sum_{j \in B} L_{cls}(\boldsymbol{r}_j^{u^*}, b_j^*)$$

Normalized on the batch

Ground-truth angle bin

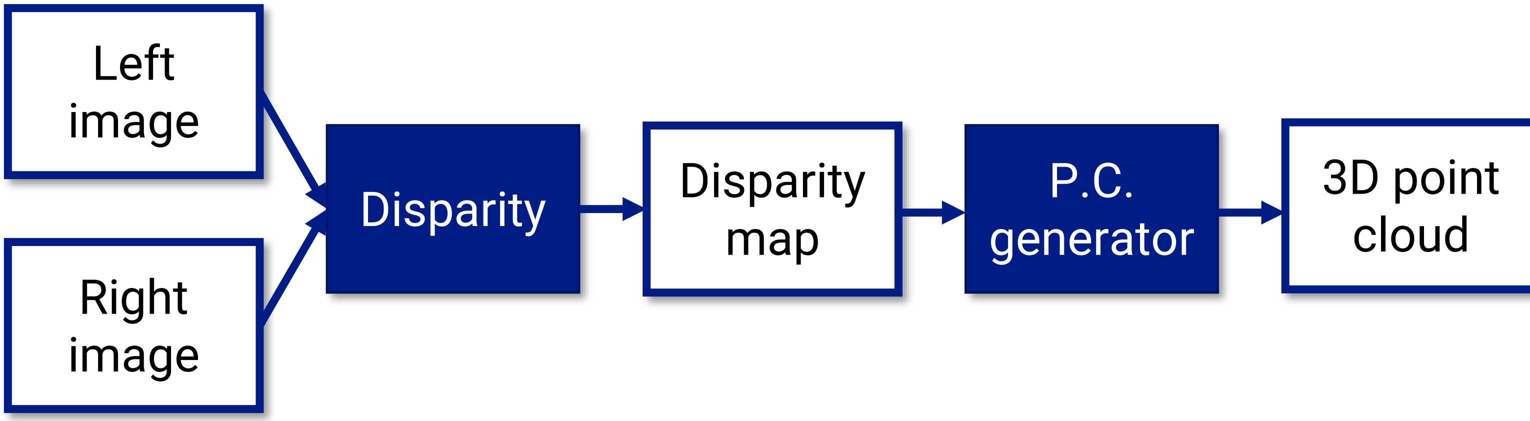
Agenda



Stereo RGB Images
Object Detection and Orientation Inference
3D reconstruction
Extrinsics Parameters Calibration
Objects Localization
Local Scene Model

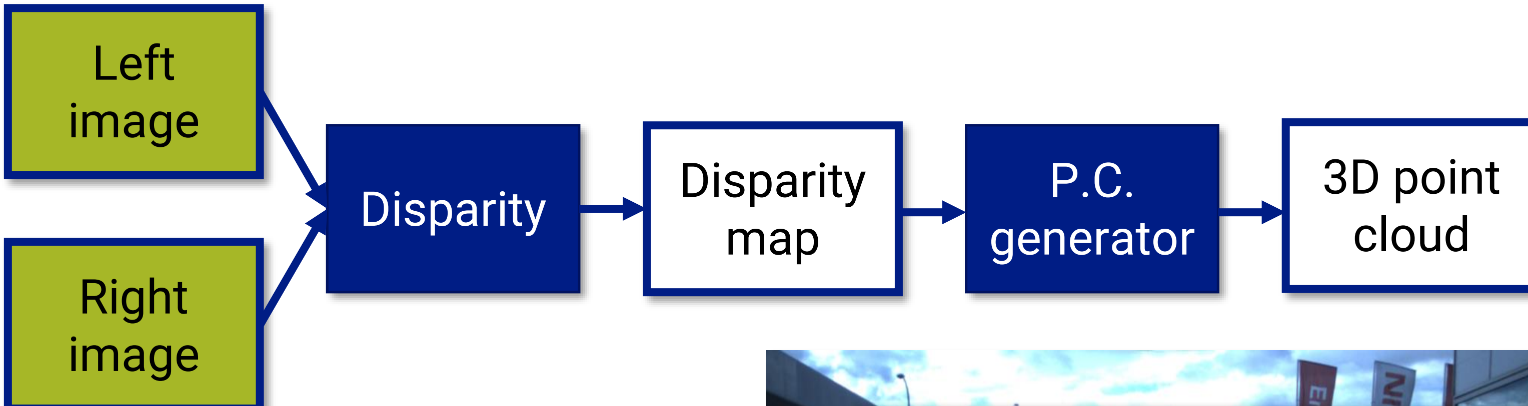
Scene modeling

3D Reconstruction

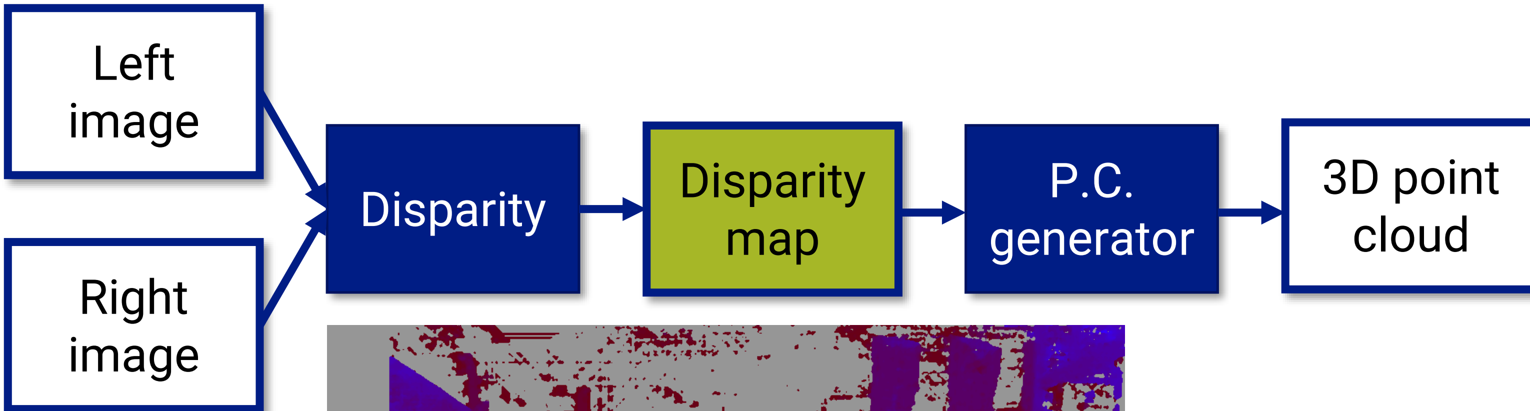


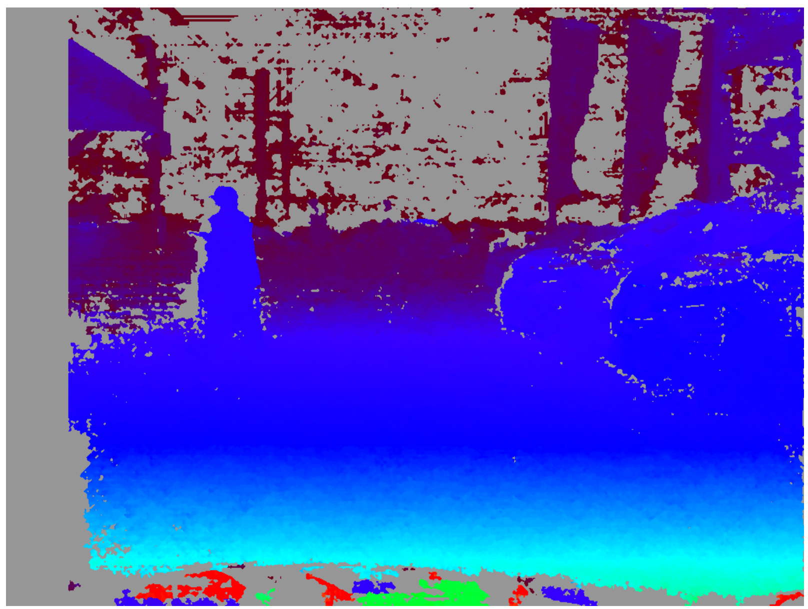
Scene modeling

3D Reconstruction



3D Reconstruction

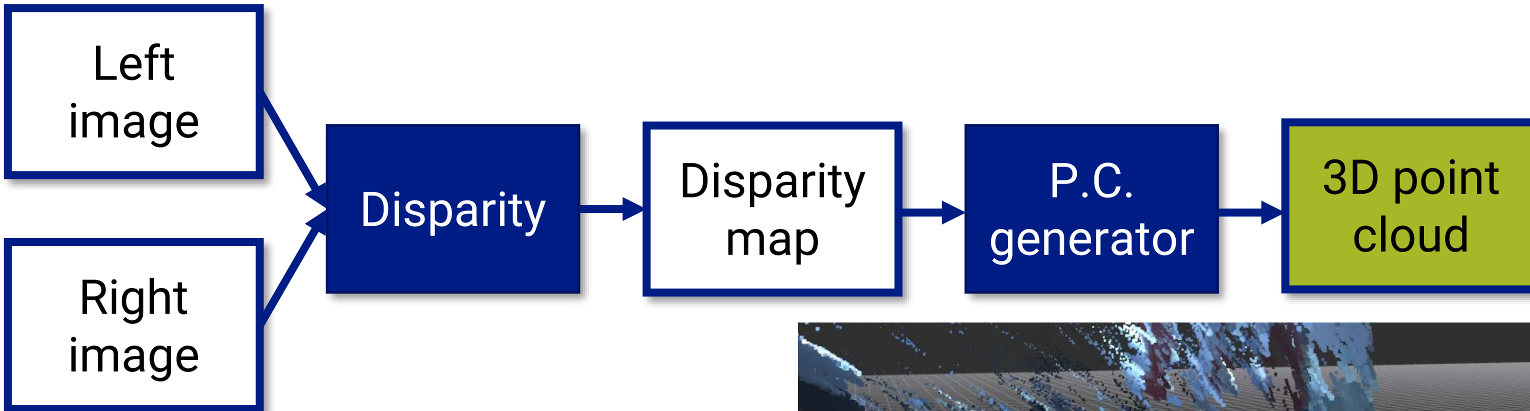




SGM stereo matching
Suitable for environments with lack of texture, illumination changes, etc.

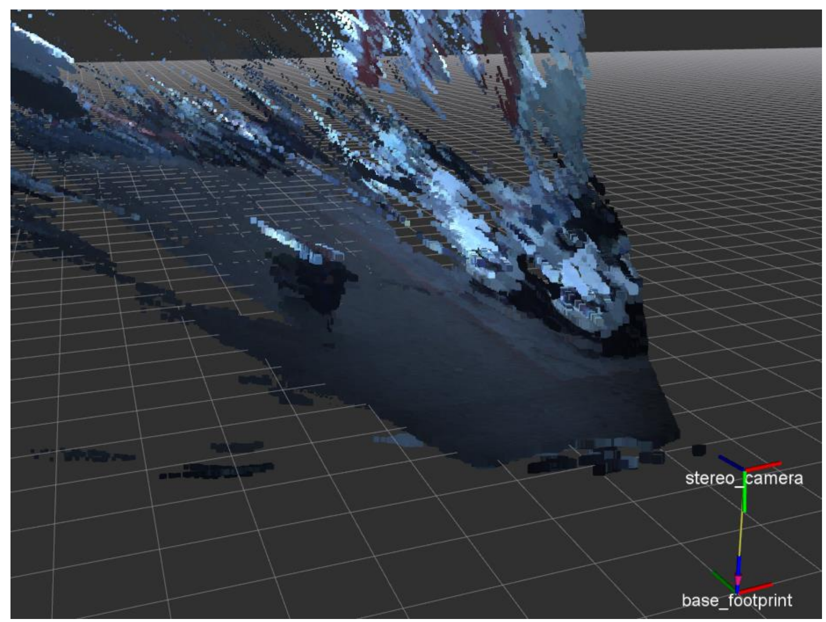
Scene modeling

3D RECONSTRUCTION



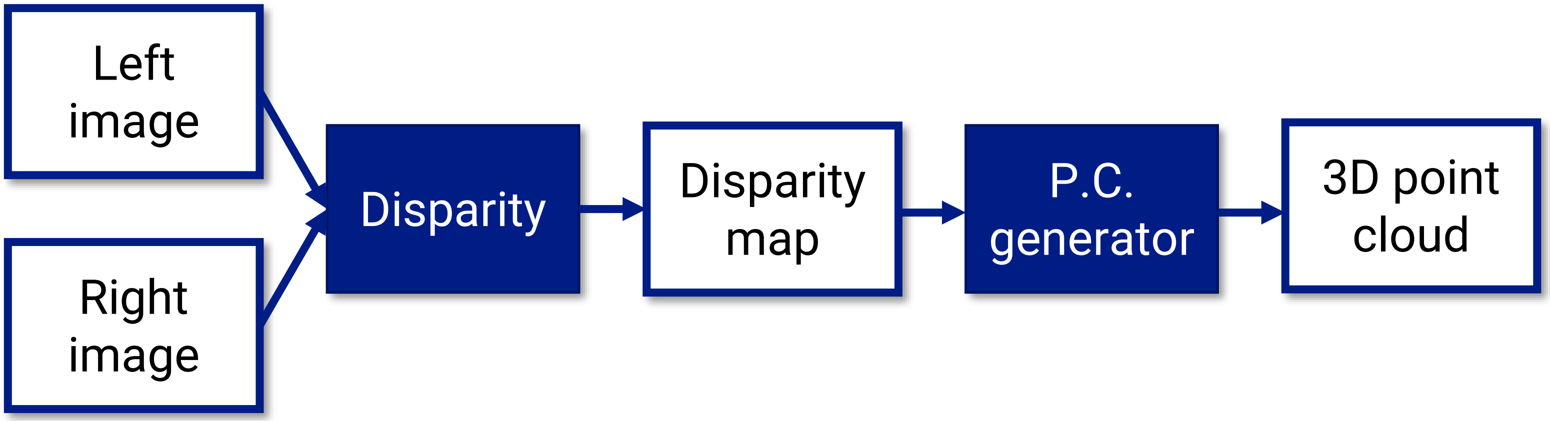
Pin-hole + disparity

We build a XYZRGB cloud from the left image and the disparity map

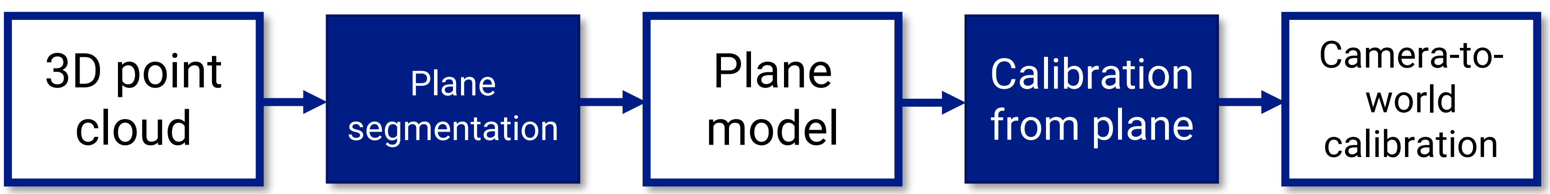


Scene modeling

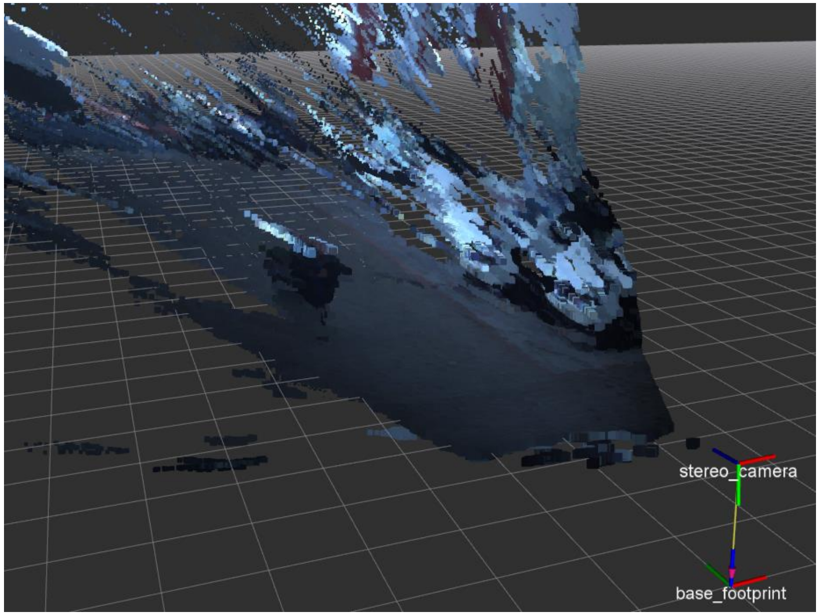
3D Reconstruction



Extrinsic Parameters Auto-calibration



Scene modeling



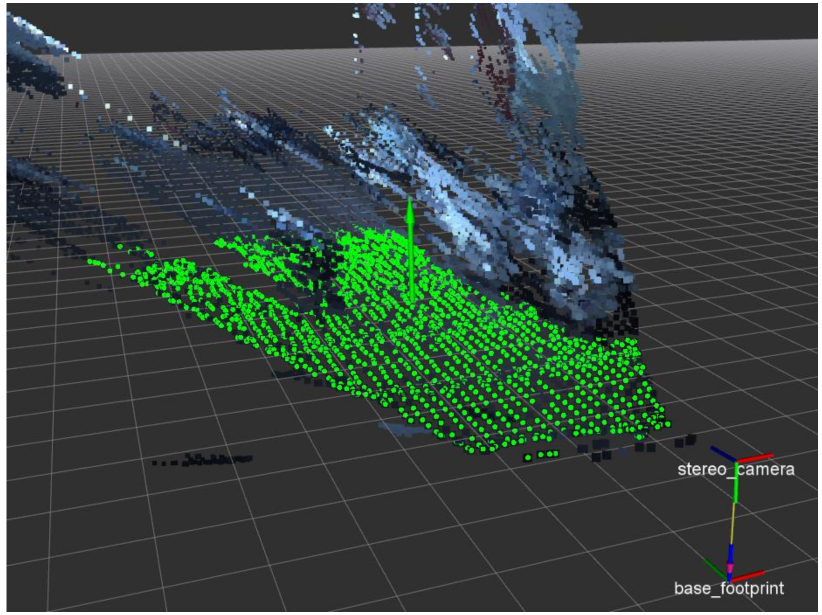
Voxel grid dowsampling
The cloud from the 3D reconstruction pipeline is downsampled (grid size: 20 cm)

EXTRINSIC PARAMETERS AUTO-CALIBRATION

3D point cloud → Plane segmentation → Plane model → Calibration from plane → Camera-to-world calibration

Scene Modeling

…Pass through filters
Vertical axis: 0-2 m
Depth axis: 0-20 m

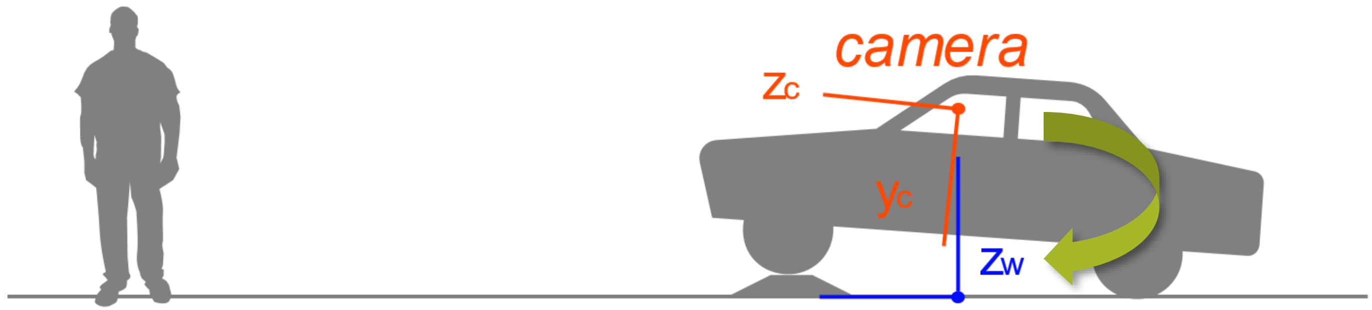


Planar segmentation
Using RANSAC with a 10 cm threshold, and a small angular tolerance.

EXTRINSIC PARAMETERS AUTO-CALIBRATION

3D point cloud → Plane segmentation → Plane model → Calibration from plane → Camera-to-world calibration

Scene modeling



Plane: $ax_c + by_c + cz_c + d = 0$

pitch

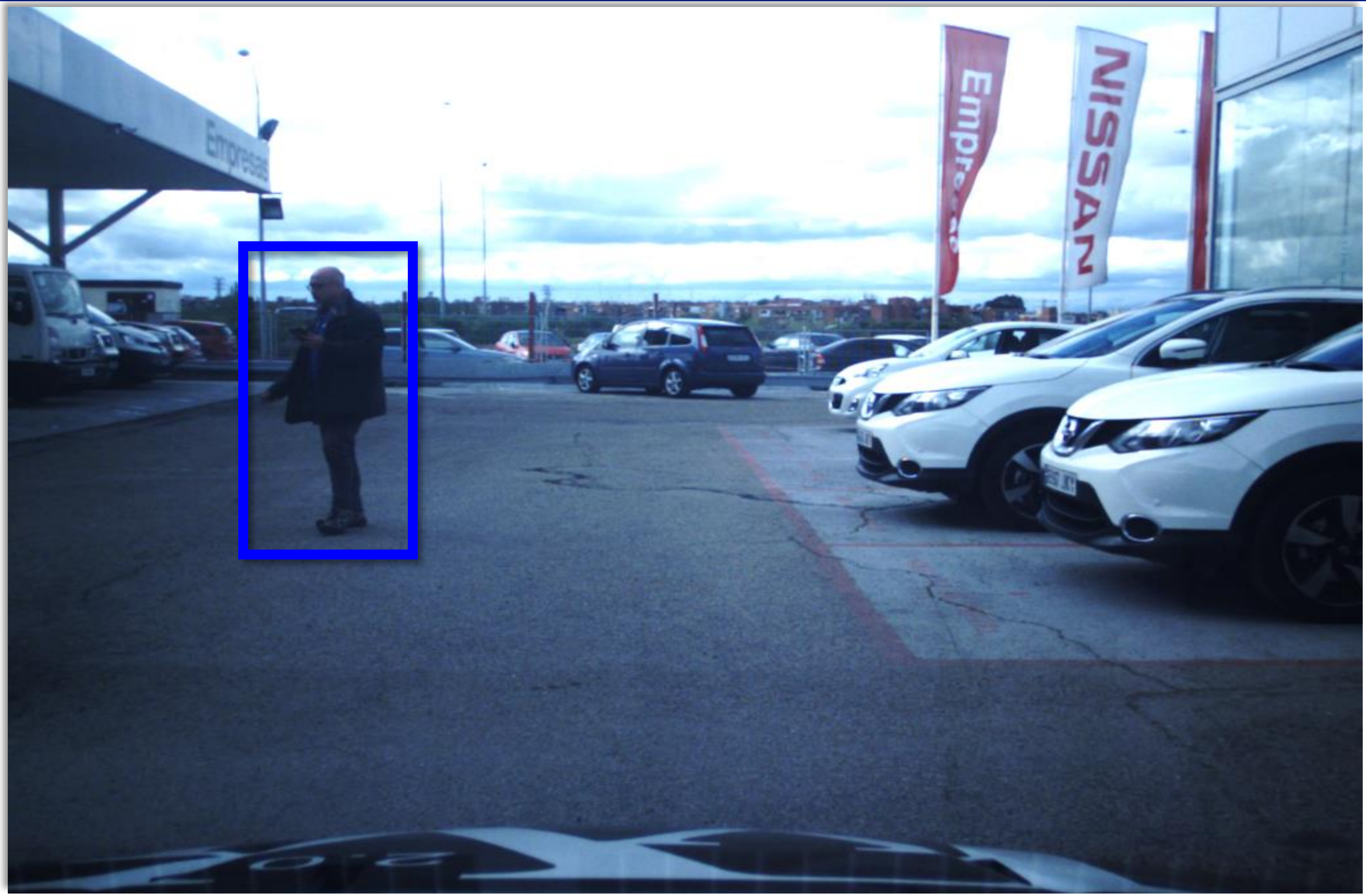
roll

$$\psi = \arcsin(a) \qquad \phi = \arctan\left(\frac{-c}{b}\right) \qquad h = d$$

EXTRINSIC PARAMETERS AUTO-CALIBRATION

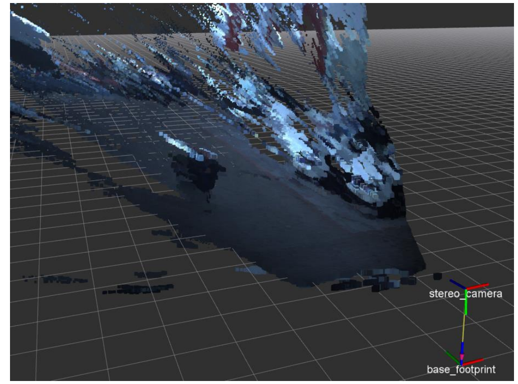
3D point cloud → Plane segmentation → Plane model → Calibration from plane → Camera-to-world calibration

Object localization



Object localization

Organized pointcloud



Filtered cloud
Without points belonging to the ground or too close to the camera

Object ROI for localization
11 central rows of the object's bounding box

Coordinates on the world's XY plane for every point

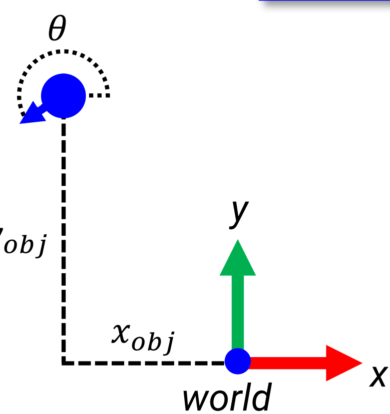
$$\boldsymbol{x} = (x, y, z)$$

$$\boldsymbol{x}_{obj} = median(\boldsymbol{x})$$

$$\theta_{obj} = \alpha - \text{atan2}(y_{obj} - x_{obj})$$

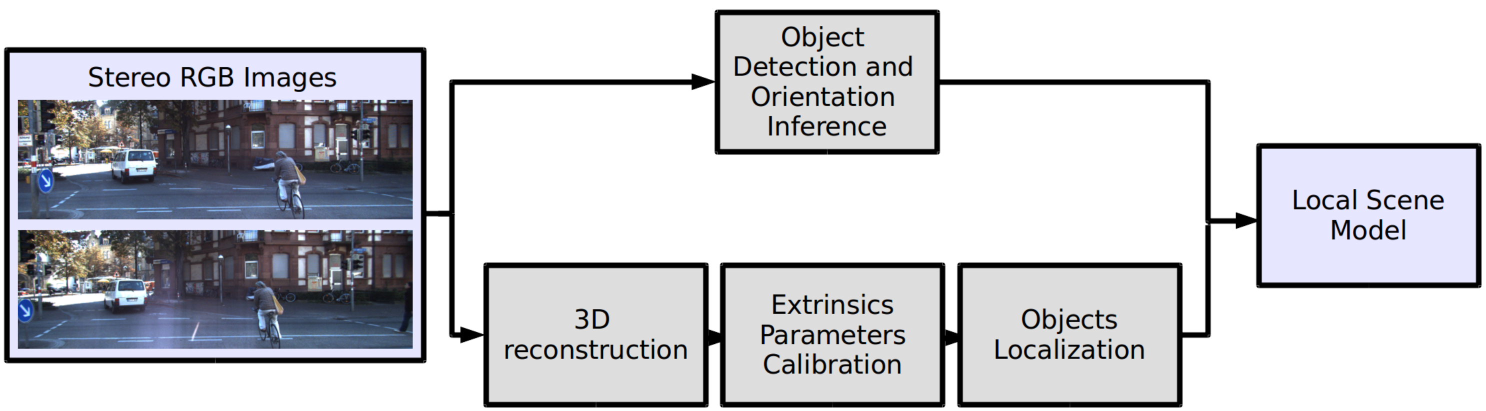
yaw

Top-down view

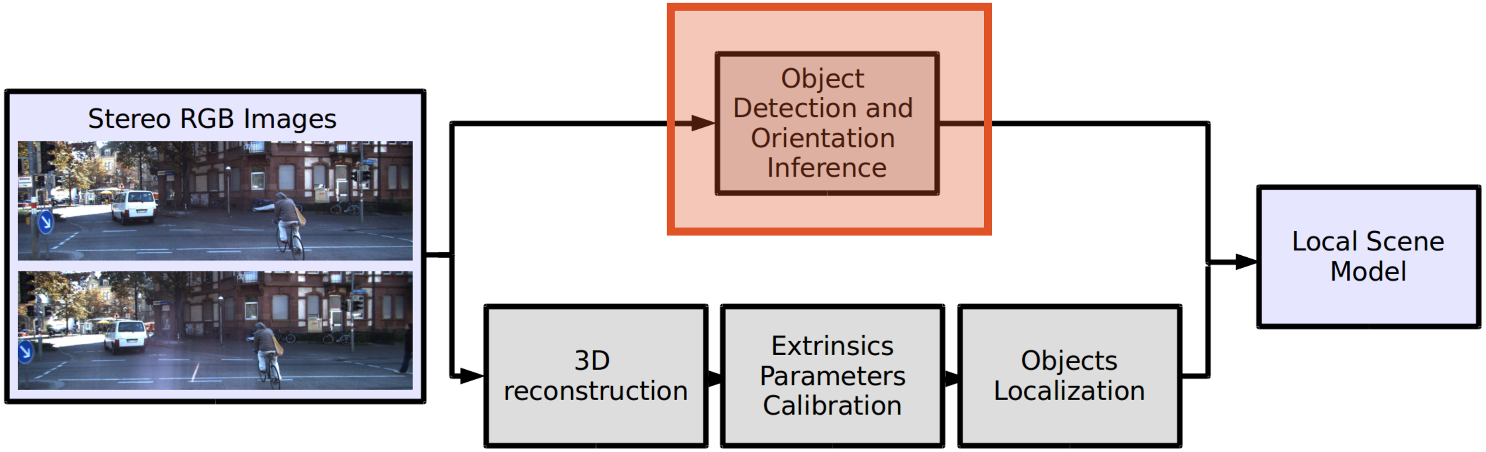


Agenda

1. Introduction
2. Obstacle detection
3. Scene modeling
4. **Results**
5. Conclusion



Results: Detection and viewpoint estimation



- KITTI Object Detection Benchmark
  - 5,576 images for training and 2,065 for validation
  - Labels for class and orientation available
- Evaluation metric
  - Average Orientation Similarity (AOS)

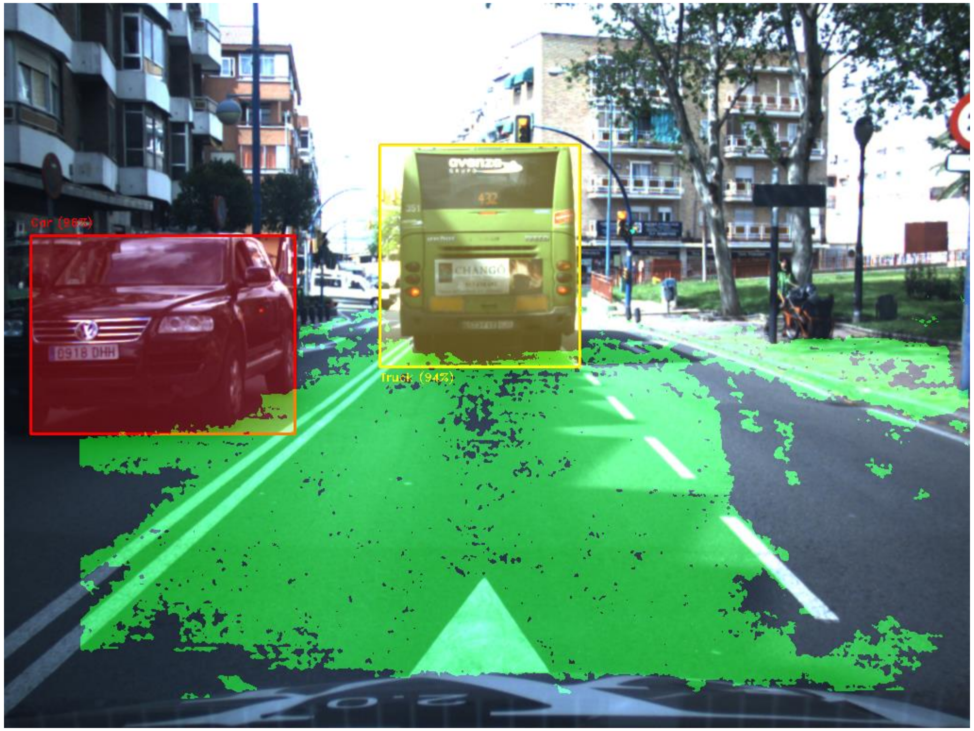
$$AOS = \frac{1}{11} \sum_{r \in \{0, 0.1, .., 1\}} \max_{\tilde{r}:\tilde{r} \geq r} s(\tilde{r}) \qquad s(r) = \frac{1}{|\mathcal{D}(r)|} \sum_{i \in \mathcal{D}(r)} \frac{1 + \cos \Delta_\theta^{(i)}}{2} \delta_i$$

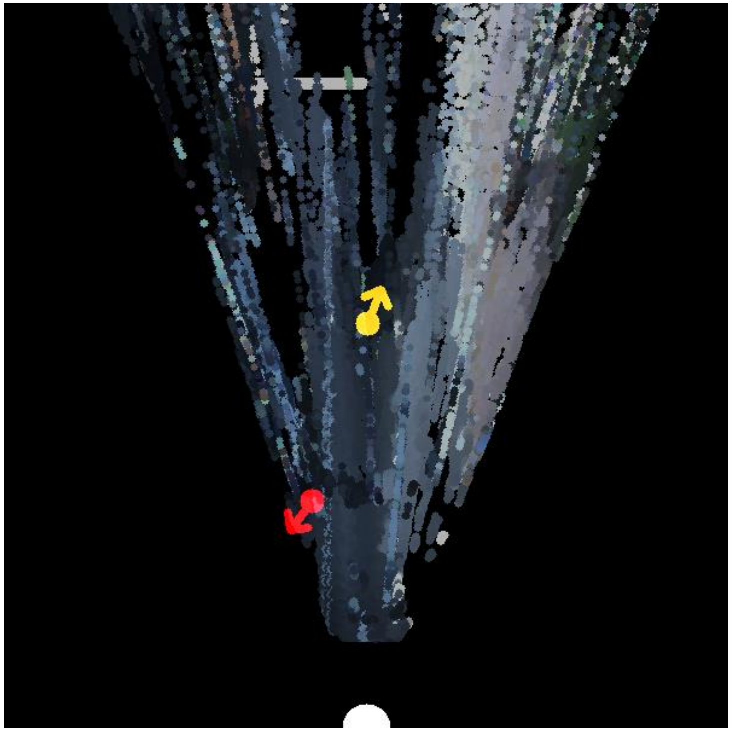
Results: Detection and viewpoint estimation

- Two different architectures:
  - ZF (lightweight) and VGG 16-layer (more complex)
- Three different scales (height in pixels):
  - 375, 500, 625

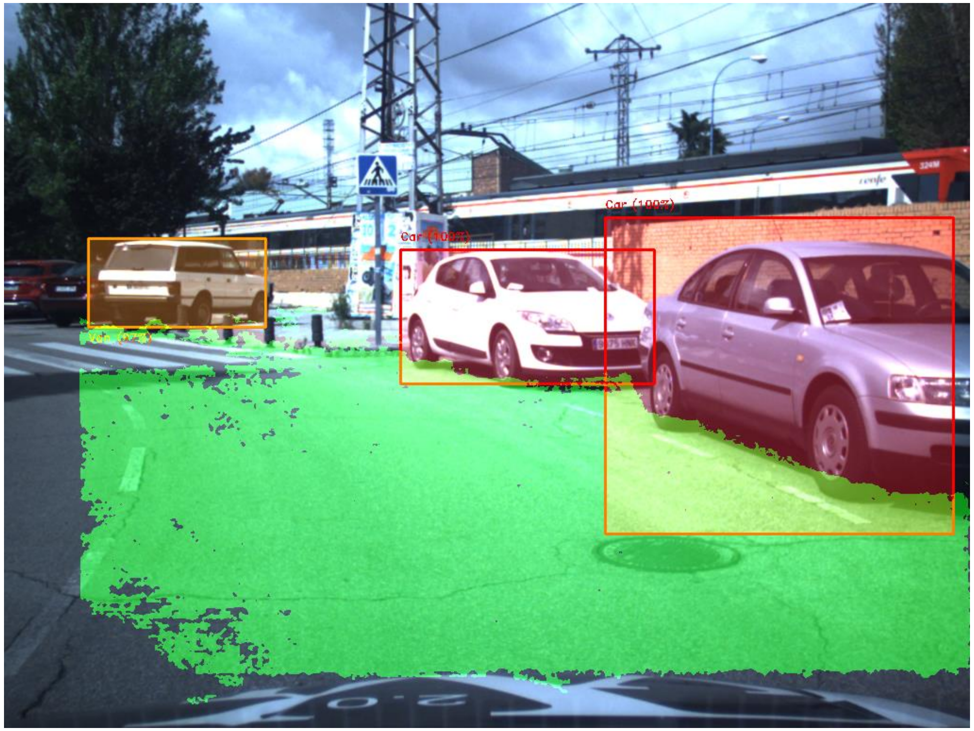
Net	Scale	Car	Pedest.	Cyclist	Van	Truck	mean	Time (ms)
ZF	375	44.2	35.6	16.1	8.5	3.2	21.5	46
	500	52.7	43.7	18.4	12.9	3.5	26.2	73
	625	51.6	40.7	22.7	15.1	5.3	27.1	90
VGG	375	64.8	54.7	25.0	22.9	8.5	35.2	79
	500	**74.7**	**61.0**	**33.0**	**30.0**	**12.1**	**42.2**	**112**
	625	75.7	60.9	35.2	31.1	15.4	43.7	144
Top-performing comparable method in the KITTI ranking		88,43	66,28	63,41	N.A.	N.A.	N.A.	2 sec.

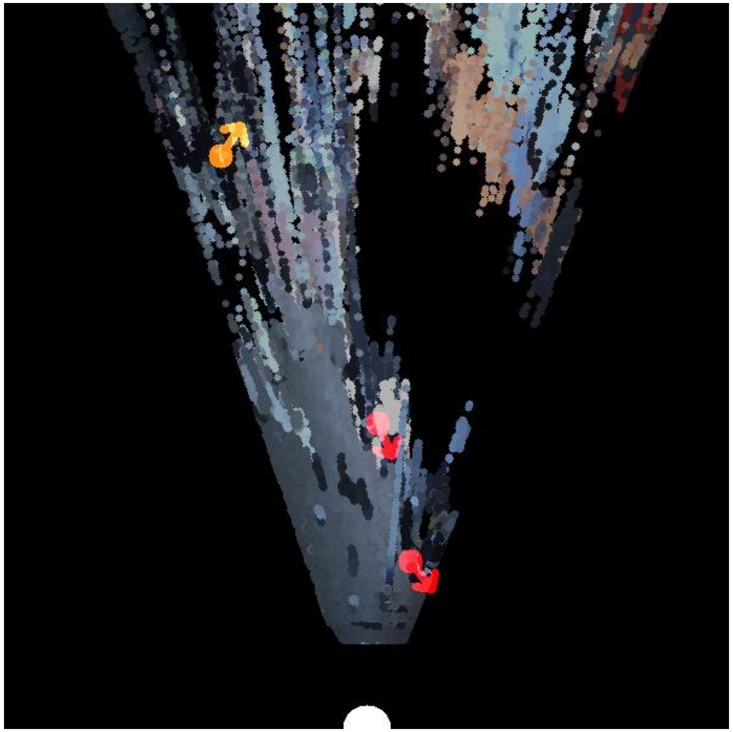
Results: Scene modeling



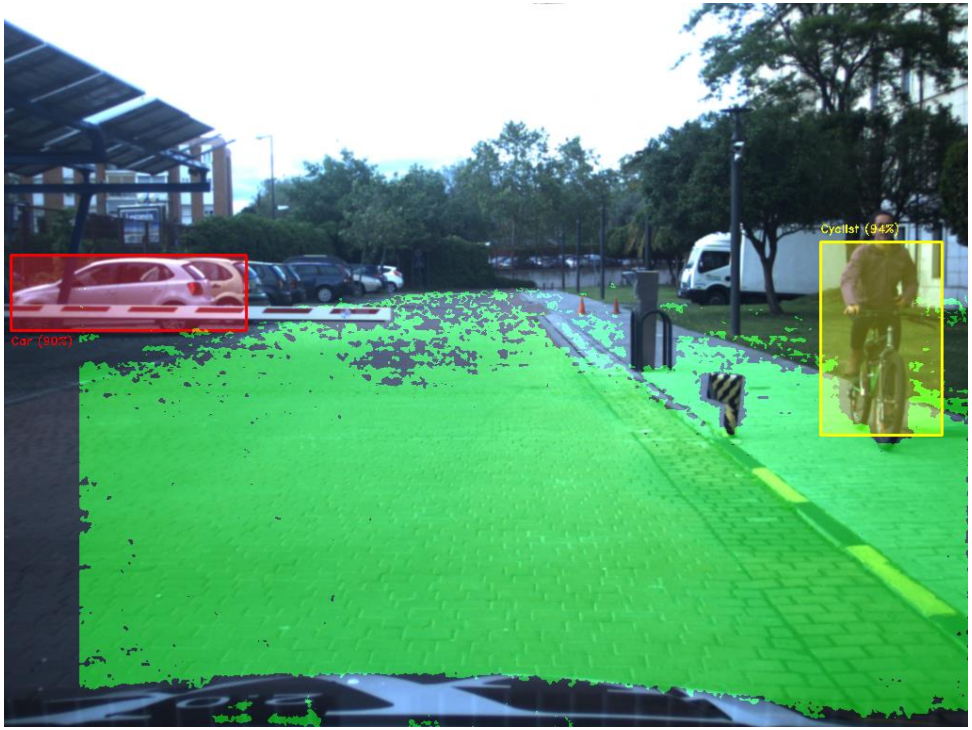


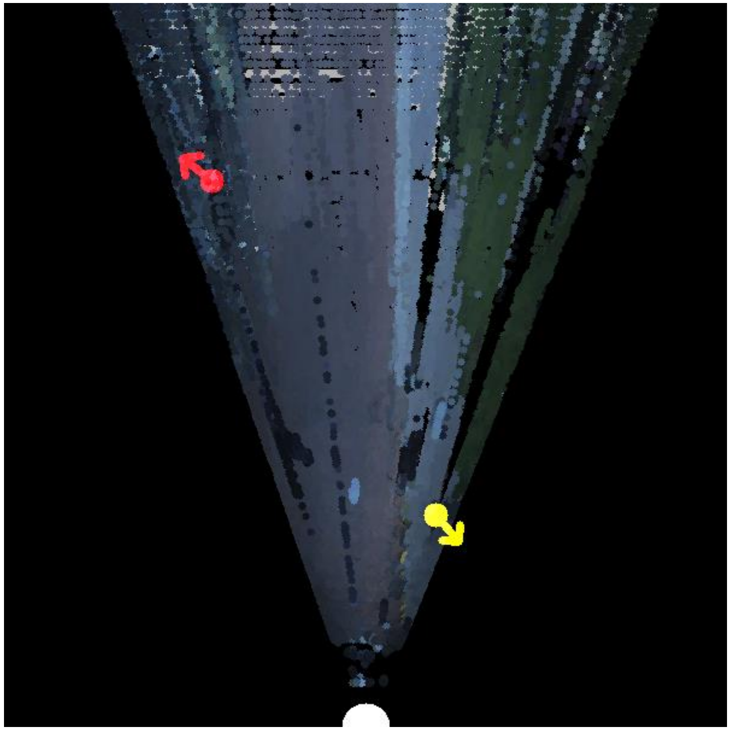
Results: Scene modeling





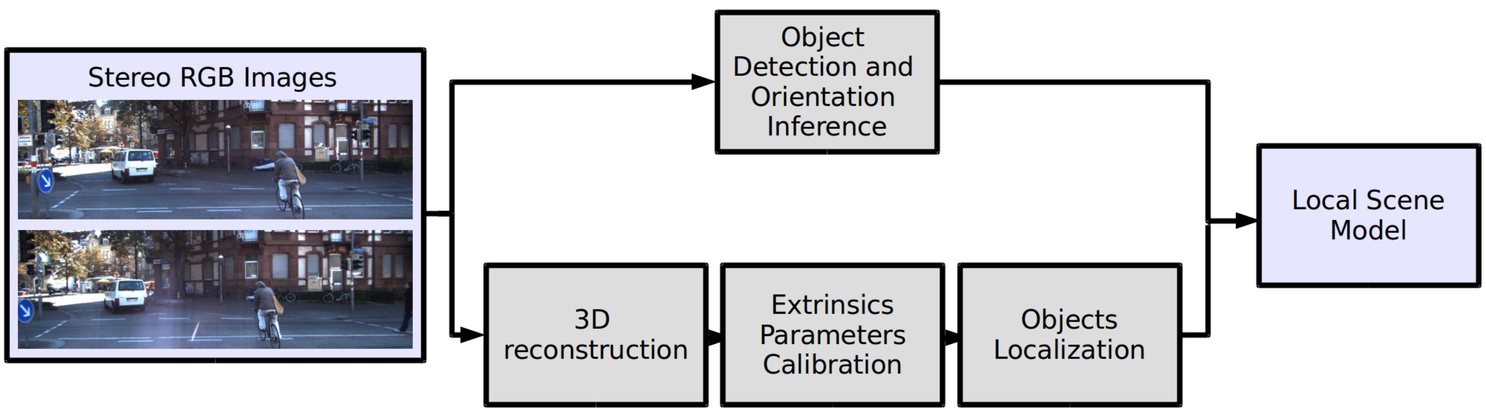
Results: Scene modeling





Agenda

1. Introduction
2. Obstacle detection
3. Scene modeling
4. Results
5. **Conclusion**



Conclusion

- Towards a full object-based scene understanding
  - CNN-based detection and viewpoint inference
  - Efficient approach: the same set of features is used for all tasks
- Stereo-vision 3D information is included for situation assessment
- Results validate our approach

Future work

- New categories of traffic elements
- Extension to the time domain
  - Tracking, filtering, etc.
- Including information from other perception modules
  - E.g., semantic segmentation

Code for CNN detection & viewpoints available at
https://github.com/cguindel/**lsi-faster-rcnn**

THANKS FOR YOUR ATTENTION

23 November 2017

ROBOT'2017 - Third Iberian Robotics Conference

Intelligent Systems Lab
www.uc3m.es/islab

Carlos Guindel · **cguindel.github.io** · cguindel@ing.uc3m.es