Visual Scene Understanding for Autonomous Driving

Raquel Urtasun

University of Toronto

Oct 3, 2014

Autonomous Driving

State of the art

- Localization, path planning, obstacle avoidance

Autonomous Driving

State of the art

- Localization, path planning, obstacle avoidance
- Heavy usage of Velodyne and detailed (recorded) maps

Autonomous Driving

State of the art

- Localization, path planning, obstacle avoidance
- Heavy usage of Velodyne and detailed (recorded) maps

Goal: autonomous driving cheap sensors and little prior knowledge

Autonomous Driving

State of the art

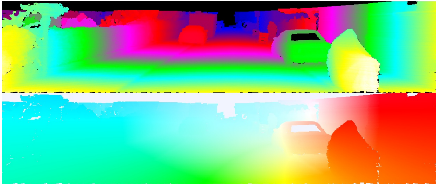
- Localization, path planning, obstacle avoidance
- Heavy usage of Velodyne and detailed (recorded) maps

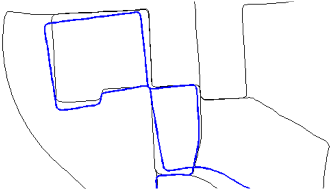
Goal: autonomous driving cheap sensors and little prior knowledge

Problems for computer vision

- Stereo, optical flow, visual odometry, structure-from-motion

Autonomous Driving





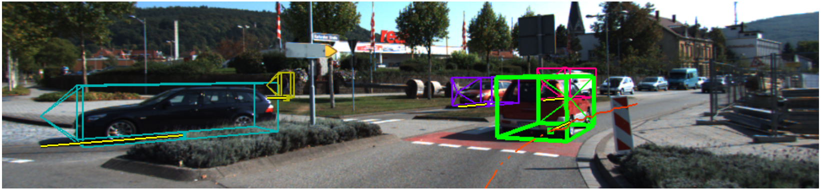
State of the art

- Localization, path planning, obstacle avoidance
- Heavy usage of Velodyne and detailed (recorded) maps

Goal: autonomous driving cheap sensors and little prior knowledge

Problems for computer vision

- Stereo, optical flow, visual odometry, structure-from-motion
- Object detection, recognition and tracking



State of the art

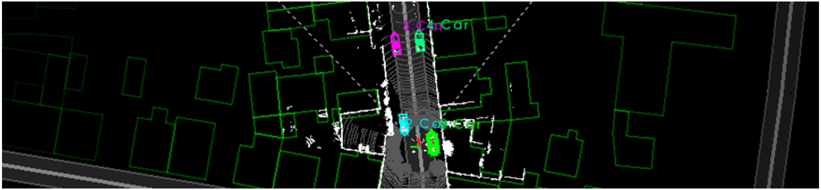
- Localization, path planning, obstacle avoidance
- Heavy usage of Velodyne and detailed (recorded) maps

Goal: autonomous driving cheap sensors and little prior knowledge

Problems for computer vision

- Stereo, optical flow, visual odometry, structure-from-motion
- Object detection, recognition and tracking
- 3D scene understanding

Autonomous Driving



State of the art

- Localization, path planning, obstacle avoidance
- Heavy usage of Velodyne and detailed (recorded) maps

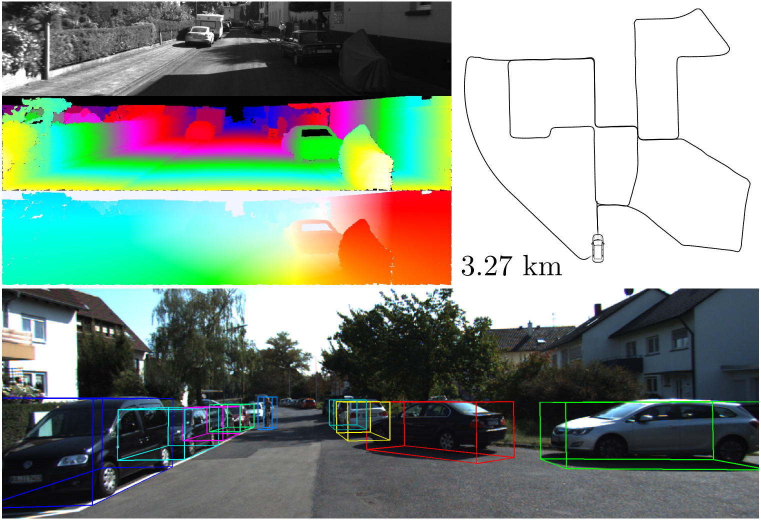
Goal: autonomous driving cheap sensors and little prior knowledge

Problems for computer vision

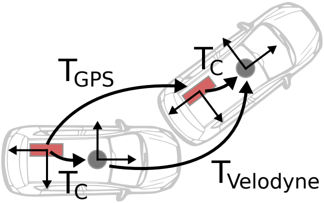
- Stereo, optical flow, visual odometry, structure-from-motion
- Object detection, recognition and tracking
- 3D scene understanding

Benchmarks: KITTI Data Collection

- **Two stereo rigs** (1392 $\times$ 512 px, 54 cm base, 90° opening)
- **Velodyne** laser scanner, **GPS+IMU** localization
- **6 hours** at 10 frames per second!

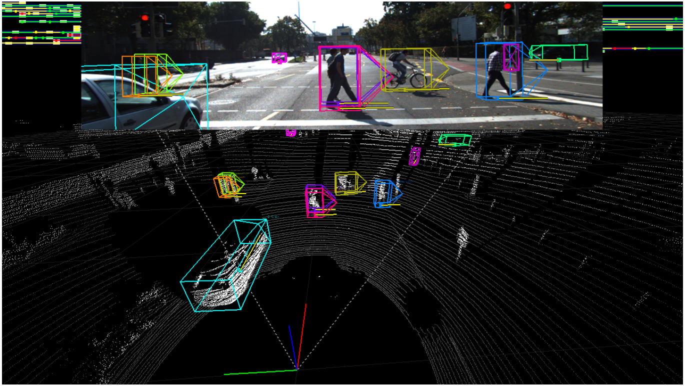

3.27 km

First Difficulty: Sensor Calibration



- Camera calibration [Geiger et al., ICRA 2012]
- Velodyne $\leftrightarrow$ Camera registration
- GPS+IMU $\leftrightarrow$ Velodyne registration

Second Difficulty: Object Annotation



- **3D object labels:** Annotators (undergrad students from KIT working for months)
- **Occlusion labels:** Mechanical Turk

One more Difficulty: Evaluation

- More than 200 submissions, 8000 downloads since CVPR 2012!

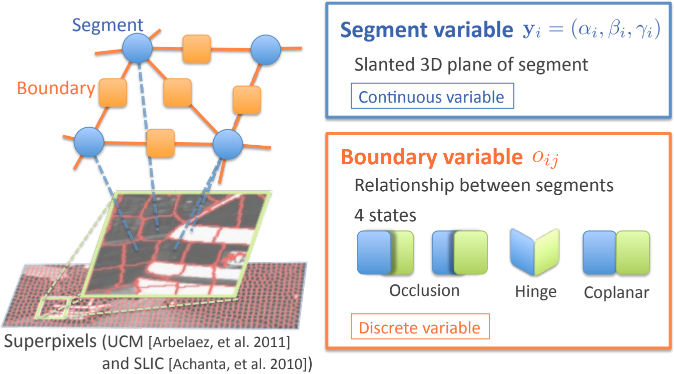
An autonomous system has to <u>sense the environment</u>

3D Reconstruction

- Goal: given 2 cameras mounted on top of the car, reconstruct the environment in 3D.

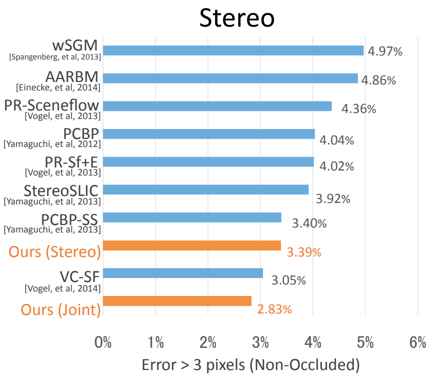
Joint Stereo, Flow, Occlusion and Segmentation

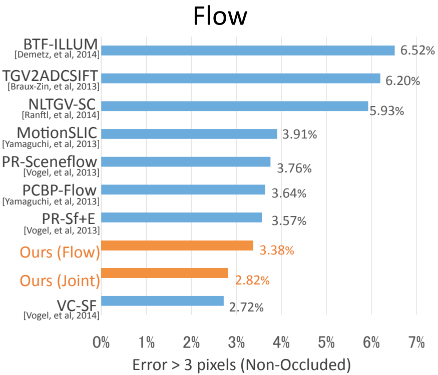
- Slanted-plane MRF with explicit occlusion handling which also computes an over-segmentation of the image into superpixels
- MRF on continuous variables (slanted planes) and discrete var. (boundary, super pixel assignments, outliers)



- Energy that looks at shape, compatibility and boundary length

Comparison to the State-of-the-art on KITTI



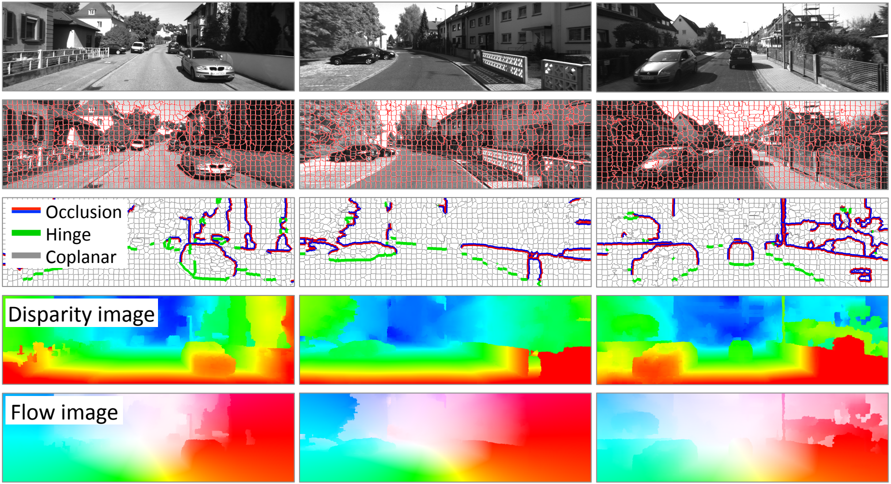


- Runtime on 1Core@3.5GHz for average resolution 1237.1 x 374.1 pixels

	Joint	Stereo only	Flow only
Total runtime	26.3 sec.	4.8 sec.	11.0 sec.

Results on KITTI

[K. Yamaguchi, D. McAllester and R. Urtasun, ECCV 2014]

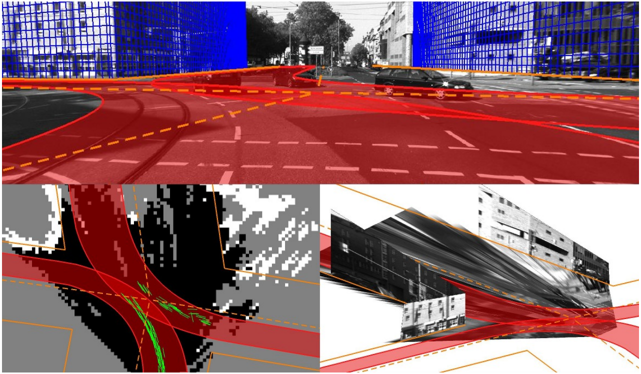


An autonomous system has to understand the scene in 3D

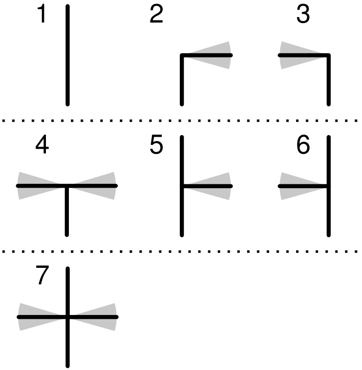
3D Scene Understanding

Goal: Infer from a short (≈10s) video sequence:

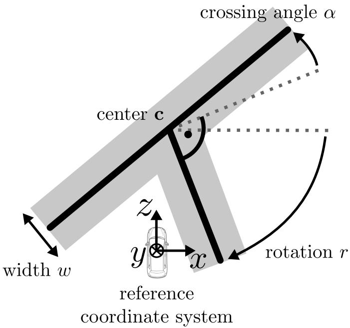
- **Geometric properties**, e.g., street orientation
- **Topological properties**, e.g., number of intersecting streets
- **Semantic activities**, e.g., traffic situations at an intersection
- **3D objects**, e.g., cars



Geometric Model



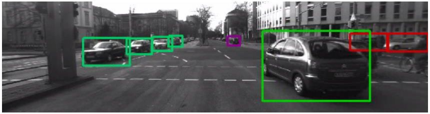
(Model topology)



(Geometric parameters)

Observations

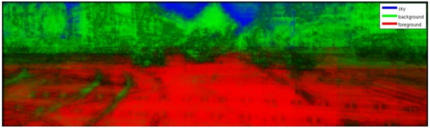
- **3D Tracklets**: Generate tracklets from 2D detections in 3D by employing the orientation as well as size of the bounding boxes



Static and Dynamic Observations

Observations

- **3D Tracklets**: Generate tracklets from 2D detections in 3D by employing the orientation as well as size of the bounding boxes
- **Segmentation** of the scene into semantic labels.

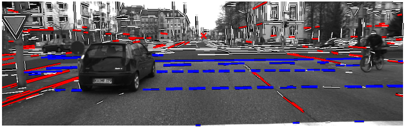


- **Lines** that follow the dominant orientations in the scene (i.e., reasoning about vanishing points).

Static and Dynamic Observations

Observations

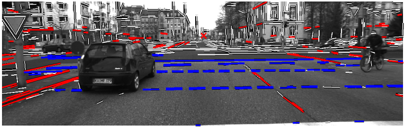
- **3D Tracklets**: Generate tracklets from 2D detections in 3D by employing the orientation as well as size of the bounding boxes
- **Segmentation** of the scene into semantic labels.
- **Lines** that follow the dominant orientations in the scene (i.e., reasoning about vanishing points).



Static and Dynamic Observations

Observations

- **3D Tracklets**: Generate tracklets from 2D detections in 3D by employing the orientation as well as size of the bounding boxes
- **Segmentation** of the scene into semantic labels.
- **Lines** that follow the dominant orientations in the scene (i.e., reasoning about vanishing points).

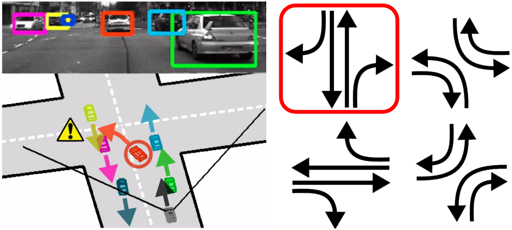


Representation

- We will reason about dynamics in bird eye's perspective and static in the image.

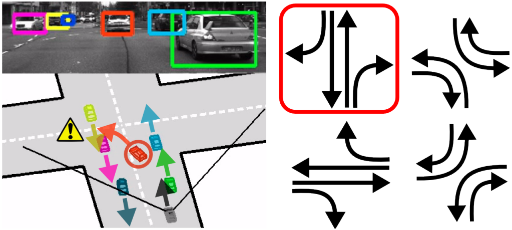
Why high-order semantics?

- Certain behaviors are not possible given the traffic "patterns"



Why high-order semantics?

- Certain behaviors are not possible given the traffic "patterns"

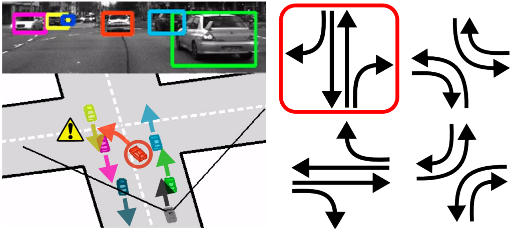


- We learned those patterns from data. Example of **traffic patterns** learned from data for 4 way intersections

- The arrows represent our concept of **lane**

Why high-order semantics?

- Certain behaviors are not possible given the traffic "patterns"



- We learned those patterns from data. Example of **traffic patterns** learned from data for 4 way intersections

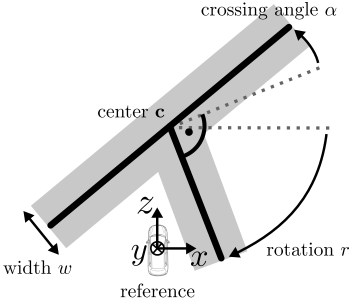
- The arrows represent our concept of **lane**

Joint Model

- Let a be the traffic pattern, and l_n the lane associated with a tracklet
- Road parameters are $\mathcal{R} = \{\theta, r, \mathbf{c}, w, \alpha\}$,
- The joint distribution is

$$p(\mathcal{E}, \mathcal{R}) = \underbrace{p(\mathcal{R})}_{\text{prior}} \underbrace{\left[\sum_a \prod_{n=1}^{N} \sum_{l_n} p(\mathbf{t}_n, l_n, a|\mathcal{R}) \right]}_{\text{Vehicle Tracklets}} \underbrace{p(v_f|\mathcal{R}) p(v_c|\mathcal{R})}_{\text{Vanishing Points}} \quad \underbrace{p(\mathbf{S}|\mathcal{R})}_{\text{Semantic Labels}}$$

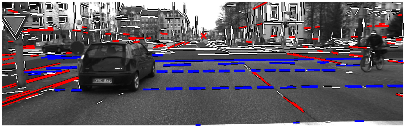
with $\mathcal{E}$ the image evidence.



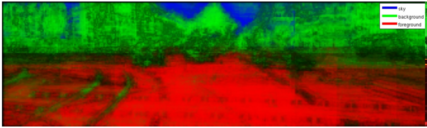
Vanishing Points and Segmentation Likelihoods

$$p(\mathcal{E},\mathcal{R}) = \underbrace{p(\mathcal{R})}_{\text{prior}} \underbrace{\left[\sum_a \prod_{n=1}^{N} \sum_{l_n} p(\mathbf{t}_n, l_n, a|\mathcal{R})\right]}_{\text{Vehicle Tracklets}} \underbrace{p(v_f|\mathcal{R})p(v_c|\mathcal{R})}_{\text{Vanishing Points}} \quad \underbrace{p(\mathbf{S}|\mathcal{R})}_{\text{Semantic Labels}}$$

- Make geometry agree with the **vanishing points**

- Make geometry agree with the **segmentation**



Full Graphical Model

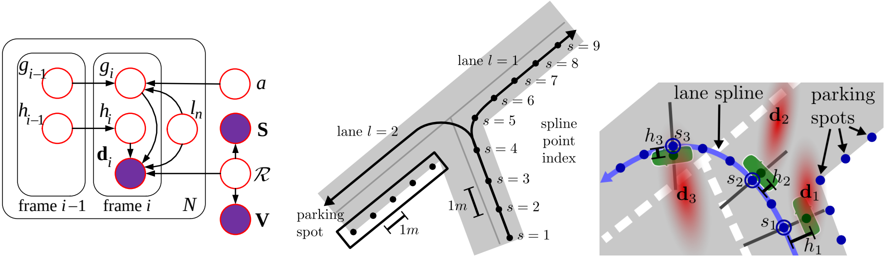
- The joint distribution is

$$p(\mathcal{E}, \mathcal{R}) = \underbrace{p(\mathcal{R})}_{\text{prior}} \underbrace{\left[\sum_{a} \prod_{n=1}^{N} \sum_{l_n} p(\mathbf{t}_n, l_n, a | \mathcal{R}) \right]}_{\text{Vehicle Tracklets}} \underbrace{p(v_f | \mathcal{R}) p(v_c | \mathcal{R})}_{\text{Vanishing Points}} \quad \underbrace{p(\mathbf{S} | \mathcal{R})}_{\text{Semantic Labels}}$$

  with $\mathcal{E}$ the image evidence, $\mathcal{R}$ the intersection variables, l_n the lane index and a the traffic pattern

- The vehicle tracklets are a little bit more complicated than described so far

Tracklet model



We reason about:

- parked cars: in which spot?
- moving vehicles: in which lane and where in the lane are they?
- the traffic situation (i.e., traffic pattern)

Our tracklet formulation $p(\mathbf{t}_n, l_n, a|\mathcal{R})$ combines a HMM with a dynamical system with constraints

Results: Geometry and Trackets estimation

Inference is done via Metropolis Hastings sampling

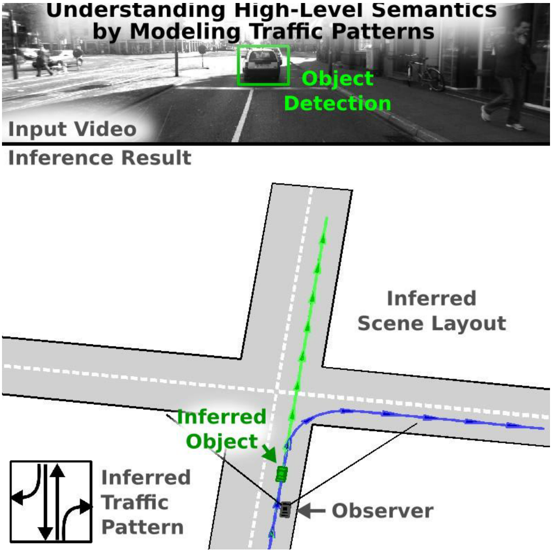
Method	Location		Orientation		Overlap		Pattern error	
	3-arm	4-arm	3-arm	4-arm	3-arm	4-arm	3-arm	4-arm
[Geiger11]	**4.3 m**	5.4 m	3.3 deg	8.0 deg	58.7%	56.0%	–	–
Ours	5.7 m	**4.9 m**	**2.4 deg**	**4.3 deg**	**61.5%**	**61.3%**	**18.2%**	**19.4%**

Table : Geometry estimation

Method	T-L error (all)		T-L error (>10m)	
	3-arm	4-arm	3-arm	4-arm
[Geiger11]	46.7%	49.9%	17.9%	30.1%
Ours	**15.2%**	**30.1%**	**3.6%**	**14.0%**

Table : Tracklet accuracy

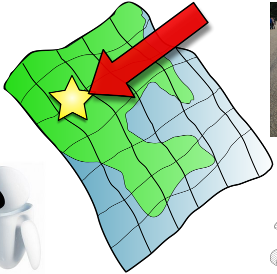
[H. Zhang, A. Geiger and R. Urtasun, ICCV 2013]

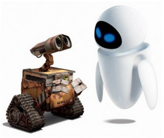


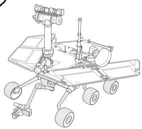
An autonomous system has to <u>self-localize</u>

Motivation

- Localization is crucial for autonomous systems







- GPS has limitations in terms of reliability and availability
- Place recognition techniques use image features or depth maps and a database of previously collected images (e.g., Google car)
- We develop an inexpensive technique for localizing to 3m in unseen regions

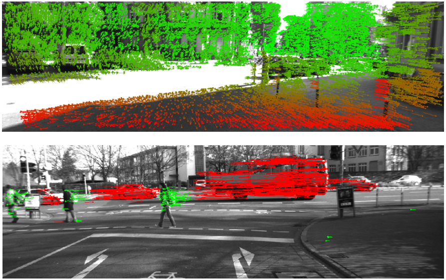
Humans as an inspiration

- Humans are able to use a map, combined with visual input and exploration, to localize effectively
- Detailed, community developed maps are freely available (OpenStreetMap)
- How can we exploit maps, combined with visual cues, to localize a vehicle?

Probabilistic Localization using Visual Odometry

- Visual odometry provides a strong source of information for localization
- Visual odometry has some issues
  - Over short time periods it can be noisy and highly ambiguous
  - Over long time periods it drifts when integrated

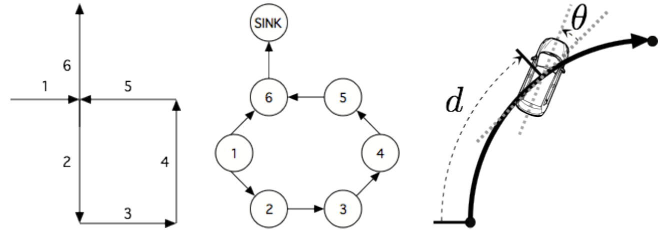
We adopt a probabilistic approach to represent and maintain this uncertainty



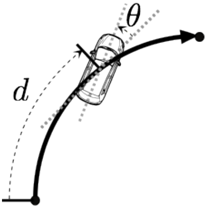
[Geiger et al, IV 2011]

Probabilistic Localization using Visual Odometry

- Maps can be considered as a graph
    - Nodes of the graph represent street segments
    - Edges represent intersections and allowed transitions between these segments
- Position is defined by the current street and the distance travelled **d**, and orientation θ on that street



Probabilistic Localization using Visual Odometry



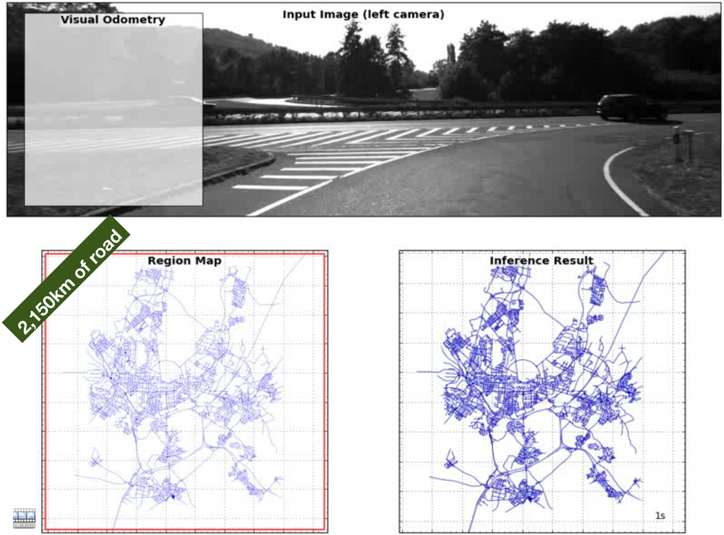
- The complete state includes
  - u_t the current street segment
  - $\mathbf{s}_t = (d_t, \theta_t, d_{t-1}, \theta_{t-1})$ the current and previous position and orientation on the street segment
- Odometry observation $\mathbf{y}_{1:t} = (\mathbf{y}_1, \cdots, \mathbf{y}_t)$

- Localization is formulated as posterior inference $p(u_t, \mathbf{s}_t | \mathbf{y}_{1:t})$

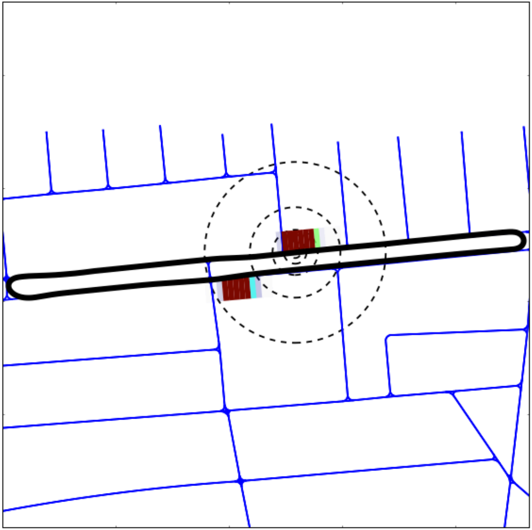
$$\propto \underbrace{p(\mathbf{y}_t | u_t, \mathbf{s}_t)}_{\text{likelihood}} \sum_{u_{t-1}} \int \underbrace{p(u_t | u_{t-1}, \mathbf{s}_{t-1})}_{\text{street transition}} \underbrace{p(\mathbf{s}_t | u_t, u_{t-1}, \mathbf{s}_{t-1})}_{\text{pose transition}} \underbrace{p(u_{t-1}, \mathbf{s}_{t-1} | \mathbf{y}_{1:t-1})}_{\text{previous posterior}} d\mathbf{s}_{t-1}$$

Results

[M. Brubaker, A. Geiger and R. Urtasun, CVPR13 best paper runner up award]

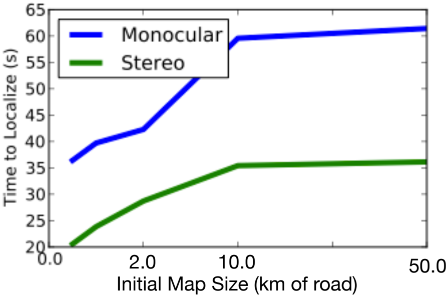


Ambiguous Sequences



Quantitative Experiments

Average	Stereo Odometry	Monocular Odometry	Map Projection
Position Error	3.1m	18.4m	1.4m
Heading Error	1.3°	3.6°	-
Localization Time	36s	62s	-



Acknowledgements

- Marcus Brubaker
- Andreas Geiger
- Tamir Hazan
- Philip Lenz
- David McAllester
- Jian Peng
- Alex Schwing
- Christoph Stiller
- Koichiro Yamaguchi
- Hongyi Zhang

Conclusions

Autonomous systems should

- Sense the environment: stereo, flow, layout estimation
- Recognize the 3D world: detection, segmentation
- Interact with it

We can do fairly complex reasoning with cheap sensors (i.e., 1 or 2 cameras)

Near Future:

- Close the loop between localization and semantics: use of maps
- Learning deep structure models
- Online memory/computation bounded tracking
- Real-time: HW accelerators