Visual Scene Understanding for Autonomous Driving

Raquel Urtasun

University of Toronto

Oct 3, 2014
State of the art

- Localization, path planning, obstacle avoidance
State of the art

- Localization, path planning, obstacle avoidance
- Heavy usage of Velodyne and detailed (recorded) maps
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
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
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
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 × 512 px, 54 cm base, 90° opening)
- **Velodyne** laser scanner, **GPS+IMU** localization
- **6 hours** at 10 frames per second!
The KITTI Vision Benchmark Suite
First Difficulty: Sensor Calibration

360° Velodyne Laserscanner

- Camera calibration [Geiger et al., ICRA 2012]
- Velodyne ↔ Camera registration
- GPS + IMU ↔ Velodyne registration
Second Difficulty: Object Annotation

- **3D object labels**: Annotators (undergrad students from KIT working for months)
- **Occlusion labels**: Mechanical Turk
More than 200 submissions, 8000 downloads since CVPR 2012!
An autonomous system has to sense the environment
3D Reconstruction

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

Stereo Camera Rig

Monochrome | Color
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)

Segment variable $y_i = (\alpha_i, \beta_i, \gamma_i)$

- Slanted 3D plane of segment
- Continuous variable

Boundary variable $o_{ij}$

- Relationship between segments
- 4 states
  - Occlusion
  - Hinge
  - Coplanar
- Discrete variable

Energy that looks at shape, compatibility and boundary length
Comparison to the State-of-the-art on KITTI

### Stereo

<table>
<thead>
<tr>
<th>Method</th>
<th>Error &gt; 3 pixels (Non-Occluded)</th>
</tr>
</thead>
<tbody>
<tr>
<td>wSGM [Spangenberg, et al, 2013]</td>
<td>4.97%</td>
</tr>
<tr>
<td>PR-SceneFlow [Vogel, et al, 2013]</td>
<td>4.36%</td>
</tr>
<tr>
<td>PR-Sf+E [Vogel, et al, 2013]</td>
<td>4.02%</td>
</tr>
<tr>
<td>StereoSLIC [Yamaguchi, et al, 2013]</td>
<td>3.92%</td>
</tr>
<tr>
<td>Ours (Stereo)</td>
<td>3.39%</td>
</tr>
<tr>
<td>VC-SF [Vogel, et al, 2014]</td>
<td>3.05%</td>
</tr>
<tr>
<td>Ours (Joint)</td>
<td>2.83%</td>
</tr>
</tbody>
</table>

### Flow

<table>
<thead>
<tr>
<th>Method</th>
<th>Error &gt; 3 pixels (Non-Occluded)</th>
</tr>
</thead>
<tbody>
<tr>
<td>BTF-ILLUM [Demetz, et al, 2014]</td>
<td>6.52%</td>
</tr>
<tr>
<td>MotionSLIC [Yamaguchi, et al, 2013]</td>
<td>5.93%</td>
</tr>
<tr>
<td>PR-Sceneflow [Vogel, et al, 2013]</td>
<td>3.91%</td>
</tr>
<tr>
<td>PR-Sf+E [Vogel, et al, 2013]</td>
<td>3.64%</td>
</tr>
<tr>
<td>Ours (Flow)</td>
<td>3.38%</td>
</tr>
<tr>
<td>Ours (Joint)</td>
<td>2.82%</td>
</tr>
<tr>
<td>VC-SF [Vogel, et al, 2014]</td>
<td>2.72%</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th></th>
<th>Joint</th>
<th>Stereo only</th>
<th>Flow only</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total runtime</td>
<td>26.3 sec.</td>
<td>4.8 sec.</td>
<td>11.0 sec.</td>
</tr>
</tbody>
</table>
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)
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.
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).
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).
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 $\mathcal{R} = \{\theta, r, c, w, \alpha\}$.
- The joint distribution is
  \[
p(E, \mathcal{R}) = p(\mathcal{R}) \sum_a \prod_{n=1}^{N} \sum_{l_n} p(t_n, l_n, a | \mathcal{R}) p(v_f | \mathcal{R}) p(v_c | \mathcal{R}) p(S | \mathcal{R})
  \]
  with $E$ the image evidence.

- Vehicle Tracklets
- Vanishing Points
- Semantic Labels

- Simulation results with $E$ the image evidence.
Vanishing Points and Segmentation Likelihoods

\[ p(\mathcal{E}, \mathcal{R}) = p(\mathcal{R}) \left[ \sum_{a} \prod_{n=1}^{N} \sum_{l_n} p(t_n, l_n, a | \mathcal{R}) \right] \]

- Make geometry agree with the **vanishing points**
- Make geometry agree with the **segmentation**
The joint distribution is

\[ p(\mathcal{E}, \mathcal{R}) = p(\mathcal{R}) \sum_{a} \prod_{n=1}^{N} \sum_{l_n} p(t_n, l_n, a|\mathcal{R}) \]

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.
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 \( p(t_n, l_n, a|R) \) combines a HMM with a dynamical system with constraints.
Inference is done via Metropolis Hastings sampling

<table>
<thead>
<tr>
<th>Method</th>
<th>Location</th>
<th>Orientation</th>
<th>Overlap</th>
<th>Pattern error</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>3-arm</td>
<td>4-arm</td>
<td>3-arm</td>
<td>4-arm</td>
</tr>
<tr>
<td>[Geiger11]</td>
<td>4.3 m</td>
<td>5.4 m</td>
<td>3.3 deg</td>
<td>8.0 deg</td>
</tr>
<tr>
<td>Ours</td>
<td>5.7 m</td>
<td>4.9 m</td>
<td>2.4 deg</td>
<td>4.3 deg</td>
</tr>
</tbody>
</table>

**Table**: Geometry estimation

<table>
<thead>
<tr>
<th>Method</th>
<th>T-L error (all)</th>
<th>T-L error (&gt;10m)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>3-arm</td>
<td>4-arm</td>
</tr>
<tr>
<td>[Geiger11]</td>
<td>46.7%</td>
<td>49.9%</td>
</tr>
<tr>
<td>Ours</td>
<td>15.2%</td>
<td>30.1%</td>
</tr>
</tbody>
</table>

**Table**: Tracklet accuracy
Semantic Scene Understanding

Understanding High-Level Semantics by Modeling Traffic Patterns

Input Video

Inference Result

Inferred Scene Layout

Inferred Object

Inferred Traffic Pattern

Observer

[H. Zhang, A. Geiger and R. Urtasun, ICCV 2013]
An autonomous system has to **self-localize**
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]
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 $\theta$ on that street
The complete state includes
- $u_t$ the current street segment
- $s_t = (d_t, \theta_t, d_{t-1}, \theta_{t-1})$ the current and previous position and orientation on the street segment

Odometry observation
$y_{1:t} = (y_1, \cdots, y_t)$

Localization is formulated as posterior inference $p(u_t, s_t | y_{1:t})$

$\propto p(y_t | u_t, s_t) \sum_{u_{t-1}} \int p(u_t | u_{t-1}, s_{t-1}) p(s_t | u_t, u_{t-1}, s_{t-1}) p(u_{t-1}, s_{t-1} | y_{1:t-1}) ds_{t-1}$
Results

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

<table>
<thead>
<tr>
<th></th>
<th>Stereo Odometry</th>
<th>Monocular Odometry</th>
<th>Map Projection</th>
</tr>
</thead>
<tbody>
<tr>
<td>Position Error</td>
<td>3.1m</td>
<td>18.4m</td>
<td>1.4m</td>
</tr>
<tr>
<td>Heading Error</td>
<td>1.3°</td>
<td>3.6°</td>
<td>-</td>
</tr>
<tr>
<td>Localization Time</td>
<td>36s</td>
<td>62s</td>
<td>-</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Initial Map Size (km of road)</th>
<th>Time to Localize (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0</td>
<td>20</td>
</tr>
<tr>
<td>2.0</td>
<td>25</td>
</tr>
<tr>
<td>10.0</td>
<td>35</td>
</tr>
<tr>
<td>50.0</td>
<td>65</td>
</tr>
</tbody>
</table>
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