Extensible and Verifiable Nets

Gijs Dubbelman and Panagiotis Meletis

Mobile Perception Systems
Electrical Engineering Department
Eindhoven University of Technology
Mobile Perception Systems

tue-mps.org

- 6 PhDs, postdoc, assistant prof, project manager, software engineer
- Research: 3D Computer Vision, Visual SLAM, Deep Learning
- Projects: H2020 INLANE, Cloud-LSVA, VI-DAS, Autopilot, etc.
Deep Learning for Autonomous Driving

Deep Learning brings SAE 5 driving closer to reality

But how far are we?

Pang et al. 2017
TomTom RoadDNA
Muñoz-Bulnes et al. 2017
Chabot et al. 2017
Google
Behl et al. 2017
Meletis et al. 2017
Deep Learning for Autonomous Driving

Lets try to make all the latest and greatest nets real-time and put them in a car...

Huge challenges in efficiency of networks and hardware
Make Networks More Efficient

• Aim: reuse computation for multiple classifiers

• Our task: Semantic Scene Segmentation

• Goal: Extend the number of classes, without an extra labeling effort and by maximally reusing feature computation layers
Approach: Hierarchical Network

Datasets tree:
- **Core dataset:** high-level classes
  - traffic sign, road, vehicle, ...
- **Auxiliary dataset 1:** sign subclasses
  - keep right, yield, roundabout, ...
- **Auxiliary dataset 2:** road subclasses
  - ego lane, lane markings, ...
- **Auxiliary dataset 3:** vehicle subclasses
  - bus, car, truck, ...
- **Auxiliary dataset 4:** car subclasses
  - per image annotations
  - car models, brands, ...

Classifiers tree:
- **Core classifier**
- **Auxiliary classifier 1**
- **Auxiliary classifier 2**
- **Auxiliary classifier 3**
- **Auxiliary classifier 4**

Semantic masks
Approach: Hierarchical Network

- Hierarchical Decision Rule: each pixel receives labels from a path along the classifiers tree.
- Hierarchical Loss: each classifier is trained only on the True Positive pixels of its parent.
Experiments: Cityscapes and GTSDB

- Cityscapes: 19 classes, with per-pixel annotations
- GTSDB: 43 traffic sign sub classes, with bounding box annotations

Goal: per-pixel segmentation of 19+43 classes
Results

- 19 Cityscapes classes and add 43 GTSDB classes = 200% incr.
- Computational increase < 6%
- No extra labeling effort needed
Quantitative Comparison

- Goal: Train traffic sign sub-classes on GTSDB and test on Cityscapes
- Compare with a flat classifier with our hierarchical classifier

The hierarchical classifier approach does better than a flat classifier approach, even when the flat classifier is trained on the target dataset.
Conclusion and Future Work

• Hierarchical classifier has specific benefits: **efficient class extensibility without extra labeling**

• Currently only traffic sign sub-classes

  • Add vulnerable Road User sub-classes
    - child, elderly, youngster, etc.

  • Add road attribute markings
    - lanes, temporary lanes, arrows, etc.

• **Do this without an extra labelling effort**