
Driverseat: Crowdstrapping Learning Tasks for Autonomous Driving

Pranav Rajpurkar
Toki Migimatsu
Jeff Kiske

Stanford University Computer Science Dept., 353 Serra Mall, Stanford, CA 94305 USA

PRANAVSR@CS.STANFORD.EDU
TAKATOKI@CS.STANFORD.EDU
JKISKE@STANFORD.EDU

Royce Cheng-Yue

3265 Falerno Way, San Jose, CA 95135 USA

RCHENGYUE@GMAIL.COM

Sameep Tandon

Tao Wang

Andrew Ng

Stanford University Computer Science Dept., 353 Serra Mall, Stanford, CA 94305 USA

SAMEEP@STANFORD.EDU
TWANGCAT@STANFORD.EDU
ANG@CS.STANFORD.EDU

Abstract

While emerging deep-learning systems have out-classed knowledge-based approaches in many tasks, their application to detection tasks for autonomous technologies remains an open field for scientific exploration. Broadly, there are two major developmental bottlenecks: the unavailability of comprehensively labeled datasets and of expressive evaluation strategies. Approaches for labeling datasets have relied on intensive hand-engineering, and strategies for evaluating learning systems have been unable to identify failure-case scenarios. Human intelligence offers an untapped approach for breaking through these bottlenecks. This paper introduces *Driverseat*, a technology for embedding crowds around learning systems for autonomous driving. Driverseat utilizes crowd contributions for (a) collecting complex 3D labels and (b) tagging diverse scenarios for ready evaluation of learning systems. We demonstrate how Driverseat can *crowdstrap* a convolutional neural network on the lane-detection task. More generally, crowdstrapping introduces a valuable paradigm for any technology that can benefit from leveraging the powerful combination of human and computer intelligence.

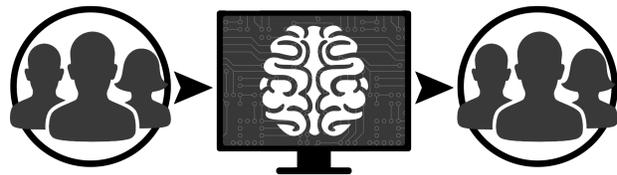


Figure 1. Embedding the crowd around learning systems for both training and evaluation, we can leverage the powerful combination of human and computer intelligence.

1. Introduction

Autonomous driving is a mission critical technology of the future. Road traffic accidents are responsible for nearly 1.3 million deaths globally each year, and are the leading cause of death among young people aged 15-29 (World Health Organization, 2013). In the quest to making roads accident-free, Advanced Driver Assistance Systems (ADAS) are making significant milestones: Adaptive Cruise Control systems adjust vehicle speed to maintain safe distance from vehicles ahead, and Automatic Braking systems react to imminent collisions (Markoff & Sengupta, 2013). The next generation of ADAS will introduce features such as Traffic Jam Assistance, Traffic Light Detection, and Lane Change Assist, which enable a new suite of critical safety advancements (Bar Hillel et al., 2012). Detection, or perception, is a key piece of the puzzle that the next generation of ADAS must solve (Ng, 2014): systems must perceive road boundaries, lane topologies, locations of other cars, pedestrians, signs, and miscellaneous obstacles. (Levinson et al., 2011). Classic techniques to solve the detection problem have relied on intensive hand-engineering, and have been unable

to capture the seemingly endless array of possibilities and conditions encountered on the road.

Deep learning systems represent an alternative approach (Hadsell et al., 2008). Powered by large network architectures and fueled by the emergence of comprehensively labeled datasets (Ng, 2015), deep learning systems have made significant leaps in visual object recognition (Krizhevsky et al., 2012), speech (Hannun et al., 2014), and language understanding tasks (Socher et al., 2013). In the context of autonomous driving, progress for learning systems is bottlenecked by the unavailability of both comprehensively labeled datasets and measures to evaluate performance robustness.

On one end of the learning pipeline, approaches to labeling datasets for detection tasks in autonomous driving have focused on hand-engineering (Borkar et al., 2012) or on synthetic generation of labels (Pomerleau, 1989). These approaches are (a) unable to capture the diversity and complexity of labels on roads (b) require months of heavy fine-tuning, and (c) are task-specific: an automated car-labeling system is not an effective lane-labeling system.

On the other end of the learning pipeline, approaches to evaluating the performance of learning systems for detection tasks have not been comprehensive (Geiger et al., 2012). While metrics for general system performance are useful, they give little insight into system performance in adverse lighting, road, and weather conditions. In building a production-ready system for autonomous driving, it is vital to quantify and qualify its robustness under a slew of different road, lighting and weather scenarios.

Human intelligence offers an untapped approach for both ends of the pipeline. Driving is a complex, yet quotidian, task for people. With experience in the driver’s seat, people have good intuition about road structures, car motion, lane topologies, and tricky environments. We hypothesize that we can integrate the experience of people to *crowdstrap* learning systems for autonomous driving.

As a step towards integrating human expertise into reliable autonomous driving, we present *Driverseat*, a web system that harnesses crowd contributions to bootstrap both the training and evaluation for learning systems. Two fundamental building blocks constitute *Driverseat*’s architecture:

1. **RoadEdit** leverages crowd contributions to train learning systems. It provides an toolbox for complex labeling tasks that are hard to automatically annotate.
2. **TagEval** utilizes human intelligence to evaluate learning systems. It exposes a framework to tag diverse road scenarios, on which the performance of learning systems can be evaluated.

The overarching contribution of the work is the idea of embedding human intelligence around learning systems for reliable autonomous driving. Beyond fueling the performance of learning systems, *Driverseat* is valuable in gaining intuition about the strengths and shortcomings of a learning system. For example, an early iteration of our detection system trained on data from California highways, which run North-South, performed poorly when facing the sun. In enabling such learnings, *Driverseat* guides research direction, especially important in building the perfect-performance systems expected for autonomous driving.

As a concrete demonstration of crowdstrapping learning tasks for autonomous driving, we focus on the lane detection task, an unfinished, yet vital, milestone for autonomous systems of the future. The lane detection task involves understanding the topology of the lanes around the car. It has been well investigated in the simple case, where the task is to detect the boundaries of the lane being driven in, also called the *ego-lane*, for a short distance ahead (Yenikaya et al., 2013). Approaches have relied on sophisticated modelling of road and motion (Bertozzi & Broggi, 1998; Cheng et al., 2006; Wang et al., 2004), yet have not been able to accurately model lane topologies in their full complexity.

In the remainder of this paper, we (a) describe the engineering and design aspects of *Driverseat*, (b) detail how machine automation can be leveraged to seed the crowd-labeling task, and (c) demonstrate *Driverseat*’s capability to crowdstrap a convolutional neural network for the lane detection task.

While the work uses the lane detection task in autonomous driving as an example, the idea of crowdstrapping learning systems introduces a valuable paradigm for any technologies that can benefit from leveraging the powerful combination of human and computer intelligence. Autonomous driving is a particularly powerful application, as we contribute to a future in which our lives are easier, technologies smarter, and world safer.

2. Driverseat: Engineering and Design

Driverseat is a web system that utilizes human intelligence for (a) collecting complex labels and (b) identifying scenario-specific weaknesses in learning systems. The modules for achieving those goals are called *RoadEdit* and *TagEval*, respectively.

The significant technical contribution of *Driverseat* is that it introduces a 3D web interface for crowd-interaction. Traditional labeling interfaces exploit only 2D labeling (Russell et al., 2008; Russakovsky et al., 2014). While 2D labeling is well suited for many tasks, complex labeling tasks in autonomous driving demand a more sophisticated 3D in-

interaction interface. Some of the key shortcomings of 2D labeling for a task such as lane-labeling are that (a) depth information of the points is not easily visible or modifiable (b) points further from the car are hard to label accurately, (c) segments need to be labeled redundantly when multiple camera images shared the same stretch of the road, (d) lanes occluded by cars are difficult to label, and (e) piecewise-linear segments in 2D cannot capture sharp road-curves. Driverseat is designed to overcome the shortcomings of the 2D lane-labeling system, making labeling more expressive.

2.1. RoadEdit Labeling

While some labeling tasks involve little more than drawing a bounding box around an object in an image, other labeling tasks are more complex (Kittur et al., 2011). In autonomous driving, the key requirement of depth information heightens the complexity of labeling tasks. While there are techniques for automatically estimating depth maps (Delage et al., 2007), implementation of such techniques requires a significant engineering effort. Driverseat’s RoadEdit provides a simple alternative, with an interface that empowers the crowd to take over complex 3D labeling tasks.

We demonstrate RoadEdit in the concrete context of lane-labeling, one of the most challenging labeling tasks for autonomous driving. The user goal of RoadEdit is to have the colored lane boundaries run through white lane-paint points in 3D map (shown in figure 2). The user also assigns each lane, or lane-segment, a particular type, such as *white dotted* or *yellow solid*. Each type corresponds to a particular lane color.

Lane-labeling in RoadEdit enables capture of a variety of lane structures: lanes arbitrarily curve, merge with each other, split into several paths (such as a highway exit or the fan out of an urban arterial into turning lanes at an intersection) or end abruptly (at the threshold of an intersection).

To capture these complex lane topologies, RoadEdit provides users with a variety of operations in their toolbox. Imprecise Lane boundaries can be dragged to “line up” on top of the lane markings. The drag range parameter controls the range of points affected by the drag, allowing for both short-range, and medium-range corrections. In addition to corrections, the user can use the *fork* operation to model splits, *append* to extend a lane, *delete* and *join* to model merges and abrupt ends. The combination of these operations can capture lane topologies of high complexity, which are unable to be extracted automatically.

Although the RoadEdit toolbox is lane-labeling specific, the techniques applied generalize to complex 3D labeling in and beyond the context of autonomous driving. For instance, any 3D labelling system must tackle the user-

interaction challenge that arises for 3D object manipulation on the 2D screen. For an object in 3D space, there are 6 degrees of freedom: 3 translational, and 3 rotational. Controlling all 6 degrees of freedom of an object is a nontrivial interaction task. Driverseat implements the well investigated workaround of introducing constraints on the control to limit the degrees of freedom (Fisher et al., 2011). In the context of lane-labelling, when a lane is dragged, lane points are restricted to move only on the ground plane, restricting lane movement to two degrees of freedom. Another generalizable technique that RoadEdit leverages is the embedding of gamification in the labeling task, allowing people to label while enjoying themselves (Von Ahn & Dabbish, 2004). Driverseat gamifies labelling by simulating the car’s drive on the road. The virtual car moves through the the virtual environment, following the path of the data-collection vehicle. By combining simulation with labeling, we make the labeling task interactive and intuitive.

Another facet of RoadEdit that extends to other labelling tasks is its crowd-programming pattern, which follows a fix and verify strategy (Bernstein et al., 2010). The dataset is partitioned into individual *drives*, where each drive consists of approximately 10 minute-long segments called *runs*. Each run is marked by a *marker-annotator* and verified by a *verifier-annotator*. For quality control, we (a) employ *Expert Review* (Quinn & Bederson, 2011), in which an annotated run is endorsed by an expert, and (b) use an expert to train and mentor a first-time annotator. Because Driverseat is exposed as a a web interface, trained annotators can perform labeling tasks remotely, and build on each other’s progress. While this brings scalability to labelling, it poses the challenge of aggregating user contributions. To overcome this, RoadEdit implements a rudimentary version control system that enables undo/redo operations within and across labeling sessions.

2.2. TagEval Evaluation

Evaluation of learning systems rarely gives insight into where systems succeed and fail. This insight would be valuable in guiding research direction, especially in the context of autonomous driving, for which near perfect performance is critical. If systems are able to recognize particularly precarious scenarios for which the confidence of the detections are low, the driver can be preemptively alerted.

TagEval is Driverseat’s scenario tagging interface used to tag road-conditions (splits, merges, bridges), lighting conditions (shadows, sun-facing), and weather conditions (rainy, snowy) on relevant road segments. These tags can then be used to evaluate how learning systems perform under these various scenarios.

The interface for TagEval is simple: labelers use a com-

Driverseat: Crowdstrapping Learning Tasks for Autonomous Driving



The **camera view** helps labellers see what the road actually looks like. Changes made in the 3D editor are reflected in the video in realtime.

A **context-aware toolbar** displays only the relevant commands for a given selection of points. In this case, the user can **copy**, **delete**, or change the **lane type** of the selected points.

Editor mode switches between **RoadEdit** and **TagEval**

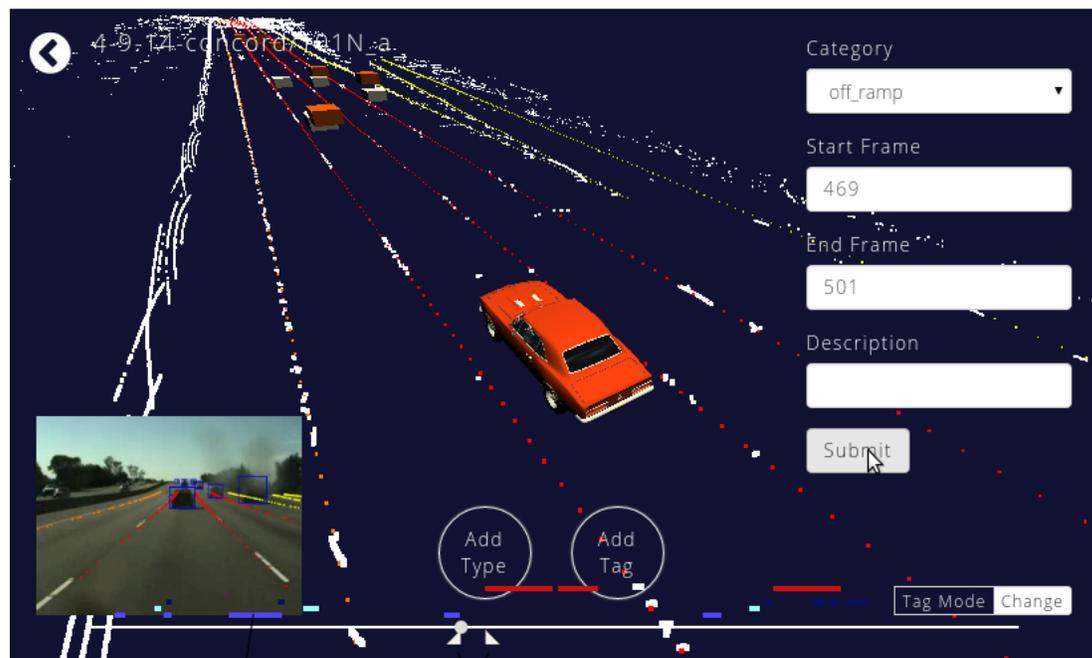


Figure 2. The Driverseat building blocks: RoadEdit (above) provides an toolbox for complex labeling tasks that are hard to automatically annotate, and TagEval (below) exposes a framework to tag diverse road scenarios.

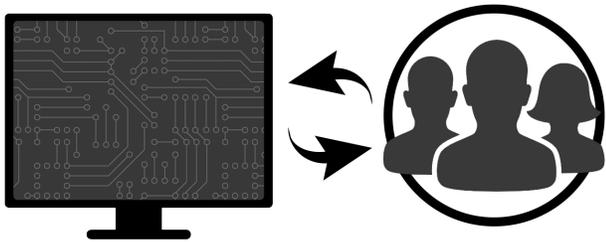


Figure 3. Driverseat leverages a symbiosis between computers and people: machine annotations bootstrap human computation, which in turn bootstrap machine learning systems.

combination of the virtual 3D environment and the 2D camera images to tag road-segment conditions (shown in figure 2). The tagging process involves selecting a start and end frame, and choosing a new/existing category associated with the tag. User access to multiple streams of information is beneficial: while the virtual 3D environment gives better clues about road and lane semantics, the 2D images give information about the environmental conditions.

3. Seeding Human Computation With Automation

Driverseat motivates the use of human computation in coordination with computer intelligence to solve complex problems (Hara et al., 2014). Having described how people can be used to label data for and evaluate learning systems, we now describe how computer automation can be used to supplement the human annotation task.

Lane-labeling is among a set of tasks that is complex for automatic labeling and time-consuming for human labeling. We describe how we can bootstrap automatic labeling to generate an initial ground-truth estimate, and ease the manual labeling task so that annotators only have to validate and make corrections to labels in complex scenarios (D’Orazio et al., 2009).

To generate an initial ground-truth estimate, we adopt a two-step engineering methodology. Firstly, we extract the left and right-lane boundaries of the ego-lane. Secondly, we extend the ego-lane estimates to multiple lanes.

For ego-lane boundary generation, we leverage information from the 3D road-maps and from data collection drives, during which lane changes are not performed by the driver. Thus, the GPS trajectory of the research vehicle runs within the ego-lane boundaries. We thus filter road points belonging to the ego-lane boundaries by keeping points that are within a lane-width distance from the GPS track. We further discriminate the points belonging to the left boundary from those belonging to the right boundary by using the

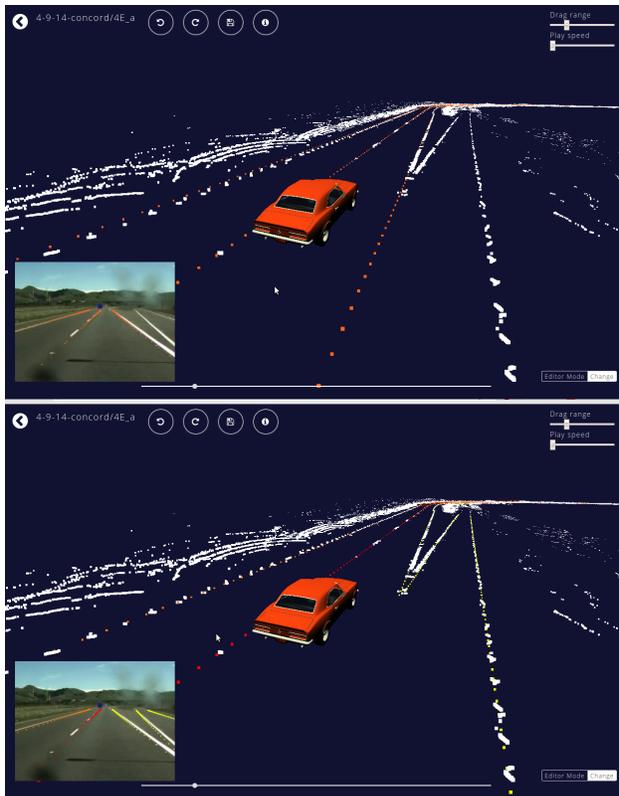


Figure 4. (a) The automatic labeling generates decent results for the ego-lane, but is unable to capture complex lane topologies such as splits. (b) The crowd-workers, seeded by automatic labeling, can capture these complexities.

sign of the lateral distance. After obtaining the points belonging to the left and right boundaries, we fit a piecewise linear curve to each boundary.

Because ground points away from the ego-lane are sparse, the technique for ego-lane generation does not generalize well in the multilane case. To generate estimates for multiple lanes, we make the simplifying assumption that neighboring lanes follow a very similar structure to the ego-lane. We can thus make a good initial guess of all the lane boundaries by shifting the auto-generated ego-lane boundaries laterally by the lane width. While we do not capture complex road topologies such as splits and merges, we obtain a good springboard for the human annotation task (refer to figure 4).

4. Crowdstrapping A Convolutional Neural Network for Lane Detection

We concretely attempt to integrate the experience of people to crowdstrap learning systems for autonomous driving by using Driverseat to train and evaluate a convolutional neural network on the lane detection task.

To acquire labelled data to train the neural network, we (a) collect data on California highways using our sensor-equipped research vehicle, (b) process the data to build 3D maps of the drives and generate an initial ground truth estimate for the lanes using automated labelling, and finally (c) use Driverseat’s RoadEdit interface to have human annotators correct and label complex lane topologies.

This labeled data serves as input to the neural network, the architecture of which is detailed in (Huval et al., 2015). The neural network’s task is to predict the pixel (x, y, depth) locations of the lane boundaries given an image of the road. We create a dataset consisting of lane-labelled images by projecting the labeled 3D data from the global (x, y, z) coordinates into the camera image. The neural network is finally trained on a large subset of this data, while the rest of the data is used for hold-out validation.

To evaluate the performance of the neural network, we compare lane predictions against ground-truth labels to compute precision and recall across a range of depths. We then use Driverseat’s TagEval framework to evaluate network performance in a variety of scenarios. Annotators tag road segments in the holdout set with interesting road, weather and lighting scenarios. Performance of the network is then evaluated specifically on images corresponding to those scenarios, and their results quantitatively and qualitatively analyzed. We can hence focus on collecting more data for the scenarios which challenge the network, and make the system robust for the road ahead.

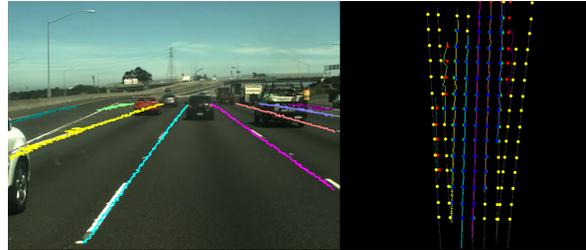
5. Results

We evaluate the performance of the convolutional neural network on the lane detection task in an array of different scenarios tagged with Driverseat’s TagEval interface, covering (a) left and right off-ramps, (b) road curves, (c) shadows, and (d) pavement changes (when the road changes texture/color).

Qualitative evaluation of network performance on these scenarios reveal strengths and difficulties for the network (see figure 5). On one hand, (a) off-ramps are challenging for the network, and emerging lanes are often missed, (b) shadows and pavement-changes are modestly difficult too, and the network becomes susceptible to misinterpreting reflective patches on the road with lane markings. On the other hand, the network is (a) robust to curves on highways, and (b) is able to detect lane boundaries even in the presence of occluding vehicles.

Quantitative evaluation gives further insight (see figure 6). For ego-lane boundary detection, we note that (a) performance on road-curves, though comparable with general performance, declines rapidly with distance (b) shadows and pavement changes are the most challenging conditions

for ego-lane detection. Graphing performances on the lanes adjacent to the ego-lane on either side, we identify (a) off-ramps injure detection performance on the splitting lane, and (b) shadows maintain to be a challenging scenario.



(a) Occasional left-side off-ramps challenge network detection of the leftmost lanes.



(b) Labels from TagEval reveal that the network is weak at capturing lanes in scenarios with off-ramps. However, note that on the left, the network is able to make lane detections even in the presence of the occluding truck.



(c) Shadows continue to pose a natural challenge for lane detection systems.



(d) Curves on the road, which we hypothesized would hurt the neural network’s performance, are handled well.

Figure 5. Example of neural network outputs on a slew of different scenarios.

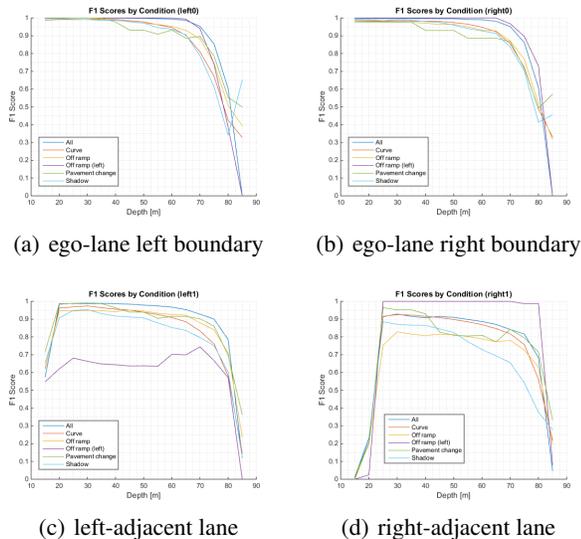


Figure 6. Quantitative evaluation of performance in a diverse range of scenarios.

6. Conclusions

The work presents Driverseat, a system that embeds crowds around, or *crowdstraps*, learning systems. In training of learning systems, we demonstrate how Driverseat uses crowd contributions to annotate labels that are hard to auto-generate. In the evaluation of learning systems, we demonstrate how Driverseat can leverage human intelligence to pinpoint scenario-specific weaknesses. To further motivate how human computation can be combined with machine computation, we demonstrate how automation can be used to provide initial estimates of ground-truth for the human annotation task. We conclude by applying all these techniques to the concrete task of lane detection in autonomous driving, a salient feature for the next generation of autonomous vehicles. We evaluate the performance of the network on a variety of scenarios, some of which are handled better than others.

As our learning systems continue to get better and more advanced, new challenges and opportunities arise. As we apply learning systems to increasingly complex problems, we need to explore more sophisticated strategies for combining human and computer intelligence. In this work, we have shown how we can integrate people’s knowledge and experience on the roads to “teach” machines to drive, en route to our goal of making roads accident-free. Driverseat points to a future in which artificial intelligence works hand-in-hand with human intelligence to solve the most complex of problems.

7. Acknowledgements

We would like to thank Professor Michael Bernstein for helpful conversations about this work. We would also like to thank Mary McDevitt her support in the writing of this paper.

References

- Bar Hillel, Aharon, Lerner, Ronen, Levi, Dan, and Raz, Guy. Recent progress in road and lane detection: a survey. *Machine Vision and Applications*, 25(3):727–745, 2012. doi: 10.1007/s00138-011-0404-2.
- Bernstein, Michael S, Little, Greg, Miller, Robert C, Hartmann, Björn, Ackerman, Mark S, Karger, David R, Crowell, David, and Panovich, Katrina. Soylent: a word processor with a crowd inside. In *Proceedings of the 23rd annual ACM symposium on User interface software and technology*, pp. 313–322. ACM, 2010.
- Bertozzi, M. and Broggi, A. GOLD: a parallel real-time stereo vision system for generic obstacle and lane detection. *IEEE Trans. on Image Processing*, 7(1), 1998. doi: 10.1109/83.650851.
- Borkar, A., Hayes, M., and Smith, M.T. A novel lane detection system with efficient ground truth generation. *Intelligent Transportation Systems, IEEE Transactions on*, 13(1):365–374, March 2012. ISSN 1524-9050. doi: 10.1109/TITS.2011.2173196.
- Cheng, Hsu-Yung, Jeng, Bor-Shenn, Tseng, Pei-Ting, and Fan, Kuo-Chin. Lane detection with moving vehicles in the traffic scenes. *Intelligent Transportation Systems, IEEE Transactions on*, 7(4):571–582, 2006.
- Delage, Erick, Lee, Honglak, and Ng, Andrew Y. Automatic single-image 3d reconstructions of indoor manhattan world scenes. In *Robotics Research*, pp. 305–321. Springer, 2007.
- D’Orazio, T., Leo, M., Mosca, N., Spagnolo, P., and Mazzeo, P.L. A semi-automatic system for ground truth generation of soccer video sequences. In *Advanced Video and Signal Based Surveillance, 2009. AVSS ’09. Sixth IEEE International Conference on*, pp. 559–564, Sept 2009. doi: 10.1109/AVSS.2009.69.
- Fisher, B., Dawson-Howe, K., and O’Sullivan, C. *Virtual and Augmented Architecture (VAA01): Proceedings of the International Symposium on Virtual and Augmented Architecture (VAA01), Trinity College, Dublin, 21 -22 June 2001*. Springer London, 2011. ISBN 9781447103370.
- Geiger, A., Lenz, P., and Urtasun, R. Are we ready for autonomous driving? the kitti vision benchmark suite. In

- Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 3354–3361, June 2012. doi: 10.1109/CVPR.2012.6248074.
- Hadsell, R., Erkan, A., Sermanet, P., Scoffier, M., Muller, U., and LeCun, Yann. Deep belief net learning in a long-range vision system for autonomous off-road driving. In *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pp. 628–633, Sept 2008. doi: 10.1109/IROS.2008.4651217.
- Hannun, Awni Y., Case, Carl, Casper, Jared, Catanzaro, Bryan C., Diamos, Greg, Elsen, Erich, Prenger, Ryan, Satheesh, Sanjeev, Sengupta, Shubho, Coates, Adam, and Ng, Andrew Y. Deep speech: Scaling up end-to-end speech recognition. *CoRR*, abs/1412.5567, 2014.
- Hara, Kotaro, Sun, Jin, Moore, Robert, Jacobs, David, and Froehlich, Jon. Tohme: Detecting curb ramps in google street view using crowdsourcing, computer vision, and machine learning. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*, UIST '14, pp. 189–204, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-3069-5. doi: 10.1145/2642918.2647403.
- Huval, B., Wang, T., Tandon, S., Kiske, J., Song, W., Pazhayampallil, J., Andriluka, M., Rajpurkar, P., Migimatsu, T., Cheng-Yue, R., Mujica, F., Coates, A., and Ng, A. Y. An Empirical Evaluation of Deep Learning on Highway Driving. *ArXiv e-prints*, April 2015.
- Kittur, Aniket, Smus, Boris, Khamkar, Susheel, and Kraut, Robert E. Crowdforge: Crowdsourcing complex work. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, UIST '11, pp. 43–52, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0716-1. doi: 10.1145/2047196.2047202.
- Krizhevsky, Alex, Sutskever, Ilya, and Hinton, Geoffrey E. Imagenet classification with deep convolutional neural networks. In Pereira, F., Burges, C.J.C., Bottou, L., and Weinberger, K.Q. (eds.), *Advances in Neural Information Processing Systems 25*, pp. 1097–1105. Curran Associates, Inc., 2012.
- Levinson, Jesse, Askeland, Jake, Becker, Jan, Dolson, Jennifer, Held, David, Kammel, Soeren, Kolter, J. Zico, Langer, Dirk, Pink, Oliver, and Pratt, Vaughan et al. Towards fully autonomous driving: Systems and algorithms. *2011 IEEE Intelligent Vehicles Symposium (IV)*, 2011. doi: 10.1109/ivs.2011.5940562.
- Markoff, John and Sengupta, Somini. Drivers With Hands Full Get a Backup: The Car. *New York Times*, 2013.
- Ng, Andrew. Deep learning: Machine learning via large-scale brain simulations, 2014.
- Ng, Andrew. GPU Technology Conference, 2015.
- Pomerleau, Dean A. Alvin: An autonomous land vehicle in a neural network. In Touretzky, D.S. (ed.), *Advances in Neural Information Processing Systems 1*, pp. 305–313. Morgan-Kaufmann, 1989.
- Quinn, Alexander J and Bederson, Benjamin B. Human computation: a survey and taxonomy of a growing field. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pp. 1403–1412. ACM, 2011.
- Russakovsky, Olga, Deng, Jia, Su, Hao, Krause, Jonathan, Satheesh, Sanjeev, Ma, Sean, Huang, Zhiheng, Karpathy, Andrej, Khosla, Aditya, Bernstein, Michael S., Berg, Alexander C., and Fei-Fei, Li. Imagenet large scale visual recognition challenge. *CoRR*, 2014.
- Russell, Bryan C., Torralba, Antonio, Murphy, Kevin P., and Freeman, William T. Labelme: A database and web-based tool for image annotation. *International Journal of Computer Vision*, 77(1-3):157–173, 2008. ISSN 0920-5691. doi: 10.1007/s11263-007-0090-8.
- Socher, Richard, Perelygin, Alex, Wu, Jean Y, Chuang, Jason, Manning, Christopher D, Ng, Andrew Y, and Potts, Christopher. Recursive deep models for semantic compositionality over a sentiment treebank. Citeseer, 2013.
- Von Ahn, Luis and Dabbish, Laura. Labeling images with a computer game. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 319–326. ACM, 2004.
- Wang, Yue, Teoh, Eam Khwang, and Shen, Dinggang. Lane detection and tracking using b-snake. *Image and Vision computing*, 22(4):269–280, 2004.
- World Health Organization. *Global status report on road safety*. 2013.
- Yenikaya, Sibel, Yenikaya, Gkhan, and Dven, Ekrem. Keeping the vehicle on the road. *CSUR*, 46(1):1–43, 2013. doi: 10.1145/2522968.2522970.