

Autonomous Navigation Using Localization Priors, Sensor Fusion, and Terrain Classification

Zachariah Carmichael, Benjamin Glasstone, Frank Cwitkowitz, Kenneth Alexopoulos, Robert Relyea, Raymond Ptucha; Rochester Institute of Technology; Rochester, New York/USA

Abstract

Autonomous robots and self-driving vehicles require agents to learn and maintain accurate maps for safe and reliable operation. We use a variant of pose-graph Simultaneous Localization and Mapping (SLAM) to integrate multiple sensors for autonomous navigation in an urban environment. Our methods efficiently and accurately localize the agent across a stack of maps generated from different sensors across different periods of time. To incorporate a priori localization data, we account for the discrepancies between LiDAR observations and publicly available building geometry. We fuse data derived from heterogeneous sensor modalities to increase invariance to dynamic environmental factors, such as weather, luminance, and occlusions. To discriminate traversable terrain, we employ a deep segmentation network whose predictions increase the confidence of a LiDAR-generated cost map. Path planning is accomplished using the Timed-Elastic-Band algorithm on the persistent map created through SLAM. We evaluate our method in varying environmental conditions on a large university campus and show the efficacy of the sensor and map fusion.

Keywords: sensor fusion, terrain classification, probabilistic mapping, laser odometry, autonomous navigation

Introduction

Only a few decades ago the intelligent systems of today would have been considered far-fetched. For example, autonomous navigation necessitates a means of distilling semantic information from the outside world, making informed decisions of the next course of action, and maintaining a robust and continuous estimate of external states. The safety-critical challenge can tolerate only a marginal amount of error and must overcome trade-offs between feasibility, safety, and efficiency.

Autonomous vehicles utilize a combination of high-precision GPS and multiple sensors alongside various algorithms to produce highly accurate maps. The usage of simultaneous localization and mapping (SLAM) methodologies along with map priors can further improve upon the quality of generated maps and reduce localization error. Frequently, SLAM-based self-driving systems are evaluated under restricted operating conditions with limited variance in weather, luminance, and sensor occlusion. However, as autonomous navigation is becoming more attainable, the importance of robust performance across disparate real-world conditions must be emphasized.

Our work focuses on enhancing the robustness of autonomous transportation on a robotic golf cart (Figure 1). The fusion of multiple sensors of various modalities, as well as a priori topographic data, aid in this increase in resiliency. Furthermore,



Figure 1. The RIT autonomous people mover, the "Tiger Taxi."

a segmentation network augments the path planning capability of our autonomous system. We perform experiments over a variety of seasons, weather conditions, and times of the day within a large-scale university environment, which contains a diverse set of roadways, sidewalks, and dynamic obstacles.

Background

Autonomous driving has received increasing attention for several decades. Interest is driven largely by the vision of safer and further unified transportation. The high-level goals of autonomous driving can be decomposed into several smaller tasks, *i.e.* mapping, localization, path planning, and path execution [1]. Deployable autonomous systems must operate in real-time, a constraint which adds further complexity by demanding efficient yet accurate solutions to these various tasks.

The 2007 DARPA Urban Challenge indicated that autonomous vehicle technology was a realizable concept. Several novel challenges were prompted as many common scenarios were excluded from the event, but the influx in interest in the area accelerated progress nonetheless. Today, autonomous vehicles are on the forefront, as automobile manufacturers begin to incorporate new self-driving technology into their products and further invest in research [2, 3].

Various new research paths show promise towards fully realizing autonomous driving. Firstly, the inclusion of localization priors in a navigation framework can improve greatly the global estimate of an agent. These methods are especially appealing in SLAM systems as they mitigate directly the requisite for a loop-closure method. There have been several approaches proposed to incorporate such a priori data. Egocentric matching with an aerial image has been proposed by several others. Aerial im-

agery contains rich features (*e.g.* buildings, roads, vegetation) that can be represented similarly by either camera or laser scanner data. In [4], walls are extracted from robot-centric imagery and matched with lines extracted from aerial imagery, whereas [5] extracts and compares road-level salient features from both aforementioned visual sources. Another line-matching algorithm is proposed in [6] but utilizes walls accumulated by and segmented from LiDAR scans. A laser scanner is used to create grayscale re-emission maps in [7] which are in turn compared with aerial imagery by normalized mutual information. While these methods have proven themselves to improve global localization accuracy, they fail when sensors are unable to recognize the features of the aerial map, whether it is due to occlusion, noise, or luminance differences.

Topographical geometric data is richer and less noisy than aerial imagery data, thus it should be favored when available. In [8], georeferenced and publicly available road data is taken from OpenStreetMap. A classifier determines whether robot sensors observe a given road network. However, this approach relies heavily on road networks, which prevents the extension to off-road (*e.g.* pedestrian or non-vehicular pathways) and ignores other useful world features provided by OpenStreetMap. [9, 10] take another approach which instead makes use of building geometry provided by OpenStreetMap and directly uses the data as edge constraints for the optimization of the SLAM pose graph. We extend [10] to this work.

All autonomous systems must sense their environment for the purpose of either creating a map or localizing within one. Typically, sensor configurations with more redundancy allow a system to make more refined predictions over time, especially when one or more sensors begin to fail. An example of localization is detailed in [11] where localization accuracy with respect to GPS-only systems is improved significantly by combining the data from various sensors. The data is used to generate infrared reflectivity maps for comparison to real-time LiDAR measurements. This work was subsequently improved upon by modeling the maps as a grid-like collection of Gaussian distributions representing the infrared reflectivity [12].

State-of-the-art methods for visual scene understanding utilize convolutional neural networks (CNNs) for feature extraction and class prediction. Initial methods employed models adapted from object recognition problems that use the CNN hierarchy to extract rich features [13]. These features can be upsampled back to the initial resolution, resulting in a mask where each pixel represents a class prediction. Furthermore, it was shown in [14] that the upsampling process could be improved by using a symmetric decoder to fine-tune predictions. To achieve real-time predictions from the fully-convolutional model, speed is taken into consideration by reducing the number of parameters in the model and downsampling the input early. This act lowers the computational requirements of the model and allows it to be deployed on a single GPU. The model adapted to this work is the Efficient Network (ENet) [15] since it is capable of making scene predictions at a rate of at least 30 frames per second.

Prior works have focused on accuracy and consistency as an agent navigates within constrained conditions. Our research focuses on the non-ideal environmental states which are present in everyday scenarios. By developing an algorithm to traverse a large university campus across several seasons, we demonstrate

standard SLAM algorithms require additional mechanisms and data in order to handle significantly more variation. Multimodal sensory input, *a priori* topological information, and a segmentation network all contribute towards such generalization. Our evaluation environment contains heavy pedestrian traffic and severe weather patterns which make it an ideal location to demonstrate the robustness of our SLAM augmented navigation system.

Methodology

Sensor Fusion

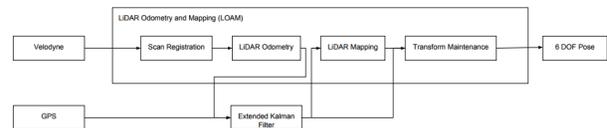


Figure 2. Block diagram of the sensor fusion model.

The basis of our localization relies on laser odometry to deduce cart motion from time-varying LiDAR sensor data. LiDAR SLAM is popular among localization methods as the LiDAR sensors yield accurate and robust depth information of an environment. The sensors provide reflectivity of surfaces as well as distance from them, which can be used to construct dense 3D graphs of an environment over time. We build off of the laser odometry method, LiDAR Odometry and Mapping (LOAM) [16], which has scored first on the KITTI odometry benchmark suite [17] for LiDAR-only methods, in this work. In order to produce high-precision and real-time state estimation, LOAM uses multiple nodes which operate in parallel. The laser odometry phase produces a rough motion estimation based on two sequential high-frequency LiDAR scans. To further tune this estimate, LOAM maintains a localized LiDAR map of the current drive and aligns LiDAR scans with this map at a lower frequency.

To further augment the LiDAR odometry model, we fuse the pose estimates from all available sensors via an extended Kalman filter (EKF) and produce a full 6-DOF pose. We elect to exploit the multi-nodal structure of LOAM by inserting the EKF between the LiDAR odometry and LiDAR mapping nodes, as illustrated by Figure 2. Fusing our additional sensors between LOAM's two state estimation nodes further ensures smooth and continuous states throughout a drive.

To mitigate the need for loop-closure detection and closing, we fuse a GPS sensor into our localization pipeline, minimizing long-term drifts with globally referenced poses. However, the GPS poses a drawback that hurts odometry estimation: the frequent discontinuity between consecutive measurements. These inconsistencies are present especially in urban canyons where GPS receivers become susceptible to shadowing and multipath effects [18]. The Rochester Institute of Technology (RIT) campus bears many tall buildings and underpasses which causes occasional GPS performance degradation or even complete failure.

For additional heading and acceleration information, an inertial measurement unit (IMU) is fused into our system. IMU performance is degraded significantly by sub-freezing temperatures and magnetic interference, both of which are present in our testing environment. To mitigate these issues, the Haversine formula is employed to aid in corrections of heading information using EKF-filtered GPS data.

Exploiting Building Geometry

We extend the ideas presented in [10] to this work – building geometry of the local and surrounding areas are pulled from OpenStreetMap to augment the localization process. The primary goal is to account for the error estimated between LiDAR observations and known buildings, which can be distinguished easily by the sensor. The error can thus be used to produce a homogeneous transformation to correct the heading and position of an agent. We integrate this pose estimate within an EFK as described in the Sensor Fusion Section. The following outlines this method.

1. Convert the publicly available building geometry to 2D georeferenced point clouds.
2. Apply the Ramer-Douglas-Peucker (RDP) algorithm to reduce the raw LiDAR scan into a polyline. For 3D LiDAR (e.g. the Velodyne VLP-16), a virtual 2D scan can be constructed using the method proposed in [19] and used in [10]. A higher RDP ϵ is acceptable as the majority of salient points belong to buildings with flat surfaces.
3. Segment the polyline into individual lines (buildings) based on potential line length and number of laser points. The linearity of the points can also be used as a threshold for consideration.

This method can be extended easily to 3D geometry, however, height information of buildings is fairly sparse within the OpenStreetMap database, which necessitates another data source or manual annotation. Building altitude can also be estimated given a number of floors. This work did not use 3D geometry.

Navigation & Obstacle Avoidance

Our system’s framework for autonomous navigation uses the ROS navigation stack [20, 21]. The navigation stack maintains local and global cost maps, which are used for the construction and temporal execution of valid paths. The sensor data is used to update the cost maps and detect anything that might obstruct the golf cart. Obstacles are placed onto the cost maps, and the cost of their surroundings is inflated to maintain a comfortable distance while driving.

Our localization system is utilized by the ROS navigation stack for odometry information. The output of our localization system is a transform from a pre-specified GPS coordinate to the current location. The transformation is utilized by OctoMap [22] to generate a 3D map of the available terrain as illustrated in Figure 3. OctoMap makes use of octrees, an efficient recursive structure for 3D data, for probabilistic mapping which aid in providing knowledge of the static obstacles that the golf cart needs to avoid. Once projected into two dimensions, this static information is handed off to the global planner, which determines the best path between the cart’s current location and a chosen goal location. This path is found using Dijkstra’s algorithm to search through the grid-like structure of a cost map, each voxel representing a node in the graph. Ultimately, if a valid path is found, no other path will lead to the target destination without incurring additional cost.

A local planner determines the optimal trajectory to follow that allows for the completion of the next set of sub-goals from the global plan. The trajectory is found using the Timed-Elastic-Band algorithm [23, 24, 25], which takes into account velocity and acceleration constraints and the Ackermann steering configuration

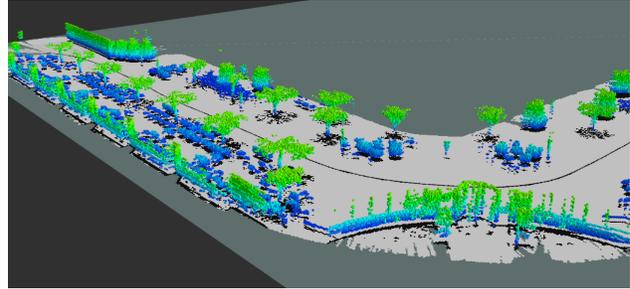


Figure 3. OctoMap 3D point cloud representation of the beginning of our odometry data set.

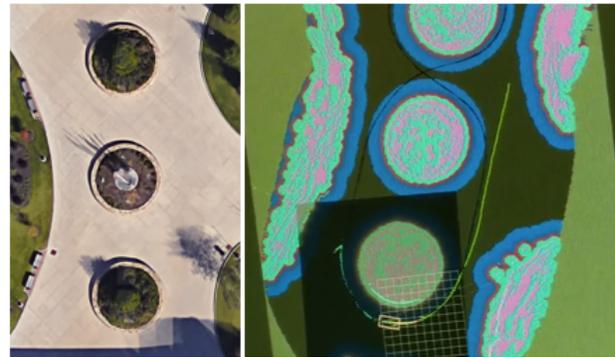


Figure 4. Representative local cost map of the corresponding region of the Rochester Institute of Technology campus. The cyan and pink indicate regions of high cost, and a green line indicates the path taken by the golf cart as produced by the local planner.

of the golf cart. The resulting path is optimized additionally for distance and the avoidance of objects which are present in the local cost map but not the global cost map. An example cost map and path are shown in Figure 4.

When a feasible trajectory is available, the local planner generates actuator commands which are fed to an on-cart Arduino microcontroller. The original commands from the navigation stack are modified such that they comply with the control of vehicles which use Ackermann steering, *i.e.* our golf cart. The commands are then converted into PWM signals which are passed to the braking, steering, and throttle mechanisms to control the golf cart.

Terrain Classification

LiDAR is capable of detecting most objects encountered, but due to its placement, it is unable to sense objects very close to the ground, such as curbs. The LiDAR is also incapable of differentiating between common ground surfaces, *e.g.* between roads and grass. To supplement the LiDAR, a roof-mounted RGB camera is used to classify the terrain directly in front of the golf cart. Figure 5 illustrates the terrain classification pipeline in which we employ a CNN to classify traversable regions. Before classification, video frames are sampled from the RGB camera at up to 30 Hz and are downsampled to a resolution of $3 \times 360 \times 600$. Each frame is then passed through a pixel-wise semantic segmentation model – the fully-convolutional ENet CNN [15] is used to perform such segmentation. Each pixel of the classification indicates a safe or unsafe area to drive. This prediction is nonlinearly

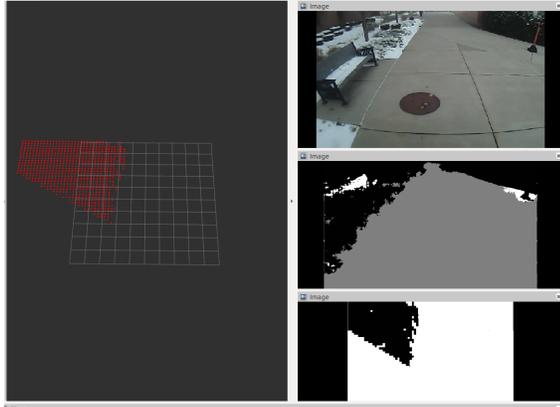


Figure 5. The terrain classification pipeline. Top right: the input frame from the monocular RGB camera. Middle right: the output prediction from the CNN (grey=sidewalk, white=road, black=unsafe). Bottom right: perspective warped prediction (white=safe, black=unsafe). Left: 3D occupancy grid computed from the warped prediction.

remapped to remove lens distortion and the prediction is warped to convert the perspective from a single point to a four-point birds-eye view based on the camera intrinsics. The warped prediction is then resized using the method described in [26] to convert the prediction into a distance map. Multiple classes, e.g. sidewalks and roads, are combined into a single “safe” probabilities and compared against the “unsafe” probabilities to determine final pixel-wise predictions. Finally, the prediction is converted into safe and unsafe 3D point clouds using the distance information, and are overlaid on the local cost map.

Terrain Classification Data Set

There are many pixel-wise semantic segmentation data sets which are publicly available, including MS-COCO and Cityscapes, but they fail to generalize to the varying conditions present in our evaluation environment. To mitigate this shortcoming, a custom data set was produced from several video captures taken around the campus across varying traffic densities, both traversable and impassable terrain textures, seasons, and times of the day. Frames from the video were annotated manually for the classes: road, sidewalk, and unsafe/unlabeled. The data set consists of a total of 1,500 training samples and 350 validation samples.

Experimental Results

We implement our autonomous solution using the ROS framework [20] primarily using the Robot Localization [27], Navigation Stack [21], and LOAM [16] packages with modification to source as necessary.

Localization Results

We evaluate the proposed localization method on a 3,700-meter path around the RIT campus, as shown in Figure 6. The georeferenced ground truth is corrected manually based on raw GPS readings and visual references from both cart-centric and aerial viewpoints. The raw GPS values drift significantly in several areas, such as through the urban canyons formed by the eastern dormitories or the western academic buildings.



Figure 6. The ground truth of the odometry data set on the Rochester Institute of Technology campus.

Results are quantified by two different methods. The first method was based on the methodology proposed alongside the KITTI odometry data set [17]. This error metric involves dividing the run into sections of varying lengths. For our run we chose section lengths 100, 200, 300, 400, 500, 600, 700, 800, 900, 1000, 1500, 2000, 2500, and 3000 meters. We compute the translational error between the beginning and the end of the run with a rotation-invariant method. The second method finds the nearest-neighbor between the ground truth and the evaluated poses and determines the Euclidean distance between the two points, ensuring a one-to-one and ordered mapping constraint is fulfilled. Table I shows the increase in performance across all metrics between LOAM and our proposed fusion method.

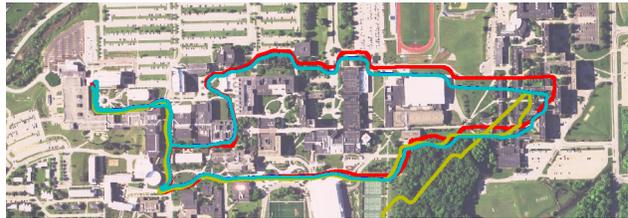


Figure 7. Plot of Localization methods vs GPS. Red - Ground Truth. Blue - NavSatTransform. Cyan - Sensor Fusion. Yellow - LOAM.



Figure 8. Close up plot of Localization methods vs GPS. Red - Ground Truth. Blue - NavSatTransform. Cyan - Sensor Fusion. Yellow - LOAM.

Figure 7 shows the estimated trajectory followed by the cart across all compared methods. On our data set, LOAM struggled to correctly estimate the trajectory around the halfway point of the drive. Since there is no loop-closure built-in to LOAM, the error grew continuously until the end of the drive. When fused with additional sensory and topographic information, our trajectory becomes much closer to the ground truth. Our fused method struggled around the tall building on the east side of campus; however, as shown in Figure 8, the cyan path indicating our proposed method mitigates the discontinuities present in raw GPS data without sacrificing global precision.

Table I: Localization results on the proposed data set. Distance-labeled headers indicate mean relative error over sequences of the specified length.

Method	100 m	1000 m	2000 m	3000 m	Mean Relative Error	Mean Squared Error
LOAM	.528%	.227%	.183%	.158%	.288%	104.0 m
Fusion	.438%	.095%	.050%	.024%	.166%	7.190 m

Terrain Classification Results

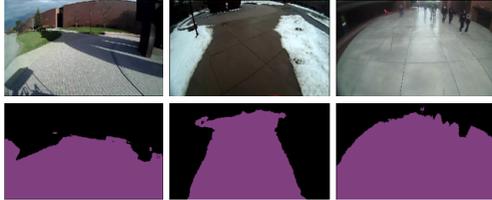


Figure 9. Terrain classification predictions in various conditions. From left to right: various terrains occluded by shadows, terrain in snowy conditions, and reflective terrain saturated in rain with pedestrian traffic.

Figure 9 illustrates ENet performance across multiple places on campus and in various environmental conditions. The variance within the data set and segmentation capability of the proposed CNN allowed for the robust differentiation of terrain in such conditions of the RIT campus. The segmentation network achieved an 86.37% mean intersection-over-union (IoU) on the validation set. The network is able to perform predictions in 16 milliseconds on average using an NVIDIA GTX 1080 Ti.

Conclusion

The inclusion of supplementary data derived from multiple sensors and known topographic sources improves upon the robustness and stability of localization systems. Furthermore, neural classification of diverse terrain and dynamic obstacles enhances the vitality of path planning. Through sensor fusion, laser alignment with known buildings, and a visual segmentation network, we realize these elements and validate their efficacy. The future success of autonomous systems, and thus widespread adoption, will depend on the productive amalgamation of complementary sensors, the exploitation of *a priori* data sources, and integration of robust machine learning algorithms. In future work, we will additionally use road networks to increase localization confidence in featureless areas, and improve upon navigation via amended semantic segmentation, considering further specific dynamic classes.

References

- [1] T. Luettel, M. Himmelsbach, and H.-J. Wuensche, "Autonomous ground vehicles-concepts and a path to the future.," *Proceedings of the IEEE*, vol. 100, no. Centennial-Issue, pp. 1831–1839, 2012.
- [2] C. Berger and B. Rumpel, "Autonomous driving-5 years after the urban challenge: The anticipatory vehicle as a cyber-physical system," *arXiv preprint arXiv:1409.0413*, 2014.
- [3] M. Aeberhard, S. Rauch, M. Braham, G. Tanzmeister, J. Thomas, Y. Pilat, F. Homm, W. Huber, and N. Kaempchen, "Experience, results and lessons learned from automated driving on germany's highways," *IEEE Intelligent Transportation Systems Magazine*, vol. 7, no. 1, pp. 42–57, 2015.
- [4] K. Y. K. Leung, C. M. Clark, and J. P. Huissoon, "Localization in urban environments by matching ground level video images with an aerial image," in *2008 IEEE International Conference on Robotics and Automation*, pp. 551–556, May 2008.
- [5] M. Noda, T. Takahashi, D. Deguchi, I. Ide, H. Murase, Y. Kojima, and T. Naito, "Vehicle Ego-Localization by Matching In-Vehicle Camera Images to an Aerial Image," in *Computer Vision ACCV 2010 Workshops, Lecture Notes in Computer Science*, pp. 163–173, Springer, Berlin, Heidelberg, Nov. 2010.
- [6] H. Roh, J. Jeong, and A. Kim, "Aerial Image Based Heading Correction for Large Scale SLAM in an Urban Canyon," *IEEE Robotics and Automation Letters*, vol. 2, pp. 2232–2239, Oct. 2017.
- [7] L. d. P. Veronese, E. d. Aguiar, R. C. Nascimento, J. Guivant, F. A. A. Cheein, A. F. D. Souza, and T. Oliveira-Santos, "Re-emission and satellite aerial maps applied to vehicle localization on urban environments," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4285–4290, Sept. 2015.
- [8] P. Ruchti, B. Steder, M. Ruhnke, and W. Burgard, "Localization on OpenStreetMap data using a 3d laser scanner," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5260–5265, May 2015.
- [9] O. Vysotska and C. Stachniss, "Exploiting building information from publicly available maps in graph-based SLAM," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4511–4516, Oct. 2016.
- [10] O. Vysotska and C. Stachniss, "Improving SLAM by Exploiting Building Information from Publicly Available Maps and Localization Priors," *ISPRS Journal of Photogrammetry, Remote Sensing and Geoinformation Science*, vol. 85, pp. 53–65, Feb. 2017.
- [11] J. Levinson, M. Montemerlo, and S. Thrun, "Map-based precision vehicle localization in urban environments.," in *Robotics: Science and Systems*, vol. 4, p. 1, Citeseer, 2007.
- [12] J. Levinson and S. Thrun, "Robust vehicle localization in urban environments using probabilistic maps," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pp. 4372–4378, IEEE, 2010.
- [13] C. Farabet, C. Couprie, L. Najman, and Y. LeCun, "Learning hierarchical features for scene labeling," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1915–1929, 2013.
- [14] V. Badrinarayanan, A. Handa, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for robust semantic pixel-wise labelling," *CoRR*, vol. abs/1505.07293, 2015.
- [15] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello, "Enet: A deep neural network architecture for real-time semantic segmentation," *CoRR*, vol. abs/1606.02147, 2016.
- [16] J. Zhang and S. Singh, "Loam: Lidar odometry and mapping in real-time.," in *Robotics: Science and Systems*, vol. 2, p. 9, 2014.
- [17] M. Menze and A. Geiger, "Object scene flow for autonomous vehicles," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.
- [18] P. Misra and P. Enge, "Global positioning system: signals, measurements and performance second edition," *Massachusetts: Ganga-Jamuna Press*, 2006.
- [19] O. Wulf, K. O. Arras, H. I. Christensen, and B. Wagner, "2d map-

- ping of cluttered indoor environments by means of 3d perception,” in *IEEE International Conference on Robotics and Automation*, vol. 4, pp. 4204–4209, IEEE; 1999, 2004.
- [20] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, “Ros: an open-source robot operating system,” in *ICRA workshop on open source software*, vol. 3, p. 5, Kobe, Japan, 2009.
- [21] R. L. Guimarães, A. S. de Oliveira, J. A. Fabro, T. Becker, and V. A. Brenner, “Ros navigation: Concepts and tutorial,” in *Robot Operating System (ROS)*, pp. 121–160, Springer, 2016.
- [22] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, “OctoMap: An efficient probabilistic 3D mapping framework based on octrees,” *Autonomous Robots*, 2013. Software available at <http://octomap.github.com>.
- [23] C. Rösmann, W. Feiten, T. Wösch, F. Hoffmann, and T. Bertram, “Trajectory modification considering dynamic constraints of autonomous robots,” in *Robotics: Proceedings of ROBOTIK 2012; 7th German Conference on*, pp. 1–6, VDE, 2012.
- [24] C. Rosmann, W. Feiten, T. Wösch, F. Hoffmann, and T. Bertram, “Efficient trajectory optimization using a sparse model,” in *Mobile Robots (ECMR), 2013 European Conference on*, pp. 138–143, IEEE, 2013.
- [25] C. Rösmann, F. Hoffmann, and T. Bertram, “Kinodynamic trajectory optimization and control for car-like robots,” in *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on*, pp. 5681–5686, IEEE, 2017.
- [26] S. Tuohy, D. O’Cualain, E. Jones, and M. Glavin, “Distance determination for an automobile environment using inverse perspective mapping in opencv,” 2010.
- [27] T. Moore and D. Stouch, “A generalized extended kalman filter implementation for the robot operating system,” in *Proceedings of the 13th International Conference on Intelligent Autonomous Systems (IAS-13)*, Springer, July 2014.

Author Biography

Zachariah Carmichael is a fifth-year B.S./M.S. student in computer engineering at the Rochester Institute of Technology. He is a graduate research assistant in the RIT Neuromorphic AI Lab with a research focus on theoretical machine learning, brain-inspired algorithms, and model interpretability. He will pursue a Ph.D. in computer science following his graduation from the Rochester Institute of Technology.

Benjamin Glasstone received his B.S. in computer engineering from the Rochester Institute of Technology in 2018. His interests include machine learning, computer vision, and applications of such within the field of robotics. Ben hopes to continue his work in autonomous driving in the industrial sector.

Frank Cwitkowitz is a fifth-year B.S./M.S. student studying computer engineering at the Rochester Institute of Technology. His research interests include music information retrieval, signal processing, machine learning, and computer vision. He will pursue a Ph.D. in music information retrieval following his graduation from the Rochester Institute of Technology.

Kenneth Alexopoulos is a fifth-year B.S./M.S. student in the computer engineering department at the Rochester Institute of Technology. His interests include machine learning, deep learning, computer vision, and autonomous driving. Kenneth is a member of the RIT Machine Intelligence Lab and hopes to work in the autonomous driving field.

Robert Relyea is a first-year computer engineering M.S. student at the Rochester Institute of Technology. He obtained his B.S. in computer

engineering at the Rochester Institute of Technology in 2018. His research focus includes applications of computer vision and deep learning techniques to robot perception and localization.

Raymond Ptucha is an Assistant Professor in Computer Engineering and the Director of the Machine Intelligence Laboratory at the Rochester Institute of Technology. His research includes machine learning, computer vision, and robotics, with a specialization in deep learning. Ray was a research scientist with the Eastman Kodak Company where he worked on computational imaging algorithms and was awarded 31 U.S. patents. He earned a Ph.D. in computer science from RIT in 2013. Ray was awarded an NSF Graduate Research Fellowship in 2010 and his Ph.D. research earned the 2014 Best RIT Doctoral Dissertation Award. Ray is a passionate supporter of STEM education, an NVIDIA-certified Deep Learning Institute instructor, and the Chair of the Rochester area IEEE Signal Processing Society.

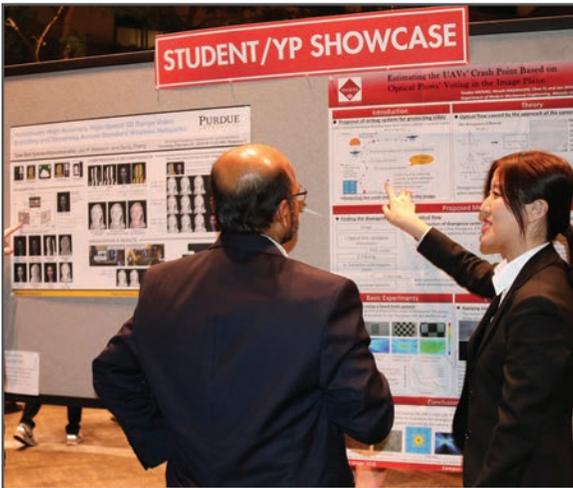
JOIN US AT THE NEXT EI!

IS&T International Symposium on

Electronic Imaging

SCIENCE AND TECHNOLOGY

Imaging across applications . . . Where industry and academia meet!



- **SHORT COURSES • EXHIBITS • DEMONSTRATION SESSION • PLENARY TALKS •**
- **INTERACTIVE PAPER SESSION • SPECIAL EVENTS • TECHNICAL SESSIONS •**

www.electronicimaging.org

