2010 IEEE International Conference on Robotics and Automation
Anchorage Convention District
May 3-8, 2010, Anchorage, Alaska, USA

Inferring Geometry from Imagery - Enabling High Speed Traversal

Gregory Broten and David Mackay
Defence Research and Development Canada – Suffield
{*firstname.lastname*}*@drdc-rddc.gc.ca*

***Abstract*— Robotic vehicles operating in outdoor environments, commonly referred to as unmanned ground vehicles (UGV), are confronted with unstructured/semi-structured environments that are variable in nature. The geographical location significantly influences the environment's appearance, there are longer term seasonal cycles, as well as immediate affects such as the weather and lighting conditions. This environmental diversity has long caused researchers considerable grief, as developing a generalized terrain classification algorithm has proven to be very difficult. Researchers have skirted this problem by relying upon ranging sensors and constructing $2\frac{1}{2}$D or, more recently, 3D world representations. Although geometric representations have been used extensively orientation errors limit the lookahead distance. An important UGV capability is high speed traversal, hence extending the lookahead distance that in turn increases the maximum attainable vehicle speed is an active area of research. This focus on high speed traversal in variable environments has pushed researchers to investigate techniques that allow learning from experience, in a more human like manner. This paper presents Defence R&D Canada – Suffield's progress in extending a $2\frac{1}{2}$D world representation using vision and learning to infer geometry.**

I. Introduction

Early research in mobile robot navigation, targeting indoor environments, focused on the construction of geometric world representations [1], [2]. This approach was later adapted to unstructured outdoor environments [3], [4]. The geometric representation approach produced robust and reliable results upon which researchers could implement obstacle avoidance and navigation capabilities [5], [6], [7]. Although a powerful technique, the geometric world representation is not without limitations. The most significant limitations, the shallow incidence angles for long ranges and the orientation error, govern the accuracy of range registration. Thus, the geometric approach with its limited *lookahead* distance produces a myopic world view [8], [9], [10]. This near sightedness of typically 20 m hinders navigation and limits the maximum attainable speed to $1-4$ $\mathrm{m/s}$, significantly impairing a vehicles utility. The maximum speed limitation, resulting from this approach, has been addressed in structured environments. Through the use of imagery, approaches have been devised that allow for high speed road traversal [11], [12], [13], [14], though these specific implementations do not adapt to environmental changes. Techniques have been developed that allow for the traversal of unstructured roads such as trails or paths. These include visual techniques to estimate the road shape [15], [16], up front learning approaches [17], [18], probabilistic techniques [19] and model based approaches [20]. Researchers have also investigated off road traversal using vision, tele-operator feedback, and supervised learning [21]. Recently, researchers have tackled the problem of environmental variability. These approaches build upon multiple sensing modalities that limit the need for *a priori* knowledge. Successful implementations correlate imagery and range data to infer geometry [22], [23]. A more common approach incorporates self-supervised learning, in which relationships between differing data sources are learned in near real-time, thus, allowing the system to adapt as the environment changes [24], [25], [26], [27]. The need for real-time adaptation has also been identified by the U.S. Defence Advanced Research Project Agency (DARPA) and investigated in the Learning Applied to Ground Robots (LAGR) project. This project targeted perception-based, off-road navigation through the incorporation of learned behaviours [28]. Run as a competition in a manner similar to the DARPA Grand and Urban Challenges, research groups were challenged by fixed trials. This competition based approach, although stimulating friendly competition, limited the relevance of the results. With a narrow focus on perception based autonomy, the baseline hardware was supplied and could not be modified. As a result, the researchers were not allowed to investigate alternative sensing modalities that may have been beneficial. For example, laser ranging devices and multi-spectral cameras where not considered. Additionally, the tight test schedules served as a disincentive to pursue higher risk approaches. Even with these limitations the project demonstrated that self-supervised learning can improve navigation in unstructured terrain [29], [30], [31], [32].

This paper details Defence R&D Canada's (DRDC) online, self-supervised learning algorithm that maps visual features to their inferred geometry. The paper is divided into five sections. Section II explains the motivations that drive this research and Section III details the implementation. Experimental results are provided in Section IV and the report finishes with the conclusions given in Section V.

II. Motivation

DRDC unmanned ground vehicles, similar to the vehicles that competed in the DARPA Grand Challenge [33], [34] operate in unstructured/semi-structured terrain that features roads of various types, flat prairie and semi-rural settings. Given the outdoor terrain and large, mobile vehicles, a key research objective is high speed traversal.

978-1-4244-5040-4/10/$26.00

A. High Speed Traversal

High speed traversal poses unique challenges. Sensor data must be acquired, processed and obstacles identified in real time allowing the UGV to make decisions in a timely manner. Speed significantly complicates this process as the traversal distance is proportional to the vehicle's speed.

1) Processing Time Window: The UGV has a finite time window in which it must acquire sensor data, create a world representation, identify obstacles, chose an action, and then execute that action. Figure 1 shows a graphical representation for a typical time window. The first four slices of the time window, data acquisition, world representation, obstacle identification and action determination, are independent of vehicle speed[1]. The final time slice, the action execution, is dependent upon speed as will be detailed in the following section.

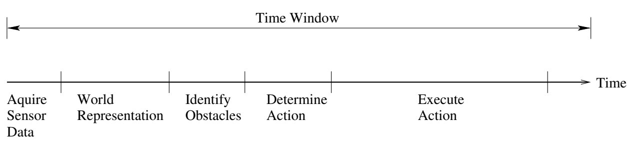


Fig. 1. Process Time Window

B. Action Execution

When confronting an obstacle a UGV relies on two strategies, either manoeuvre to avoid or stop. For high speed traversal the first strategy is limited by the vehicle's dynamic stability and the ability to plan a safe trajectory. Although manoeuvring to avoid obstacles is a desirable and necessary capability, for reasons of safety most UGV's are configured such that an emergency stop will preclude all collisions.

Pivtoraiko et al. [35] have compiled normalized braking force, F_b/mg, data obtained for an off-road vehicle during the CMU PerceptOR program. These tests encompassed vehicle speeds ranging from $1 - 4$ $\mathrm{m/s}$ and a variety of terrain slopes and surfaces. The vehicle controller behaved akin to an anti-lock braking system, not allowing the wheels to lock and thus maintaining steering. The normalized braking force, obtained in these off-road trials, ranged in value from 0.15 to 0.45, much less than typical values reported for passenger vehicle tires on dry asphalt, 0.71 [36] and 0.85 [37]. Stopping times and distances, computed using the PerceptOR normalized braking force data, for a range of vehicle speeds and terrain slopes are presented in Figure 2.

C. Sensing Limitations

In order to make timely decisions the world must be represented at a range that is adequately distant from the vehicle. The maximum sensing range or lookahead distance, $l_{\max}$, shown in Figure 3, is the maximum range at which an obstacle, e.g., a step change in the height of the terrain, can be detected.

[1]Assuming all other configuration parameters, such as the lookahead distance, remain constant.

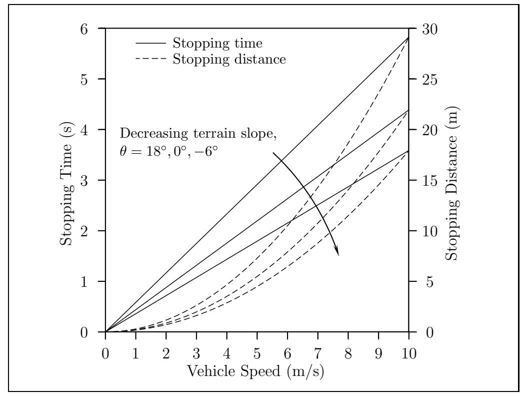


Fig. 2. Stopping times and distances on sloped terrain versus vehicle speed

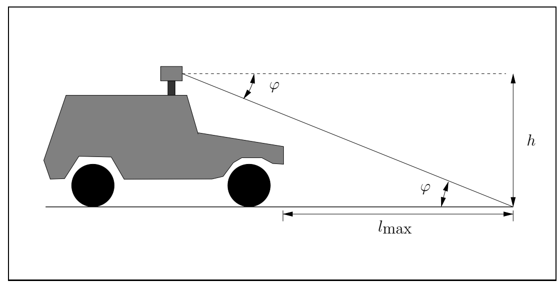


Fig. 3. Laser Mounting Geometry

The Raptor UGV uses a Velodyne HDLaser rangefinder as its primary sensor. Given a typical mounting height, h, of roughly 2 m and a lookahead distance of 30 m, the incidence angle, φ, is $3.8°$. A Novatel HG1700 Span INS provides pose estimation, with an orientation accuracy of $\pm 0.013°$. The Velodyne HDLaser's azimuth accuracy is $\pm 0.09°$. Given a moving platform, finite data acquisition and processing time, interpolating to the exact orientation for each laser scan has additional uncertainties that could degrade this accuracy by a factor of 3 or more. Thus, orientation errors as large as $\pm 0.3°$ are expected. At 30 m an error of $\pm 0.3°$ corresponds to an error in elevation of ± 16 cm, a significant obstacle for the Raptor UGV. Given typical braking distances of $20 - 25$ m at 36 $\mathrm{km/hr}$, as shown in Figure 2, the 30 m sensing range effectively limits the UGV's maximum velocity to under 30 $\mathrm{km/hr}$.

III. Inferring Geometry

A purely geometric representation limits a UGV's lookahead distance to approximately 30 m. This research extends the lookahead distance by inferring geometry through a *near-to-far* learning paradigm. It uses the local data, available from laser rangefinders and visual images, to *learn* the relationships between visual cues and the measured geometry. The unique attributes of this approach include:

- Generalized and self-supervised learning that occurs on-line in real-time,
- Local to global feature mapping, where the local features correspond to geometry and the global features correspond to visual cues,
- The learning implementation has a memory, yet is capable of forgetting and re-learning when applicable, and,
- This implementation can learn trails and paths without requiring human input or supervision.

Inferred geometry is an attractive approach as the resulting inferred geometry terrain map can be analyzed for traversability using existing capabilities.

A. *Visual Images*

A visual image's utility does not degrade with orientation inaccuracy. As can be seen in Figure 4, the road extends off towards the horizon and although the road is obvious to the human observer, autonomous vehicles do not commonly incorporate visual processing capabilities.

Fig. 4. Typical UGV Perspective

Vision researchers have developed numerous methods for decomposing an image into regions or segments. Although this process is relatively straightforward, a purely vision based approach is confronted with interpreting the region's meaning. In structured environments *a priori* knowledge is applied to extract meaning from analyzed images. In outdoor environments the lack of structure, and hence *a priori* knowledge, poses a conundrum for researchers. Using two sensing modalities this conundrum can be surmounted as shown by Dahlkamp et al [25]. Under this approach the trapezoidal region is analyzed flatness and drivability using laser range data. Once identified the region is assumed to be a road and provides data for a computer vision algorithm that classifies the entire image as either drivable or not.

B. *Self-Supervised Inferred Geometry*

This research implements a generalized, self-supervised inferred geometry technique. The essence of this approach is:

- Local features are available that are reliable and relevant,
- Global features, extending beyond the local context, are available but their relevance is uncertain,
- Learn the relationship between the global and local features harnesses the global feature's predictive power by extending the locally relevant features into the global space.

Figure 5 is a graphical representation. The process that *learns* the relationship between the local and global features may take many forms. For on-line self-supervised systems the learning algorithm must have near real-time capabilities.

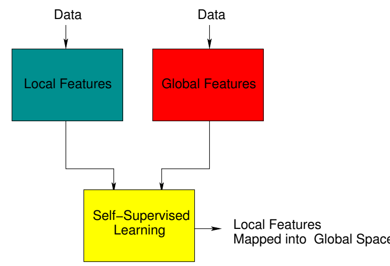


Fig. 5. Self-supervised Systems

1) Locally Weighted Learning: Self-supervised learning's power comes from its ability to learn relationships in real-time, allowing a UGV to adapt to environmental changes as it drives. Although numerous learning techniques exist, many are computationally intensive and not well suited for real-time implementations. Additionally, most learning algorithms struggle with integrating old and new training data. Locally weighted learning [38] is a form of lazy learning or memory-based learning well suited for real-time implementation. This technique defers processing the training data until a query occurs, at which time it searches the database for relevant data. Data relevance is measured by a distance function, with nearby points assigned a higher relevance value. Using the relevant data and a statistically based approximation function, the response to the query is determined. Given that only relevant data is used, the approximation function is local and, hence, relatively efficient. This approach is markedly different from most learning methods, which generally use a computationally expensive, single global model for all training data.

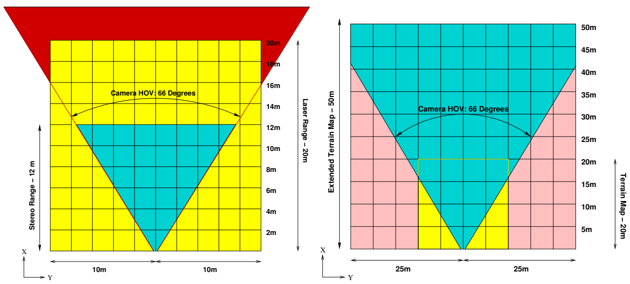
C. *Implementation*

This technique uses *near-to-far* learning where local features are situated close to the vehicle, while the global features are ubiquitous.

1) Local Features: Range data, acquired from a laser rangefinder, are fused into a $2\frac{1}{2}$D terrain map, where the map is a rectangular array of regions and each grid element captures the terrain elevation. These measured terrain elevations are the local features.

2) Global Features: Global features are extracted from imagery. A camera is a suitable sensor for this task as its field of view extends to the horizon, as can be seen in Figure 4. Colour is an obvious choice to represent global features and in this implementation colour is used, but not directly at the image pixel level. Instead the technique uses a colour map, which fuses colour information into a grid map structure where each grid element encodes the mean RGB colour of the given terrain patch.

3) Near-to-Far Learning: Under this paradigm, the relationship between local and global features is learned online and in near real-time. Although this approach is similar to the Dahlkamp [39] technique, it makes no assumptions about which terrain is or isn't a road. It simply learns the relationship between colour and geometry. In order to learn the relationship between these features there must be a correspondence between the two feature types; the left frame of Figure 6 visually shows this correspondence.



(a) Colour Grid Map and Terrain Map

(b) Terrain Map and Inferred Map

Fig. 6. Correspondence between Maps

The terrain map, shown in yellow, is defined by a square $20 \times 20\ \mathrm{m}^2$ grid. The camera's field of view is denoted by the partially obscured red trapezoid. The region shown in cyan is the colour grid map, the region where there is correspondence between the terrain and colour maps. The colour grid map's shape is defined by the camera's horizontal field of view and is limited by the maximum practical depth [2]. Once the relationship between colour and elevation has been learned, an extended $2\frac{1}{2}$D terrain map can be created. The inferred terrain map geometry is displayed in the right frame of Figure 6. The partially obscured region in yellow represents the terrain map derived from range data. The region in cyan represents the extended terrain map that is constructed from inferred geometry. Note, the inferred terrain map has dimensions of $50\ \mathrm{m}^2$.

4) Learning Elevation and Colour Correspondences: The first step in the near-to-far learning process is learning the relationship between colour and elevation. As shown in the right frame of Figure 6, the $2\frac{1}{2}$D terrain map and the colour map have an overlapping region, shown in cyan, where both colour and elevation data are present. Both the terrain map and the colour map are regularly spaced grids of size $0.2 \times 0.2\ \mathrm{m}^2$. The maps are co-registered, thus a given grid index specifies the same terrain patch in either maps. Locally weighted learning determines the relationship between elevation and colour. The steps required to implement this technique are:

- Create a terrain map of elevations,
- Create a colour map that stores mean RGB colour values,
- Update the database of elevation points and the corresponding colour,
- Query the database for the elevation data associated with a given pixel and predict the elevation, and,
- Update the inferred terrain map, using the pixel coordinate and elevation.

a) Update the Database: This locally weighted learning implementation requires a database of colour/elevation tuples. It uses a *k-d* tree for the database, since the *k-d* tree's data storage allows for the speedy retrieval of data within the neighbourhood of the query point. The database is built upon the open source libkdtree++ [40] library. Each grid element in the colour map that has a defined colour, $(\bar{r}_{ij}, \bar{g}_{ij}, \bar{b}_{ij})$, for which the terrain map has an elevation, $\bar{z}_{ij}$, forming a tuple, $[(\bar{r}_{ij}, \bar{g}_{ij}, \bar{b}_{ij}), (\bar{z}_{ij})]$, the basic data element in the *k-d* tree. When new data is acquired the *k-d* tree database grows, as tuples are added. Locally weighted learning delays *learning* until a query is made into the database. Even though a *k-d* tree allows for the fast and efficient retrieval of data, care must be taken to limit the database's size. Unrestricted growth in the database size will exponentially increase the query time, and given a UGV's real-time requirements the database size must be limited to a manageable size. Restricting the colour space is the primary means of managing the database size. A 5 bit colour space has $(2^5)^3 = 32,768$ possible colour combinations, while 6 bit colour encompasses $(2^6)^3 = 262,144$ combinations. Additionally, the database is not allowed to contain duplicate colour space entries. This not only assists in limiting the database size, but also prompts the system to *forget* stale information.

b) Inferring Geometry from Colour: For each camera pixel, (u_i, v_i), determine the (x_i, y_i) coordinate. The coordinate for each pixel is precalculated at startup, assuming a flat world model, and the results are stored in a pseudo-range sensor[3]. Then determine the inferred elevation as shown in Figure 7.

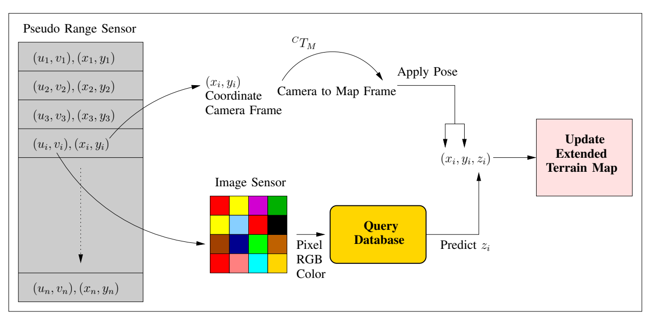


Fig. 7. Inferring Geometry from Colour

Query the *k-d* tree with the pixel (r_i, g_i, b_i) colour and retrieve the estimated elevation. Finally, update the inferred terrain map with the (x_i, y_i, z_i) data. In this manner, the inferred terrain map is populated with elevation values, thus creating a map representation using inferred geometry.

[2] This is determined by the baseline and pixel density, and for the Bumblebee stereo camera this depth is approximately 12 m.

[3] Given the pseudo-range sensor features high range densities for near distances, it can be filtered to reflect a uniform, lower density, hence reducing the computational burden.

IV. Results

A. Setup

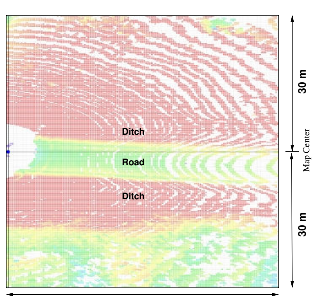
Experiments were conducted using the following setup: the Raptor UGV shown in Figure 4, a Velodyne HDLaser, a Point Grey BumbleBee2 Stereo camera, and a Novatel HG1700 Span INS. For these experiments the real-time data from all devices was logged using the Miro logging facility [41]. This logged data included all laser range data, raw camera images, rectified camera images, disparity images and the vehicle's pose. The logged data was then played back through the geometrical terrain map and an inferred map was created using the inferred geometry process described in Section III.

B. Inferred Geometry

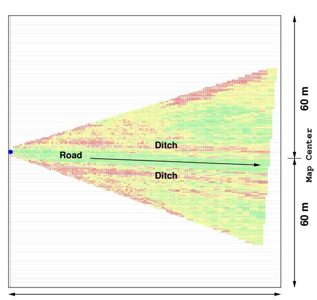
Figure 8 shows a typical outdoor road. This road is characterized by gravel track and vegetation in the centre and at the margins and extends eastward towards the horizon.

Fig. 8. Typical Suffield Track

Using the range data from the laser a $2\frac{1}{2}$D terrain map is constructed. The map extends $30\ \mathrm{m}$ in front of and to either side of the vehicle, as shown in (a) of Figure 9. In this figure, elevation is denoted by colour, where the spectrum red $\rightarrow$ yellow $\rightarrow$ green represents lower to higher elevations. The road, the ditches and the exit are clearly identifiable in the terrain map. The inferred $2\frac{1}{2}$D terrain map, created by mapping image colour to geometry, shown in slide (b) extends the map's range to $60\ \mathrm{m}$. In this map, the road, shown in green, is visible as it extends towards the horizon; the ditches on either side of the road are shown in yellow/red.



(a) $2\frac{1}{2}$D Terrain Map

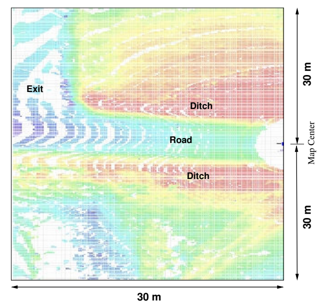


(b) Inferred $2\frac{1}{2}$D Terrain Map

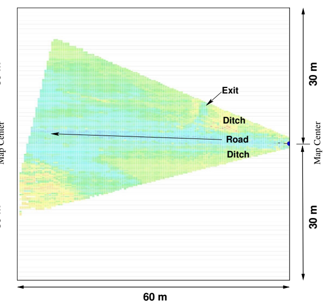
Fig. 9. Road Inferred Terrain Map Results

The slides in Figure 10 show the inferred terrain map results for an exit leaving the road. Although the exit is only partially represented in the terrain map, it is visible in the inferred map. Also of note is that the road and ditches are not well defined under the inferred terrain map.

(a) Image Scene



(b) $2\frac{1}{2}$D Terrain Map



(c) Inferred $2\frac{1}{2}$D Terrain Map

Fig. 10. Trail Inferred Terrain Map Results

Figure 11 shows the inferred terrain map along a trail where the field to the right is elevated and on the left the field is depressed. In this case, although the trail is recognizable, the delineation for ditches and fields are poor. In the image scene green represents both low and high elevations. Since a given elevation can only be associated with a unique colour all green coloured terrain is inferred as elevated, as this is the last association in the *k-d* tree.

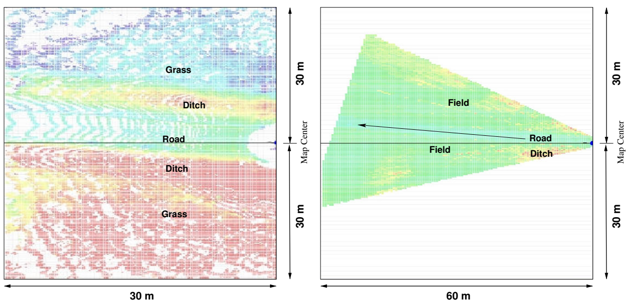
Using a 5 bit colour space, the overall time required to update the database and produce an inferred geometry terrain map is approximately 1.5 sec ± 0.5 sec. Although this time varies depending on the data set, it is close enough to the near real-time performance required for high speed traversal.

V. Conclusions

Defence R&D Canada – Suffield has developed a self-supervised learned trafficability technique that infers geometry from visual imagery. Using visual images and laser range data, this technique learns the relationship between colour and geometry in real-time. This technique's reliance on self-supervision allows the implementation to adapt to environmental changes as the vehicle moves.

Geometric world representations, such as terrain maps, based upon range data, tend to be too myopic to support high speed traversal. For common UGV platforms the interplay between geometry and orientation accuracy limits the maximum lookahead distance to roughly $20 - 30\ \mathrm{m}$. Using inferred geometry an extended terrain map has been created that effectively doubles the lookahead distance. In turn, this enables higher traversal speeds since traversable terrain can

(a) Image Scene



(b) $2\frac{1}{2}$D Terrain Map (c) Inferred $2\frac{1}{2}$D Terrain Map

Fig. 11. Trail Inferred Terrain Map Results

be identified at longer ranges, giving the vehicle more time to execute an appropriate response.

The technique described in this paper uses locally weighted learning to *learn* the relationship between colour and geometry. Using the learned relationship between colour and geometry, this research extended the Raptor UGV's maximum lookahead distance from 30 m to 60 m. Experiments have shown that under this approach, a road can be represented at ranges out to 60 m, even though the laser rangefinder only scanned out to 30 m. Additionally, given the technique's inferred geometry basis, the ability to represent the road was an inherent capability and was not the result of any special or specific assumptions. Inferred geometry will work with other types of environments such as highways, roads, paths or open prairie, as long as there are features with distinguishable colours. The implementation operates under near real-time conditions, and can produce an inferred terrain map approximately every 1.5 sec. Given a vehicle speed of under 10 m/s and a lookahead distance of 60 m, this performance is adequate.

This technique has been implemented on the DRDC Raptor UGV. Thus far, only the ability to create inferred geometry terrain maps has been tested. Future work will investigate techniques that will exploit this longer range map, thus, enabling higher UGV traversal speeds.

The current inferred geometry implementation is dependant on colour differences between features, thus its performance will degrade when distinguishing colours are not present. To reduce the impact of such colour invariant environments, DRDC researchers are adding multi-spectral capabilities to the technique. This will augment the global feature set with additional spectral bands. Additionally, it is expected that lookahead distance can be extended beyond 60 m and that further optimizations will reduce the time period required to generate an inferred terrain map.

References

[1] A. Elfes, "Using occupancy grids for mobile robot perception and navigation," *Computer*, vol. 22, no. 6, pp. 46–57, 1989.

[2] H. Moravec and A. E. Elfes, "High resolution maps from wide angle sonar," in *Proceedings of the 1985 IEEE International Conference on Robotics and Automation*, March 1985, pp. 116 – 121.

[3] A. Kelly, *Intelligent Unmanned Ground Vehicles: Autonomous Navigation Research at Carnegie Mellon*. Kluwer Academic Publishers, 1997, ch. RANGER: Feedforward Control Approach to Autonomous Navigation, pp. 105–144.

[4] M. Hebert, C. Caillas, E. Krotkov, I. Kweon, and T. Kanade, "Terrain mapping for a roving planetary explorer," in *Proceedings of the IEEE International Conference on Robotics and Automation*, May 1989, pp. 997–100.

[5] A. Lacaze, K. Murphy, and M. DelGiorno, "Autonomous mobility for the demo iii experimental unmanned vehicle," in *in Assoc. for Unmanned Vehicle Systems Int. Conf. on Unmanned Vehicles (AUVSI 02*, 2002. [Online]. Available: citeseer.ist.psu.edu/lacaze02autonomous.html

[6] P. Bellutta, R. Manduchi, L. Matthies, K. Owens, and A. Rankin, "Terrain perception for demo iii," in *Proceedings of the 2000 Intelligent Vehicles Conference*, 2000. [Online]. Available: www.cse.ucsc.edu/~manduchi/papers/IVS00.pdf

[7] M. Shneier, H. Tsai, T. Chang, H. Scott, S. Legowik, G. Cheok, C. Gilsinn, and C. Witzgall, "Terrain characterization for trl-6 evaluation of an unmanned ground vehicle," National Institute of Standards and Technology, Tech. Rep., 2004.

[8] G. Broten and J. Collier, "Continuous motion, outdoor, 2 1/2d grid map generation using an inexpensive nodding 2-d laser rangefinder," in *Proceedings of the 2006 IEEE International Conference on Robotics and Automation*, no. 2006-061, Orlando, Fl, May 2006, DRDC Suffield CP, pp. 4240–4245.

[9] P. Sermanet, R. Hadsell, J. Ben, A. N. Erkan, B. Flepp, U. Muller, and Y. LeCun, "Speed-range dilemmas for vision-based navigation in unstructured terrain," in *Proceeding of the 6th IFAC Symposium on Intelligent Autonomous Vehicles*, 2007.

[10] A. Howard, M. Turmon, A. Angelova, L. Matthies, and B. Tang, "Towards learned traversability for robot navigation: From underfoot to the far field," *Journal of Field Robotics*, vol. 11, pp. 1005–1017, 2007.

[11] C. Thorpe, M. Hebert, T. Kanade, and S. Shafer, "Vision and navigation for the carnegie-mellon navlab," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, no. 3, pp. 362–373, May 1988.

[12] T. Jochem, D. Pomerleau, and C. Thorpe, "Vision-based neural network road and intersection detection and traversal," in *IEEE Conference on Intelligent Robots and Systems (IROS '95)*, vol. 3, August 1995, pp. 344 – 349.

[13] D. Pomerleau, *Robot Learning*, 1993, ch. Knowledge-based Training of Artificial Neural Networks for Autonomous Robot Driving.

[14] T. Jochem, D. Pomerleau, and C. Thorpe, "Maniac: A next generation neurally based autonomous road follower," in *Proceedings of the International Conference on Intelligent Autonomous Systems*, February 1993, also appears in the Proceedings of the Image Understanding Workshop, April 1993, Washington D.C., USA.

[15] C. Rasmussen, "Texture-based vanishing point voting for road shape estimation," in *British Machine Vision Conference*, 2004.

[16] Y. Alon and A. F. amd A. Shashua, "Off-road path following using region classification and geometric projection constraints," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2006.

[17] M. Ollis and T. Jochem, "Structural method for obstacle detection and terrain classification," in *SPIE–The International Society for Optical Engineering, Unmanned Ground Vehicle Technology*, 2003.

[18] A. N. Erkan, R. Hadsell, P. Sermanet, J. Ben, U. Muller, and Y. LeCun, "Adaptive long range vision in unstructured terrain," in *Proc. of Intelligent Robots and Systems (IROS)*, 2007.

[19] S. Thrun, M. Montemerlo, and A. Aron, "Probabilistic terrain analysis for high-speed desert driving," in *Proceedings of the Robotics Science and Systems Conference*, G. Sukhatme, S. Schaal, W. Burgard, and D. Fox, Eds., Philadelphia, PA, 2006.

[20] L. Cremean and R. Murray, "Model-based estimation of off-highway road geometry using single-axis ladar and inertial sensing," in *Proceeding on IEEE International Conference on Robotics and Automation*, 2006.

[21] Y. LeCun, U. Muller, J. Ben, E. Cosatto, and B. Flepp, "Off-road obstacle avoidance through end-to-end learning," in *Advances in Neural Information Processing Systems (MIPS)*, 2005.

[22] C. Rasmussen, "Combing laser range, color, and texture cues for autonomous road following," in *Proc. of the International Conference on Robotics and Automation*, 2002.

[23] M. Happold, M. Ollis, and N. Johnson, "Enhancing supervised terrain classification with predictive unsupervised learning," in *Proceedings of Robotics: Science and Systems*, Philadelphia, USA, August 2006.

[24] D. Lieb, A. Lookingbill, and S. Thrun, "Adaptive road following using self-supervised learning and reverse optical flow," in *Proceedings of Robotics Science and Systems*, S. Thrun, G. Sukhatme, S. Schaal, and O. Brock, Eds. Cambridge, MA: MIT Press, 2005.

[25] H. Dahlkamp, A. Kaehler, D. Stavens, S. Thrun, and G. Bradski, "Self-supervised monocular road detection in desert terrain," in *Proceedings of Robotics: Science and Systems*, Philadelphia, USA, August 2006.

[26] D. Kim, S. Oh, , and J. M. Rehg, "Traversability classification for ugv navigation: A comparison of patch and superpixel representations," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, San Diego, CA, Oct 2007.

[27] B. Sofman, E. Lin, J. Bagnell, J. Cole, N. Vandapel, and A. T. Stentz, "Improving robot navigation through self-supervised online learning," *Journal of Field Robotics*, vol. 23, no. 12, December 2006.

[28] L. D. Jackel, E. Krotkov, M. Perschbacher, J. Pippine, and C. Sullivan, "The darpa lagr program: Goals, challenges, methodology, and phase i results," *Journal of Field Robotics*, vol. 23, no. 11-12, pp. 945–973, 2006.

[29] B. Hamner, S. Singh, and S. Scherer, "Llearning obstacle avoidance parameters from operator behavior," *Journal of Field Robotics*, vol. 23, no. 11-12, pp. 1037–1058, 2006.

[30] J. Sun, T. Mehta, D. Wooden, M. Powers, J. Rehg, T. Balch, and M. Egerstedt, "Learning from examples in unstructured, outdoor environments," *Journal of Field Robotics*, vol. 23, no. 11-12, pp. 1019–1036, 2006.

[31] A. Howard, M. Turmon, L. Matthies, B. Tang, A. Angelova, and E. Mjolsness, "Towards learned traversability for robot navigation: From underfoot to the far field," *Journal of Field Robotics*, vol. 23, no. 11-12, pp. 1005–1017, 2006.

[32] J. Albus, R. Bostelman, T. Chang, T. Hong, W. Shackleford, and M. Shneier, "Learning in a hierarchical control system: 4d/rcs in the darpa lagr program," *Journal of Field Robotics*, vol. 23, no. 11-12, pp. 975–1003, 2006.

[33] S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, G. Hoffmann, K. Lau, C. Oakley, M. Palatucci, V. Pratt, P. Stang, S. Strohband, C. Dupont, L.-E. Jendrossek, C. Koelen, C. Markey, C. Rummel, J. van Niekerk, E. Jensen, P. Alessandrini, G. Bradski, B. Davies, S. Ettinger, A. Kaehler, A. Nefian, and P. Mahoney, "Stanley: The robot that won the darpa grand challenge," *Journal of Field Robotics*, vol. 23, no. 9, pp. 661–692, September 2006.

[34] C. Urmson, J. Anhalt, D. Bartz, M. Clark, T. Galatali, A. Gutierrez, S. Harbaugh, J. Johnston, H. Kato, P. L. Koon, W. Messner, N. Miller, A. Mosher, K. Peterson, C. Ragusa, D. Ray, B. K. Smith, J. M. Snider, S. Spiker, J. C. Struble, J. Ziglar, and W. R. L. Whittaker, "A robust approach to high-speed navigation for unrehearsed desert terrain," *Journal of Field Robotics*, vol. 23, no. 8, pp. 467–508, August 2006.

[35] M. Pivtoraiko, A. Kelly, P. Rander, and J. Bares, "Efficient braking model for off-road mobile robots," in *Field and Service Robotics*, Port Douglas, Australia, October 2005.

[36] L. Alvarez, J. Yi, R. Horowitz, and L. Olmos, "Dynamic friction model-based tire-road friction estimation and emergency braking control," *Transactions of the ASME Journal of Dynamic Systems, Measurement, and Control*, vol. 127, pp. 22 – 32, March 2005.

[37] T. Day and S. Roberts, "A simulation model for vehicle braking systems fitted with abs," in *SAE 2002 World Congress*, ser. SAE Technical Paper Series 2002-01-0559, Detroit, Michigan, March 2002, pp. 1 – 19.

[38] C. Atkeson, A. Moore, and S. Schaal, "Locally weighted learning," *Artificial Intelligence Review*, vol. 11, pp. 11–73, 1997.

[39] S. Thrun, M. Montemerlo, H. D. D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, G. Hoffmann, K. Lau, C. Oakley, M. Palatucci, V. Pratt, P. Stang, S. Strohband, C. Dupont, L.-E. Jendrossek, C. Koelen, C. Markey, C. Rummel, J. van Niekerk, E. Jensen, P. Alessandrini, G. Bradski, B. Davies, S. Ettinger, A. Kaehler, A. Nefian, and P. Mahoney, "Stanley: The robot that won the darpa grand challenge," *DARPA Grand Challenge Website*, vol. 23, pp. 661–692, 2006.

[40] M. F. Krafft, "libkdtree++," http://libkdtree.alioth.debian.org/, Dec 2007, c++ I(mplementation of KDTrees.

[41] H. Utz, S. Sablatnog, S. Enderle, and G. Kraetzschmar, "Miro - middleware for mobile robot applications," *IEEE Transactions on Robotics and Automation*, vol. 18, pp. 493–497, June 2002.