Defence Research and Development Canada
Recherche et développement pour la défense Canada

DEFENCE R&D DÉFENSE

World Representation Using Terrain Maps

Enabling High-Speed Navigation

G. Broten, J. Giesbrecht and S. Monckton
DRDC Suffield

Technical Report
DRDC Suffield TR 2005-248
December 2005

Canada

World Representation Using Terrain Maps

Enabling High-Speed Navigation

G. Broten, J. Giesbrecht and S. Monckton
DRDC Suffield

Defence R&D Canada – Suffield

Technical Report

DRDC Suffield TR 2005-248

December 2005

Author

G. Broten

Approved by

D.M. Hanna

HTVSS

Approved for release by

P.A. D'Agostino

DRP Chair

Abstract

This report introduces a new terrain map technique for representing unstructured environments. Using an inexpensive nodding 2-D laser rangefinder this technique captures multiple terrain data sets and optimally fuses the data into a wrappable terrain map. Traditional terrain map implementations scan a terrain patch only once, thus integrate a single data point into the map. The nodding laser, allowing multiple scans of a given terrain patch, must fuse multiple data sets into the terrain map. A moving vehicle, with a limited pose accuracy, demands a data fusion technique that minimizes the effects of pose errors. This research investigated and tested a variance weighted statistical technique to optimally fuse terrain data sets into a terrain map. Simulations and experiments were conducted, which demonstrated the performance of the variance weighted technique as superior to classical statistical methods. DRDC implemented a variance weighted terrain map under the Miro architecture and tested its performance during the Autonomous Land Systems (ALS) demonstration. This terrain map technique performed admirably and its accurate terrain representation allowed the Raptor Unmanned Ground Vehicle to successfully navigate while avoiding obstacles and hazards.

Résumé

Ce rapport présente une nouvelle technique de cartes morphographiques visant à représenter les milieux non structurés. Cette technique peu coûteuse qui utilise un télémètre laser bidimensionnel de balayage en site capture des ensembles multiples de données de terrain et fusionne les données de manière optimale en une carte morphographique avec bouclage automatique. Les implémentations classiques de cartes morphographiques ne balaient une parcelle de terrain qu’une seule fois et n’intègre un qu’un seul point de donnée dans la carte. Le laser de balayage en site permet d’effectuer des balayages multiples d’une parcelle donnée de terrain et doit fusionner des ensembles multiples de données dans la carte morphographique. Un véhicule en mouvement, ayant une exactitude de pose limitée, exige une technique de fusions des données qui minimise les effets des erreurs de poses. Cette recherche a étudié et testé une technique statistique de variance pondérée visant à fusionner de manière optimale les ensembles de données d’un terrain dans une carte morphographique. On a conduit des simulations et des expériences démontrant que le rendement de la technique de variance pondérée est supérieur à celui des méthodes statistiques classiques. RDDC a implémenté une carte morphographique de variance pondérée sous une architecture d’échange multimédia d'objets extraits (MIRO) et a testé son rendement durant les démonstrations des Systèmes de véhicules terrestres autonomes. Cette technique des cartes morphographiques a admirablement fait ses preuves et la représentation exacte du terrain a permis au Véhicule terrestre sans pilote Raptor, de réussir à naviguer tout en évitant des obstacles et des dangers.

This page intentionally left blank.

Executive summary

Background: Unmanned Ground Vehicle (UGV) Research and Development within the Autonomous Land Systems (ALS) project will assist the Canadian Forces (CF) in fulfilling their future mandate. DRDC's Technology Investment Strategy [1] defines an Autonomous Intelligent Systems as "...automated or robotic systems that operate and interact in the complex unstructured environments of the future battlespace". An accurate world representation is a fundamental autonomous intelligence requirements.

An autonomous vehicle (UxV) creates a world representation, analyzes it, and denotes obstacle locations. Thus, ths Uxv can avoid hazards and maneuver through obstruction free terrain. The most common world representation, the terrain map, traditionally uses "single sample" perception devices, or "stop and scan" implementations. DRDC's continuous motion UxV requires a terrain map technique that can fuse multiple terrain samples, acquired under different poses, into a single coherent map.

Principle Conclusions: DRDC researched, proposed and implemented a new and novel terrain map data fusion technique. This variance weighted statistical technique optimally minimizes pose error effects, thus creating a coherent terrain map from multiple terrain scans. The variance weighted statistical technique's performance was compared to a classical statistical technique, and was determined to be superior.

Significance of Results: DRDC implemented a variance weighted terrain map under the MIRO framework. Testing on the Raptor UGV revealed the map's efficient implementation consumed relatively few computational resources. A demonstration, conducted in the fall of 2005, exercised the map's capabilities. During these trials, DRDC's terrain map successfully and adequately represented the world. The map remained egocentric while the Raptor performed translational and rotational maneuvers. The terrain map played a critical role allowing the Raptor UGV to successful avoid obstacles and hazards while autonomously navigating toward a goal.

Future Results: Although the current map implementation supported 3-D data from both laser ranging devices and stereo vision cameras, the ALS demonstration relied exclusively on laser data. DRDC's stereo vision implementation was not sufficiently mature and often produced disparity maps that suffered from fringing effects. Without the high 3-D data densities produced by stereo vision, the terrain map sometimes suffered from data starvation. Future work will investigate means to minimize the impact of fringing effects. The current terrain map, implementing a $2\frac{1}{2}$D world representation, can not represent obstacle and hazards with 3 dimensional properties. Obstacles such as overhangs, tunnels and passageways have height restrictions that can preclude safe navigation. Future research will investigate multiple map implementations that should allow the representation of such 3-D obstacles.

G.S. Broten, J.L. Giesbrecht, S.P. Monckton. 2005. World Representation Using Terrain Maps. DRDC Suffield TR 2005-248. Defence R&D Canada – Suffield.

Sommaire

Contexte : La recherche et développement sur les Véhicules terrestres autonomes au sein du projet des Systèmes de véhicules terrestres autonomes aideront les Forces canadiennes (CF) à exécuter leur mandat futur. La Stratégie d'investissements technologiques [1] définit un Système intelligent autonome comme un « système automatisé ou robotique qui opère et interagit dans les milieux complexes et non structurés des futurs champs de bataille ». Une représentation exacte du monde est un besoin essentiel du domaine de l'intelligence autonome.

Un véhicule autonome crée une représentation du monde, analyse cette dernière et note l'endroit des obstacles. Un tel véhicule est en mesure d'éviter les dangers et de manœuvrer à travers un terrain ne comportant pas d'obstructions. La représentation du monde la plus commune, la carte morphographique, utilise normalement des dispositifs de perception « d'échantillon unitaire » ou des implémentations de « balayage à l'arrêt ». Ces véhicules en motion continuelle de RDDC exigent une technique de carte morphographique pouvant fusionner, dans une carte unique cohérente, de multiples échantillons de terrains acquis sous différentes poses.

Conclusions principales : RDDC a étudié, proposé et implémenté une nouvelle technique de fusion des données dans une carte morphographique. Cette technique statistique à variance pondérée minimise de manière optimale les effets d'erreur à l'arrêt créant ainsi une carte morphographique cohérente à partir de balayages multiples du terrain. On a comparé le rendement de la technique statistique à variance pondérée avec celle d'une technique statistique classique et on a déterminé qu'elle était supérieure à cette dernière.

La portée des résultats : RDDC a implémenté la carte morphographique à variance pondérée dans le cadre d'une architecture d'échange multimédia d'objets extraits (MIRO). Les tests effectués sur le Véhicule terrestre sans pilote Raptor ont révélé que l'implémentation efficace de la carte n'exigeait que peu de ressources calculatrices. Une démonstration conduite durant l'automne 2005 a mis les capacités de la carte à l'épreuve. Durant ces essais, la carte morphographique de RDDC a réussi à produire une représentation adéquate du monde. La carte est demeurée égocentrique alors que le Raptor effectuait des manœuvres de translation et de rotation. La carte morphographique a joué un rôle critique permettant au Véhicule terrestre sans pilote de réussir à éviter les obstacles et les dangers tout en naviguant vers un but de manière autonome.

Les résultats futurs : Bien que l'implémentation actuelle de la carte soutienne des données 3-D provenant à la fois des appareils de télémétrie par laser et des caméras de stéréovision, la démonstration des Systèmes terrestres autonomes dépendait exclusivement des données provenant des lasers. L'implémentation de la stéréovision de RDDC n'était pas suffisamment évoluée et produisait souvent des cartes disparates souffrant d'effets de franges de distorsion. Sans les hautes densités de données 3-D produites par la stéréovision, la carte morphographique souffrait quelquefois de manque de données. Les travaux futurs étudieront les moyens de minimiser l'impact des effets de franges de distorsion. La carte morphologique actuelle, implémentant une représentation du monde en $2\frac{1}{2}$D ne peut pas représenter des obstacles et des dangers ayant des propriétés tridimensionnelles. Des obstacles tels que les encorbellements, les tunnels et les voies de passages ont des restrictions en rapport avec les hauteurs pouvant interdire une navigation sécuritaire. La recherche future étudiera des implémentations multiples de cartes qui devraient permettre la représentation en 3-D des obstacles.

G.S. Broten, J.L. Giesbrecht, S.P. Monckton. 2005. World Representation Using Terrain Maps. DRDC Suffield TR 2005-248. R & D pour la défense Canada – Suffield.

Table of contents

List of figures

List of tables

1. Introduction

To autonomously navigate rough, unstructured outdoor environments an unmanned vehicle (UxV) must create an accurate representation of its world. The most common form for a world representation, the terrain map [1, 2, 3, 4, 5, 6], requires three dimensional sensing. Active and passive ranging sensors have been extensively investigated and used for this purpose. Herbert [7] surveys range sensing technologies including laser rangefinders (LRF), triangulation rangefinders and passive stereo vision.

The data from perception sensors is processed and fused a terrain map, creating coherent world representation. This world represention is a critical element required for global navigation and for obstacle avoidance.

This report investigates terrain maps relying on nodding 2-D laser rangefinder (LRF) data. The quality of a nodding laser's 3-D data depends on the accuracy of the pose estimation, as the conversion from range to 3-D data is especially sensitive to the pose's roll, pitch and yaw components [8]. Roll, pitch and yaw measurement errors can result in 3-D positional errors with magnitudes equal to or larger than the autonomous vehicle's platform size. This research proposes a variance weighted statistical technique to optimally fuse 3-D range data into a terrain map.

This report is organized into 8 Sections. Section 2. defines the terrain map's co-ordinate system. Section3. introduces the wrappable, rotating terrain map concept. Fusing 3-D into the terrain map is covered in Section 4., and Section 5. derives generic terrain map requirements. Section 6. details DRDC's specific terrain map implemention. Experimental results are presented in Section 7.1 and the report finishes with the conclusions presented in Section 8.

2. Definition of Co-ordinate Systems

Any map, used for navigational purposes, requires a co-ordinate system. Under local navigation circumstances DRDC defines and uses an egocentric co-ordinate system. Figure 1 shows this egocentric co-ordinate system. This co-ordinate system defines the X axis as parallel to the forward motion vector and the Z axis along the gravity vector. The Y axis is subsequently defined using the righthand rule. Roll occurs about the X axis, pitch is about the Y axis and yaw occurs about the Z axis.

3. Terrain Maps for Navigation

UxV's employ many navigation strategies based upon a variety of assumptions with respect to their operational environment [9]. Early research followed the sense, model, plan and act paradigm (SMPA) [10, 11], where its environment was structured, predictable and assumed to be known. Real environments are unstructured and

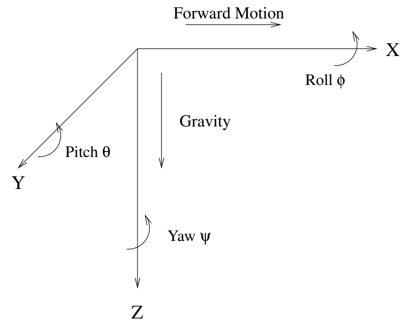


Figure 1: *Map Co-ordinate System*

unpredictable, thus the deliberative SMPA approach performs poorly. The reactive school of thought, pioneered by Brooks [12, 11], used the environment itself as the world model for the robot. This approach resulted in robust behaviour for unstructured environments, but it did not yield useful applications due the inability to direct the behaviour of the robot.

Current robotic research exploits both the SMPA and deliberative approaches by implementing a hybrid strategy. Like the SMPA paradigm, the hybrid approach requires the robot create a world representation. This research, with its control strategy based upon the hybrid approach, uses a grid map, formed by a rectangular array of regions, as its world representation.

The simplest grid map, the *occupancy grid*, represents regions as either occupied or empty and is suitable for flat indoor environments. In unstructured outdoor environments a terrain map [1] [3, 4, 1, 2, 5] captures the terrain elevation of each grid in the map. The grid map, shown in Figure 2, illustrates the physical form and arrangement of a *terrain map*. Each grid element contains the terrain mean elevation, $\bar{z}$, and the elevation variance σ^2.

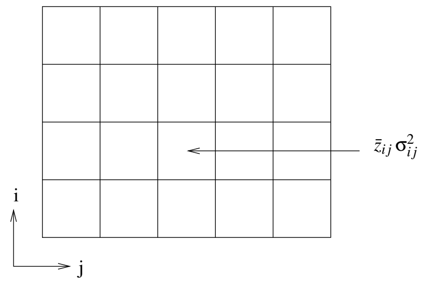


Figure 2: *Terrain Map Representation*

A terrain map is capable of representing obstacles and other features found in outdoor environments such as hills, slopes, bumps and dips.

[1] Terrain maps are also commonly referred to as digital terrain maps, digital elevation maps or $2\frac{1}{2}$D grid maps.

3.1 Terrain Map Size

Sensor characteristics limit terrain map size. Assume a vehicle has a set of range sensors, $S = [S_1 \ldots S_n]$, each with a maximum range S_R, and a horizontal field of view S_α. The maximum size of the terrain map can be determined using:

$$X_{max} = \max(S_{R_i}), \;\; Y_{max} = \max(S_{R_i} \sin(\frac{S_{\alpha_i}}{2})) \tag{1}$$

assuming all sensors scan to the horizon. Figure 3 shows a terrain map, with a grid size G_l, yields a map with dimensions of

$$N_x = \frac{X_{max}}{G_l}, \;\; N_y = \frac{Y_{max}}{G_l} \tag{2}$$

where N_x and N_y are the number of grid elements along the X and Y axes respectively.

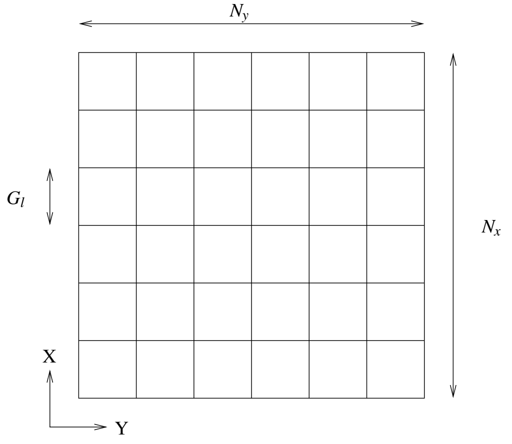


Figure 3: *Terrain Map sized to Range Sensing Capabilities*

3.2 Wrappable Map Representation

The *terrain map* is *ego-centric*, fixing the map in the vehicle local coordinate system [13, 14]. Using the fixed dimensions of $X_{max} \times Y_{max}$, the map translates and rotates the observed world using actual vehicle motion. For this map the X coordinate is always parallel to the vehicle's direction of movement. Thus, in the X direction of motion, the map recycles departing elements on map boundaries into new elements on entering across the opposite boundary, wrapping the map as the vehicle moves. Wrappable maps are highly efficient structures where a simple index update allows the map to track the vehicle's forward movement.

3.2.1 Translation

Figure 4 shows the simplest wrappable map mode, where the vehicle experiences pure translation. The variable m_i, shown in Figure 4, tracks the current index into the wrappable map.

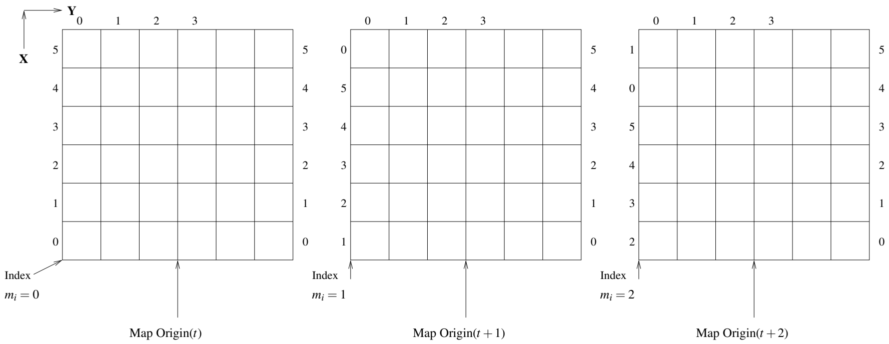


Figure 4: *Wrappable Map Representation*

This figure illustrates the map wrapping and grid element recycling process for a vehicle driving along a straight vector. As time progresses from $t = 0 \rightarrow 2$, the map index m_i moves from $m_i = 0 \rightarrow 2$. The map index is dependent upon the vehicle's motion. Once the incremental forward motion of the vehicle has exceeded the grid element size, G_l, map index, m_i, is incremented, thus causing the map to wrap. At this point all data for map elements indexed by $m_i - 1$ or greater[2] are denoted as invalid since this index represents terrain that has already been traversed.

3.2.2 Rotation

A holonomic vehicle spinning on its axis subjects the terrain map to a purely rotational vector. Figure 5 shows the rotation vector, $\theta(t)$, applied to the terrain map.

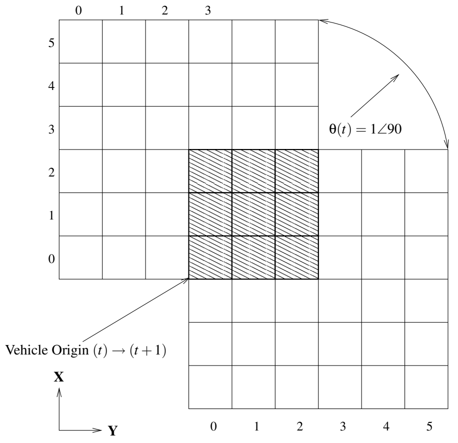


Figure 5: *Vehicle Rotating 90 Degrees About Axis*

[2] Multiple grid elements may have been traversed during the pose update period.

For rotational motion the map is mathematically transformed by the magnitude of the vehicle's yaw angle and the map data is shuffled into its corresponding grid location. In Figure 5 a 90° rotation is applied to the terrain map. The hatched region denotes the map area at time $t+1$ that overlaps with the initial map at time t.

3.2.3 Translation and Rotation

Vehicles operating in the real world have a motion vector, M that includes both a translation and a rotation component. Figure 6 show a terrain map subjected to the motion defined by vector $M_t = x_t \angle \theta_t$, over the time period $t = 0 \rightarrow 1$.

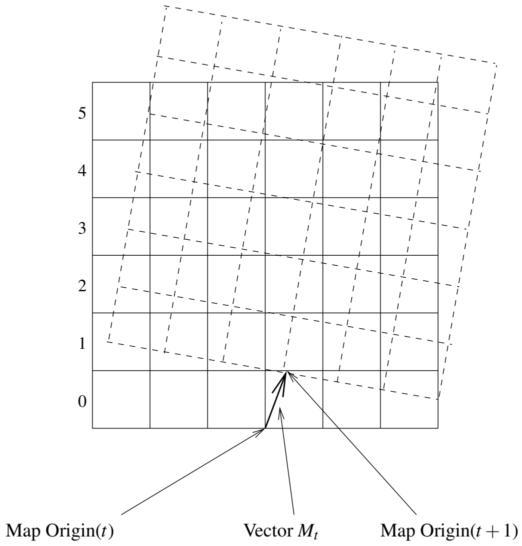


Figure 6: *A terrain map with translation and rotation*

The motion vector, M_t, can be decomposed into its translation and rotation components, thus allowing the map update to be implemented as a two stage process.

3.3 Indexing the Map

To ensure persistence while suppressing transients, the terrain map fuses new range data along with previous data into the map's grid elements. The range data, represented by a the scan point (x) in the laser beam co-ordinate system [8], is first transformed to a 3-D data point, (x, y, z), in the map co-ordinate system. The 3-D data point's (x, y) can then be transformed into the map grid element index, (i, j), using:

$$(3) \qquad i = \text{int}\left(\frac{\text{mod}(\frac{x}{X_{max}})}{G_l}\right)$$

$$(4) \qquad j = \text{int}\left(\frac{\text{mod}(\frac{y}{Y_{max}})}{G_l}\right)$$

where G_l defines the dimensions of a grid element, and X_{max}, Y_{max} define the map size. The *mod* operator returns the floating point remainder of the function and the *int* operator returns the integer portion. Figure 7 shows this transformation from the scan point's (x,y) to the map index (i,j).

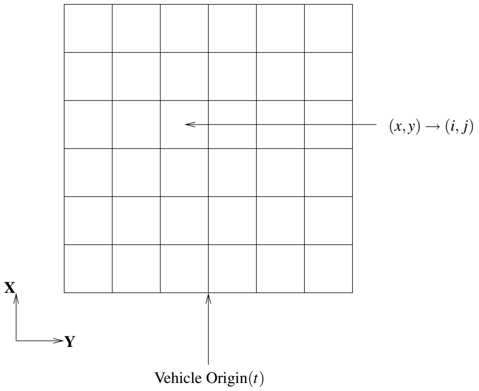


Figure 7: *X,Y Co-ordinate to Grid Map Index*

3.3.1 Indexing the Wrappable Map

Section 3.3 presented the equations to transform a scan point to index pair, $(x,y) \rightarrow (i,j)$, for the static map situation. This Section expands these equations to support the indexing of a wrappable map, as given below.

$$(5) \qquad \hat{i} = \text{int}\left(\frac{\text{mod}(\frac{x}{x_{max}})}{G_l}\right) + m_i$$

$$(6) \qquad i = \begin{cases} \hat{i} & \text{if } \hat{i} < N_x \\ \hat{i} - N_x & \text{if } \hat{i} \geq N_x \end{cases}$$

$$(7) \qquad j = \text{int}\left(\frac{\text{mod}(\frac{y}{Y_{max}})}{G_l}\right)$$

$$(8) \qquad m_i = \begin{cases} m_{i-1} + 1 & \text{if } x(t) - x(t-1) > G_l \\ m_{i-1} & \text{otherwise} \end{cases}$$

where $m_i =$ is the grid map index in the **X** direction. The transformation $y \rightarrow j$ remains unchanged, but the transformation $x \rightarrow i$ is modified to include the grid map index and the wrapping situation. Figure 8 shows the scan point to index transformation for a terrain with a map index $m_i = 1$.

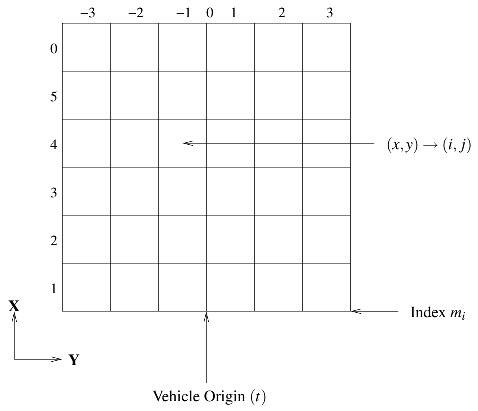


Figure 8: *X,Y Co-ordinate to Wrappable Grid Map Index*

3.3.2 Map Rotation

As shown in Figure 5 yaw, given by θ, is about the z-axis. In order to reduce the computational burden associated with applying a transformation for each individual yaw update, a map rotatation occurs at a fixed angle. When the total yaw, $\sum_{t=1}^{n} \theta_t$, exceeds the maximum yaw angle, θ_d, a map yaw transformation occurs. Figure 9 shows this rotational transformation.

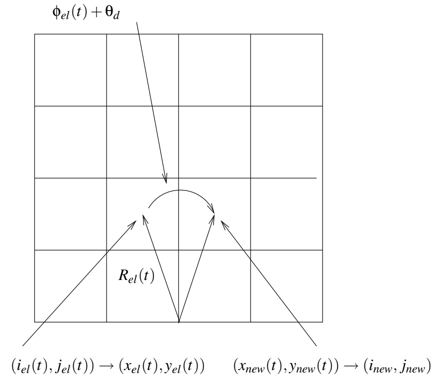

$\phi_{el}(t) + \theta_d$
$R_{el}(t)$
$(i_{el}(t), j_{el}(t)) \rightarrow (x_{el}(t), y_{el}(t))$
$(x_{new}(t), y_{new}(t)) \rightarrow (i_{new}, j_{new})$

Figure 9: *Coordinate Transformation due to Rotation*

The first step of the transformation uses the map indices (i, j) to determine the grid element's position coordinate.

$$x_{el}(t) = i_{el}(t) \cdot G_l + \frac{G_l}{2} + x_{re}(t-1) \tag{9}$$

$$y_{el}(t) = j_{el}(t) \cdot G_l + \frac{G_l}{2} + y_{re}(t-1) \tag{10}$$

The remainders from the previous rotation, given by $x_{re}(t-1)$ and $y_{re}(t-1)$, negate the quantization effects resulting from binning the (x, y) location into the grid.

The next step determines the vector, $\mathbf{R_{el}}$, from the map origin to the grid element position.

$$R_{el}(t) = \sqrt{x_{el}(t)^2 + y_{el}(t)^2} \tag{11}$$

$$\phi_{el}(t) = \tan^{-1}(\frac{y_{el}(t)}{x_{el}(t)}) \tag{12}$$

Rotating the vector, $\mathbf{R_{el}}$, by the yaw angle, θ_d, yields the new, (x_{new}, y_{new}), grid element position, as well as the remainders.

$$x_{new}(t) = R_{el}(t)\cos(\phi_{el}(t) + \theta_d) \tag{13}$$

$$y_{new}(t) = R_{el}(t)\sin(\phi_{el}(t) + \theta_d) \tag{14}$$

$$x_{re}(t) = \mathrm{mod}(x_{new}(t), G_l) \tag{15}$$

$$y_{re}(t) = \mathrm{mod}(y_{new}(t), G_l) \tag{16}$$

As described in Section 3.3, the $(x_{new}(t), y_{new}(t))$ coordinate pair converts into a map index pair, (i_{new}, j_{new}), which becomes the holder for the map data previously indexed by the map pair (i_{el}, j_{el}).

3.4 Optimizing the Wrappable Map Implementation

The wrappable map's translation component is not computationally demanding, but the rotational component has the potential to consume large amounts of processing power. The rotation algorithm, described in Section 3.3.2, can be implemented as a runtime calculation or it may be implemented using a lookup table map. The runtime calculation requires index pair, (i_{el}, j_{el}), the grid element size G_l and the remainder from the previous rotation, $(x_{re}(t), y_{re}(t))$, in order to determine the grid element position, $\mathbf{R_{el}}$. Of these variables, only the remainder component, $(x_{re}(t), y_{re}(t))$, is runtime dependent. Thus a lookup table, mapping index pairs, (i_{el}, j_{el}), to grid element positions, $x_{el}(t), y_{el}$, can be precalculated as shown below.

$$x_{el}(t) = i_{el}(t) \cdot G_l + \frac{G_l}{2} \tag{17}$$

$$y_{el}(t) = j_{el}(t) \cdot G_l + \frac{G_l}{2} \tag{18}$$

This lookup table reduces the calculations required during the runtime phase. Additionally, trigonometric functions such as cosine, sine and tangent are computationally expensive. A trigonometric lookup table would also ease the computational burden.

3.5 Pose Errors

If the grid element's variance, σ_x^2 or σ_y^2 exceeds the grid element size, G_l, this grid element's data should be fused into all affected neighbouring elements. Figure 10 shows grid element (i, j), with a mean elevation, $\bar{z}_{(i,j)}$, and variance, $\sigma^2_{x(i,j)} = \sigma^2_{y(i,j)}$, do not affect neighbouring grid elements. Conversely, grid element $(i+2, j)$, with a a x variance of $\sigma^2_{x(i+2,j)}$ and a y variance of $\sigma^2_{y(i+2,j)}$ influences two of its neighbours.

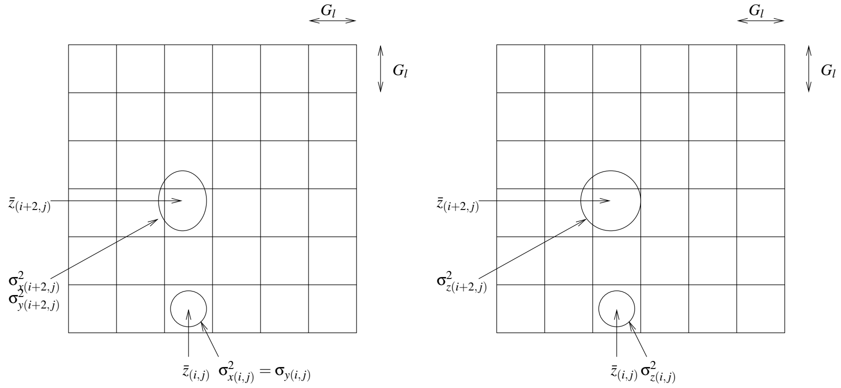


Figure 10: *Scan Point Fusion into Grid*

3.6 Grid Element Size

Traditionally a terrain map's grid element size corresponds to the minimum obstacle size that can be safely traversed by the vehicle. With a persistent terrain map it is desirable to scan the terrain as often as possible, since more sample data yields a better statistical representation of the world [8].

Although it is desirable to sample as often as possible, perception sensors, such as stereo vision devices and laser range finders, have limited sampling rates. The update rate for stereo vision devices is limited by the camera shuttle speed and the available processing power. For a nodding LRF the sampling frequency is dependent upon the nodding profile; for the uniform sample distance profile the sampling frequency is inversely related to the sample distance and for a constant rotational rate the sample frequency is fixed. [8]. The preferred uniform sample distance profile can match the sample distance with the map's grid element size. This size matching is intuitive since a sample distance smaller than the grid size results in empty grid elements, and a larger sample distance results in a loss in fidelity. Selecting a sample distance smaller, than the required minimum, is undesirable since it reduces the sampling frequency. Thus, a persistent terrain map should maximize the sampling frequency while selecting a grid element size that is appropriate for the vehicle's capabilities.

4. Terrain Map Creation using 3-D Data

Any 3-D data sources may be used by a terrain map, including stereo vision data, and the 3-D data returned from a LRF. For a vehicle operating under continuous motion conditions[3], a persistant terrain map must fuse range data where each terrain scan is acquired at a different vehicle pose. A vehicle travelling at 3 m/s will require 10 seconds to traverse the 30 m of terrain represented by a map. Thus the terrain map must fuse new 3-D data, acquired over a significant time period, while minimizing the effects of pose errors.

4.1 Confidence in the Range Data

The confidence level associated with range, expressed as a statistical variance, is a function of both the range sensor and the vehicle pose accuracy. The vehicle pose has two components, the position and the orientation. The ALS project uses a NovaTel DL-4-RT2 DGPS receiver with a position accuracy of ±3 cm. The Microstrain 3DM-GX1 IMU provides the vehicle's orientation to an accuracy of ±2 degrees. The relationship defining the range data's variance must be derived for each individual range sensor type.

4.1.1 Nodding SICK Laser

The nodding SICK laser is the primary perception sensor for the ALS project. As derived in ”The Characterization of an Inexpensive Nodding Laser” [8], orientation[4] is the most significant error source affecting the laser's accuracy. The SICK laser's range reading defines a point in space that is encircled by a spherical error region. This error is proportional to both the range and the estimated orientation error. Equation 19 shows the range/orientation relationship for the SICK laser.

$$\Delta s \simeq \sqrt{2(r\Delta\varepsilon)^2} \tag{19}$$

where r is the range and $\Delta\varepsilon$ is the estimated error in orientation. The sphere's radius, Δs^2, defines the region that bounds the true position of the range reading. Thus Δs can be interrupted as the variance associated with the range reading.

$$\sigma_r^2 = \Delta s^2 \tag{20}$$

[3] I.E. The UxV platform does not allow for a stop and scan configuration

[4] Given by roll, pitch and yaw

4.2 Fusion of Data

A grid based terrain map implementation fuses range data into the appropriate grid element, as derived in Section 3.3.1, using a statistical approach. The data fusion process takes individual range 3-D position readings and calculates the first and second moment statistics for the grid element. Thus a grid element represents terrain using the average elevation, $\bar{z}$. The confidence in this elevation value is given by the variance, σ^2, associated with this terrain patch.

Figure 11 illustrates the data fusion process, showing grid element $(\bar{z}_{(i,j)}, \sigma^2_{z(i,j)})$ with a small variance is more accurate than grid element $(\bar{z}_{(i+2,j)}, \sigma^2_{z(i+2,j)})$.

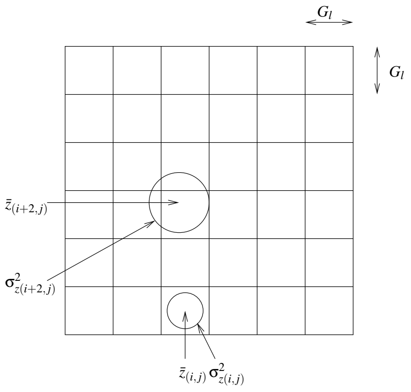


Figure 11: *Range Data Fusion into a Grid Element*

This data fusion approach is not tied to a single statistical technique, thus the researcher is free to experiment with different implementations.

4.2.1 Classical Statistics

The traditional technique to fuse new 3-D range data into a terrain map is based upon first and second order moment statistics [4]. The classical mean and variance is given by:

$$\bar{\tau} = \frac{\sum_{i=1}^{N} \tau_i}{N} \tag{21}$$

$$\sigma^2 = \frac{\sum_{i=1}^{N} (\tau_i - \bar{\tau})^2}{N-1} \tag{22}$$

Although classical statistics have been successfully implemented on terrain maps, they do not use geometry and error source knowledge and thus are a sub-optimal implementation.

4.2.2 Variance Weighted Statistics

Classical statistics do not build upon error sources, mapped through the physical geometry of the system, when fusing data into the terrain map. As described in Section 4.1.1, the orientation error when mapped through the geometry of a nodding laser, yields an estimate of the range point's variance. Thus a technique is required that optimally combines data with differing variances. The ALS project implemented a variance weighted statistical technique to optimally[5] fuse new 3-D range data into the map. The variance weighted mean is given below:

(23)
$$\bar{\tau} = \frac{\sum_{i=1}^{N} \frac{\tau_i}{\sigma_i^2}}{\sum_{i=1}^{N} \frac{1}{\sigma_i^2}}$$

It can easily be calculated on-line without the need to store all data points using:

(24)
$$\overline{\tau_{i+1}} = \frac{\sum_{i=1}^{N+1} \frac{\tau_i}{\sigma_i^2}}{\sum_{i=1}^{N+1} \frac{1}{\sigma_i^2}} = \frac{\frac{\tau_1}{\sigma_1^2} + \frac{\tau_2}{\sigma_2^2} + \frac{\tau_3}{\sigma_3^2} + \cdots + \frac{\tau_N}{\sigma_N^2} + \frac{\tau_{N+1}}{\sigma_{N+1}^2}}{\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2} + \frac{1}{\sigma_3^2} \cdots \frac{1}{\sigma_N^2} + \frac{1}{\sigma_{N+1}^2}} = \frac{\sum_{i=1}^{N} \frac{\tau_i}{\sigma_i^2} + \frac{\tau_{N+1}}{\sigma_{N+1}^2}}{\sum_{i=1}^{N} \frac{1}{\sigma_i^2} + \frac{1}{\sigma_{N+1}^2}}$$

Thus the new variance weighted mean can be simply calculated from the previous value and the new data. The variance weighted mean requires several sums be maintained:

1. The number of sample points: N
2. Sum of squared sample over variance squared: $\sum_{i=1}^{N} \frac{\tau_i}{\sigma_i^2}$
3. The sum of the variances: $\sum_{i=1}^{N} \frac{1}{\sigma_i^2}$

The definition of variance weighted variance is:

(25)
$$\sigma^2 = \frac{\sum_{i=1}^{N} \frac{(\tau_i - \bar{\tau})^2}{\sigma_i^2}}{\sum_{i=1}^{N} \frac{1}{\sigma_i^2}}$$

Equation 25 is not a computationally efficient form. Expanding this equation results in:

[5] Variance weighted means are the underlying premise of Kalman Filtering

$$(26) \qquad \sigma^2 = \frac{\sum_{i=1}^{N} \frac{\tau_i^2 - 2\tau_i\bar{\tau} + \bar{\tau}^2}{\sigma_i^2}}{\sum_{i=1}^{N} \frac{1}{\sigma_i^2}} = \frac{\sum_{i=1}^{N} \frac{\tau_i^2}{\sigma_i^2} - \sum_{i=1}^{N} \frac{2\tau_i\bar{\tau}}{\sigma_i^2} + \sum_{i=1}^{N} \frac{\bar{\tau}^2}{\sigma_i^2}}{\sum_{i=1}^{N} \frac{1}{\sigma_i^2}}$$

The middle term can be simplified as follows:

$$(27) \qquad \frac{\sum_{i=1}^{N} \frac{2\tau_i\bar{\tau}}{\sigma_i^2}}{\sum_{i=1}^{N} \frac{1}{\sigma_i^2}} = \frac{2\bar{\tau}\sum_{i=1}^{N} \frac{\tau_i}{\sigma_i^2}}{\sum_{i=1}^{N} \frac{1}{\sigma_i^2}} = 2\bar{\tau}\left(\frac{\sum_{i=1}^{N} \frac{\tau_i}{\sigma_i^2}}{\sum_{i=1}^{N} \frac{1}{\sigma_i^2}}\right)$$

Remembering the definition of the variance weighted mean, the above simplifies to:

$$(28) \qquad \frac{\sum_{i=1}^{N} \frac{2\tau_i\bar{\tau}}{\sigma_i^2}}{\sum_{i=1}^{N} \frac{1}{\sigma_i^2}} = 2\bar{\tau}\bar{\tau} = 2\bar{\tau}^2$$

Substituting back into this expanded equation and noting that $\bar{\tau}^2$ is a constant for any given index results in:

$$(29) \quad \sigma^2 = \frac{\sum_{i=1}^{N} \frac{\tau_i^2}{\sigma_i^2}}{\sum_{i=1}^{N} \frac{1}{\sigma_i^2}} - 2\bar{\tau}^2 + \frac{\sum \frac{\bar{\tau}^2}{\sigma_i^2}}{\sum_{i=1}^{N} \frac{1}{\sigma_i^2}} = \frac{\sum_{i=1}^{N} \frac{\tau_i^2}{\sigma_i^2}}{\sum_{i=1}^{N} \frac{1}{\sigma_i^2}} - 2\bar{\tau}^2 + \frac{\bar{\tau}^2\sum_{i=1}^{N} \frac{1}{\sigma_i^2}}{\sum_{i=1}^{N} \frac{1}{\sigma_i^2}}$$

Rationalizing for a common denominator gives:

$$(30) \; \sigma^2 = \frac{\sum_{i=1}^{N} \frac{\tau_i^2}{\sigma_i^2} - 2\bar{\tau}^2\sum_{i=1}^{N} \frac{1}{\sigma_i^2} + \bar{\tau}^2\sum_{i=1}^{N} \frac{1}{\sigma_i^2}}{\sum_{i=1}^{N} \frac{1}{\sigma_i^2}} = \frac{\sum_{i=1}^{N} \frac{\tau_i^2}{\sigma_i^2} - \bar{\tau}^2\sum_{i=1}^{N} \frac{1}{\sigma_i^2}}{\sum_{i=1}^{N} \frac{1}{\sigma_i^2}}$$

Once again using the definition of the variance weighted mean and factoring the common elements yields:

$$(31) \quad \sigma^2 = \frac{\sum_{i=1}^{N} \frac{\tau_i^2}{\sigma_i^2} - \left(\frac{\sum_{i=1}^{N} \frac{\tau_i}{\sigma_i^2}}{\sum_{i=1}^{N} \frac{1}{\sigma_i^2}}\right)^2 \sum_{i=1}^{N} \frac{1}{\sigma_i^2}}{\sum_{i=1}^{N} \frac{1}{\sigma_i^2}} = \frac{\sum_{i=1}^{N} \frac{\tau_i^2}{\sigma_i^2} - \frac{\left(\sum_{i=1}^{N} \frac{\tau_i}{\sigma_i^2}\right)^2}{\sum_{i=1}^{N} \frac{1}{\sigma_i^2}}}{\sum_{i=1}^{N} \frac{1}{\sigma_i^2}}$$

Rationalizing to a common denominator yields the equation in its final computing form:

(32)
$$\sigma^2 = \frac{\sum_{i=1}^{N} \frac{1}{\sigma_i^2} \sum_{i=1}^{N} \frac{\tau_i^2}{\sigma_i^2} - \left(\sum_{i=1}^{N} \frac{\tau_i}{\sigma_i^2}\right)^2}{\left(\sum_{i=1}^{N} \frac{1}{\sigma_i^2}\right)^2}$$

For the computing form of the variance weighted variance there are only three sums that must be maintained:

1. Sum of squared variances: $\sum_{i=1}^{N} \frac{1}{\sigma_i^2} = \sum_{i=1}^{N-1} \frac{1}{\sigma_i^2} + \frac{1}{\sigma_N^2}$
2. Sum of squared sample over variance squared:$\sum_{i=1}^{N} \frac{\tau_i^2}{\sigma_i^2} = \sum_{i=1}^{N-1} \frac{\tau_i^2}{\sigma_i^2} + \frac{\tau_N^2}{\sigma_N^2}$
3. Sum of sample over variance squared:$\sum_{i=1}^{N} \frac{\tau_i}{\sigma_i^2} = \sum_{i=1}^{N-1} \frac{\tau_i}{\sigma_i^2} + \frac{\tau_N}{\sigma_N^2}$

These sums can, calculated in a manner analogous to variance weighted mean, define the complete computing form for variance weighted statistics.

5. Terrain Map Requirements

The previous sections of this report detailed the theoretical aspects of DRDC's terrain map implementation. This implementation efficiently collates large data sets using a variance weighted statistical approach. This new approach retains fidelity, yet is applicable to real-time operations.

The terrain map uses statistics to summarize the terrain properties associated with each grid element. The statistical values include:

1. The mean *x* position, $\bar{x}$ and the variance σ_x^2.
2. The mean *y* position, $\bar{y}$ and the variance σ_y^2.
3. The mean and the variance in the *z* elevation given by $\bar{z}$ and σ_z^2.

A minimal map implementation requires the *z* statistics. The *x* and *y* statistics are optional values that can be used in more advanced map implementations as given in Section 3.5

5.1 Calculation of the Mean and Variance

The *(x,y,z)* averages and variances are calculated using variance weighted statistics as detailed in Section 4.2. Table 1 shows the 10 sums and quantities required in the online calculation of the (x, y, z) variance weighted statistics.

Number	Description	Sum or Quantity
1	Sample Size	N
2	Sum of squared x variances	$\sum_{i=1}^{N} \frac{1}{\sigma_{xi}^2}$
3	Sum of squared x over variance squared	$\sum_{i=1}^{N} \frac{x_i^2}{\sigma_{xi}^2}$
4	Sum of x over variance squared	$\sum_{i=1}^{N} \frac{x_i}{\sigma_{xi}^2}$
5	Sum of squared y variances	$\sum_{i=1}^{N} \frac{1}{\sigma_{yi}^2}$
6	Sum of squared y over variance squared	$\sum_{i=1}^{N} \frac{y_i^2}{\sigma_{yi}^2}$
7	Sum of y over variance squared	$\sum_{i=1}^{N} \frac{y_i}{\sigma_{yi}^2}$
8	Sum of squared z variances	$\sum_{i=1}^{N} \frac{1}{\sigma_{zi}^2}$
9	Sum of squared z over variance squared	$\sum_{i=1}^{N} \frac{z_i^2}{\sigma_{zi}^2}$
10	Sum of z over variance squared	$\sum_{i=1}^{N} \frac{z_i}{\sigma_{zi}^2}$

Table 1: *Sums and quantities of a Map Element*

5.2 Memory Requirements

The memory required to implement a grid based terrain map is dependent upon:

1. The grid element size.
2. The map size.
3. The sums and quantities associated with each element of the grid.

The number of grid elements in a terrain map is derived in Section 3.1 and is given by $N_x \times N_y$. The number of sums and quantities for each grid element is $N_{sums} = 10$, as is shown in Table 1. Thus the memory required is given by:

$$M_r = N_y \times N_y \times N_{sums} \tag{33}$$

Assuming the nodding SICK laser, with a range of 30m, as the primary perception sensor, and a grid element size $G_l = 0.2$, Section 3.1 yields $N_x = N_y = \frac{30.0}{0.2} = 150$.

Thus the memory requirements for the terrain map can be calculated as:

$$M_r = N_x \times N_y \times N_{sums} = 150 \times 150 \times 10 = 225000 \tag{34}$$

If each sum or quantity is assumed to be an 8 byte floating point number then a approximately 2,000,000 bytes of memory is required. This value is readily achievable for current computing platforms.

6. Implementing a Terrain Map Under MIRO

The ALS project implemented a variance weighted terrain map under the Miro framework [15, 9, 16, 17, 18]. This map, representing the terrain in front of the Raptor UGV, feeds the traversibility analysis and thus is instrumental in enabling obstacle and hazard avoidance behaviours. The Raptor UGV is shown in Figure 12.

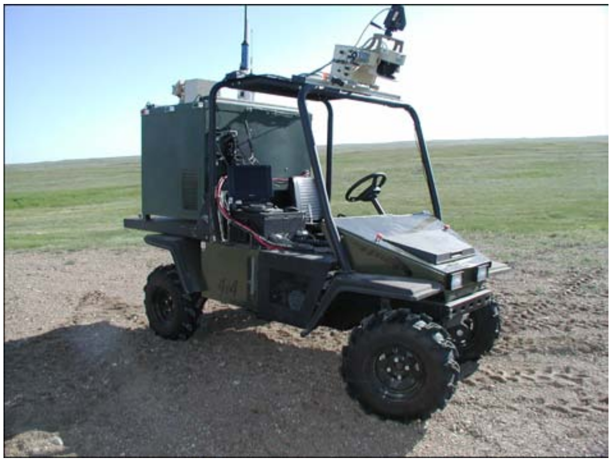


Figure 12: *The Raptor UGV*

6.1 Configuration

The Raptor UGV carries four perception sensors that feed data to the terrain map:

- Nodding SICK Laser.
- Digiclops Stereo vision cameras.
- Sokkia GSR2600 DGPS.
- Microstrain 3DM-G IMU.

The nodding SICK laser, operating at a 26ms update rate, produces range data; the Digiclops camera updates its disparity map every 500ms. The Sokkia GPS provides a new position every 250ms and the Microstrain IMU produces an orientation at 62.5ms intervals.

The *ModelServer* [19] component fuses the GPS position with the IMU orientation to publish a pose event once every 100ms.

DRDC's computationally efficient terrain map implementation fuses data and performs map wrapping at the perception sensor data rates. But, a map event is only published on 500ms intervals since the computationally intensive traversibility analysis can not handle higher data rates.

6.2 The Terrain Map Component

The MIRO framework [17] prompts component based implementations, where a component represents a unique capability or behaviour. Components share data via network transparent CORBA objects [20, 21]. The *Terrain Map* component received and publishes CORBA events, therefore the Publish-Subscribe Server design pattern served as the template for this implementation [16]. The collaboration diagram, shown in Figure 13, illustrates the *Terrain Map* component.

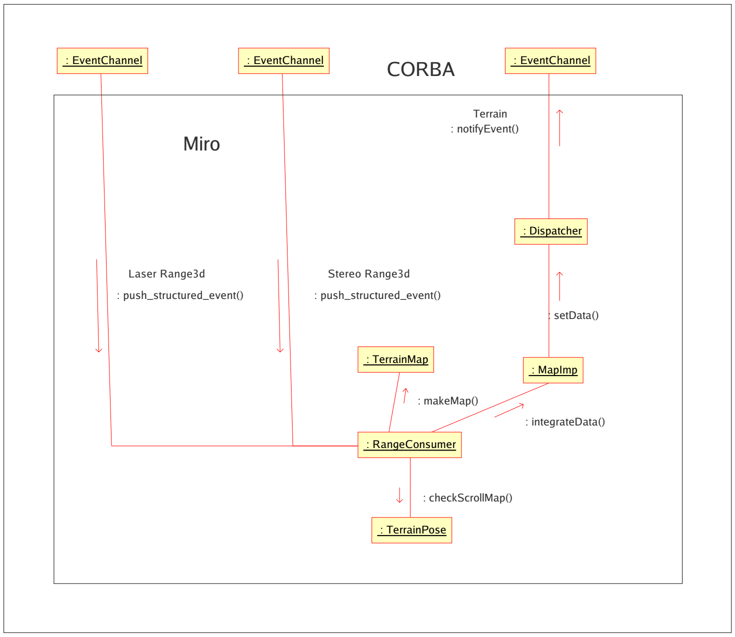


Figure 13: *Map Publish-Subscribe Server Design*

The *Terrain Map's* classes are described in the following sections.

6.2.1 RangeConsumer

Upon initialization the *RangeConsumer* class, using the CORBA Naming Service, resolves and subscribes to *Laser* and *Stereo* CORBA *Range3d* events, as well as *Pose* CORBA events.

Under operational conditions the *RangeConsumer* consumes *Range3d* CORBA events and invokes the *makeMap* method, which integrates the new 3-D range data into the terrain map. Once the 500ms update interval has expired the *RangeConsumer* calls the *integrateData* method, the first step in publishing a *Map* CORBA event.

When the *RangeConsumer* receives a *Pose* CORBA event it calls the *checkScrollMap* method.

6.2.2 TerrainMap

The *TerrainMap* class fuses new 3-D data into the terrain map using variance weighted statistics. Before data fusion occurs, the range is first transformed from its local coordinate system to the map coordinate system. This transformation is accomplished with assistance from the *ModelServer* and its transformation methods [19] and results in a 3-D data point. With the range data transformed into the appropriate coordinate system, the mapping algorithm fuses then new data into the current terrain map.

6.2.3 TerrainPose

As described in Section 3.2 the vehicle's motion vector has two components, translation along the X axis and rotation about the Z axis. The *TerrainPose* class implements two methods, corresponding to the translation and rotation components. Both of these methods are invoked from the *checkScrollMap*, as shown in Figure 13.

The translation method, *scrollMapX*, wraps the map when the vehicle's incremental forward motion exceeds the grid size, G_l. The wrapping process increments the grid map index, m_i, and initializes the traversed grid elements.

The rotation method, *scrollMapY*[6] implements a map rotation once the summed vehicle yaw exceeds the deviation angle, θ_d. To simplify the implementation and improve performance the rotation method uses a *working* map. The rotation algorithm blindly applies the rotation geometry to each grid element, determines the map indices resulting from the rotation, and sums the grid statistics into the *working* map. The *working* map then becomes the new *current* map and the previous *current* map becomes the new *working* map.

6.2.4 MapImp

The *integrateData* method, in the *MapImp* class, copies the map data into a locally allocated map. Using this local map the *Terrain Map's* server implementation responds to polling requests. Before returning, *integrateData* passes the *Terrain Map* CORBA object to the *setData* method.

6.2.5 Dispatcher

The *Dispatcher* takes the *Terrain Map* CORBA object and publishes the CORBA object on the *EventChannel*.

[6] This method's name is misleading since it implements rotation

6.3 CORBA Objects

The *Terrain Map* component supports two CORBA object types:

- *MapSeqEventIDL*: A generic map object based upon C++ sequences.
- *MapArrayEventIDL*: A fixed length, 2 dimensional array optimized for performance.

The IDL code, implementing the generic map object, is shown below.

```
//! A vector of floats creates a Map Row.
typedef sequence<double> MapRowIDL;
//! A vector of Map Rows creates a 2D Map Grid.
typedef sequence<MapRowIDL> MapGridIDL;
//! A vector of Map grids creates the full 3D map.
typedef sequence<MapGridIDL> MapSeqIDL;

struct MapSeqEventIDL
{
  //! The time the map was acquired.
  TimeIDL time;
  //! The Pose
  PoseTransformIDL pose;
  //! Define the Map type
  long maptype;
  //! Depth of the map as an index = depth/gridsize.
  long index_depth;
  //! Width of the as an index = 2*width/gridsize
  long index_width;
  //! A pointer to the current starting X index in the map
  long x_curr;
  //! The map.
  MapSeqIDL map;
 };
```

The IDL code, creating the fixed size CORBA map object, is shown below.

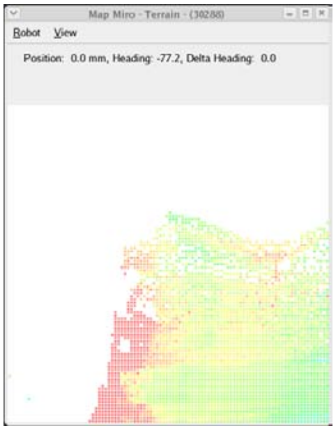
```
 //! For the purposes of speed create a fixed length array map
typedef double MapArrayIDL[ZVAR+1][INDEX_DEPTH][INDEX_WIDTH];

struct MapArrayEventIDL
{
  //! The time the scan was acquired.
  TimeIDL time;
  //! The Pose
  PoseTransformIDL pose;
  //! Define the Map type
  long maptype;
  //! Depth of the map as an index = depth/gridsize.
  long index_depth;
  //! Width of the as an index = 2*width/gridsize
  long index_width;
  //! A pointer to the current starting X index in the map
  long x_curr;
  //! The map .
  MapArrayIDL map;
};
```

The sequenced based CORBA object sacrifices speed for improved data abstraction; the fixed size array based CORBA object optimizes performance.

6.4 Debugging and Visualization Tools

The *QtMap* interface, shown in Figure 14, presents a plan view of the terrain map. Regions in red are elevated, and yellow/green regions represent flatter terrain.



Position: 0.0 mm, Heading: -77.2, Delta Heading: 0.0

Figure 14: *QtMap Interface*

The *Sensor3dGet* and *Sensor3dStream* debugging components print the mean elevation, $\bar{z}$, and the variance, σ_2, for each grid in the terrain map.

6.5 Maintaining an Egocentric Map

The terrain map is *egocentric*, as described in Section 4, and is referenced to the Raptor's front bumper. The terrain map uses the pose estimate, supplied by the

ModelServer component, to maintain the map's egocentricity. The world pose estimate is transformed to the map's coordinate system using a relative pose change approach. As shown in Equation 36, the change in pose it calculated every time a pose update is received. This change in pose change transforms the data from world coordinates, to the egocentric coordinate system required by the map.

$$\Delta x_m = \sqrt{((x_t - x_{t-1})^2 + (y_t - y_{t-1})^2)} \tag{35}$$

$$\Delta \theta_m = \theta_t - \theta_{t-1} \tag{36}$$

Where x_t and y_t are the estimated position, and θ_t is the estimated heading, both in world coordinates. Δx_m and $\Delta\theta_m$ define the incremental position change in map coordinates.

6.6 Performance

The terrain map proved to be one of the more computational demanding components in the Raptor UGV system. The resources [7] required to process laser 3-D data events and stereo 3-D data events are shown in Table 2.

CORBA Object	Update Period	CPU %	Memory %
Range3dLaserEventIDL	26.6ms	2.3	2.8
Range3dStereoEventIDL	500ms	9.5	2.5

Table 2: *Computational Resources Consumed*

7. Results

7.1 Experimental Results

The performance of a terrain map based on classical statistics was compared to a variance weighted mean terrain map. For this experiment the nodding apparatus was mounted at a height of 0.87 m above a flat floor and the nodding mechanism was commanded to create a uniform scan at a density of 10 cm per scan. Each scan point's raw range data, encoder position and time synchronization was acquired and written to file.

Matlab implemented the terrain map data fusion algorithm using both classical and variance weighted statistics. Using each technique the scanned flat floor was represented using a terrain map representation. Figure 15 shows the calculated floor height versus distance from the laser, for a slice of the terrain map at a Y co-ordinate of 0 ±5cm.

[7] For a Pentium 2.8 GHz processor with 1 GB of RAM

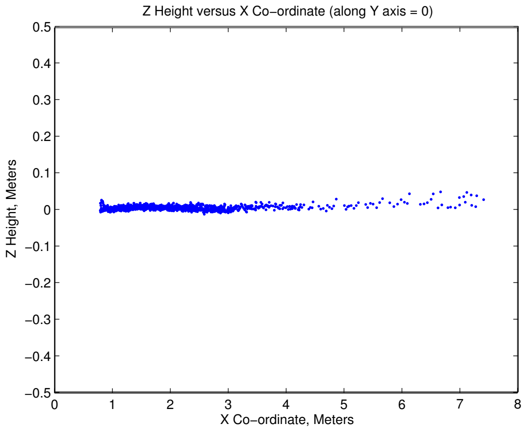


Figure 15: *Floor Height vs Distance from Robot*

As can be seen in this figure the $\bar{z}$ elevation's accuracy varies with the range from the laser. This experiment was performed with and without IMU data. Table 3 compares the two statistical technique's performance. For each technique the mean elevation, $\bar{z}$, and the variance, σ^2, is shown.

Data Fusion Technique	$\bar{z}$ (m)	σ^2 (mm)
Classical Statistics	0.0060	0.035
Variance Wt Statistics	0.0055	0.029
Classical Statistics with IMU	0.0015	0.038
Variance Wt Statistics with IMU	0.0015	0.031

Table 3: *Results scanning a flat floor*

This table shows both techniques produce excellent results, with the variance weighting producing a slightly more accurate flat floor representation. The inclusion of the IMU orientation data into the calculations did not significantly affect the results, which is not suprising since the Microstrain 3DM-G IMU, under static conditions, exhibited a standard deviation of 0.047 degrees from 0 degrees.

7.2 Simulations

Although IMUs are accurate under static conditions, their performance under dynamic conditions are typically worse by orders of magnitude. A vehicle traversing unstructured, outdoor terrain will place the IMU in a dynamic environment and thus it is important to characterize the performance of terrain map creation techniques under these conditions. The Microstrain 3DM-G specifies an accuracy of ±5 degrees while operating under dynamic conditions and the newer Microstrain 3DM-GX1 specifies an accuracy of ±2 degrees. A simulation, performed using the real flat floor data, added random noise to the IMU's orientation data to simulate the accuracy of the Microstrain IMUs under dynamic conditions. The performance of each data fusion technique is shown in Table 4.

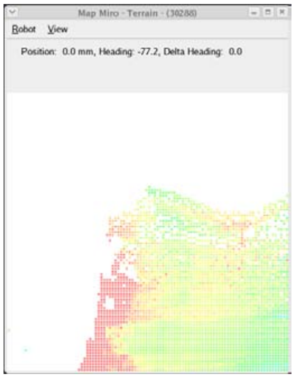
Data Fusion Technique	IMU	$\bar{z}$ (m)	σ^2 (mm)
Classical	3DM-G	-0.211	25.7
Variance Wt	3DM-G	-0.182	18.7
Classical	3DM-GX1	-0.077	3.9
Variance Wt	3DM-GX1	-0.068	2.9

Table 4: *Results a flat floor with simulated Microstrain Accuracies*

The results in Table 4 reveal the variance weighted technique is significantly less sensitive to IMU errors than is the classical statistics technique.

7.3 Demonstation Results

The ALS demonstrations, conducted in September 2005, extensively exercised the terrain mapping component. A typical terrain map generated using the nodding SICK laser is shown in Figure 16. The map extends from the Raptor's front bumper to approximately 20m in front of the vehicle and 10 m on either side of the vehicle. The red region on the map's right hand side represents a building wall, and the lighter coloured yellow and green region represent flat terrain. Since the terrain behind the wall can not be scanned, the region behind the wall is coloured white marking it as an unknown area.



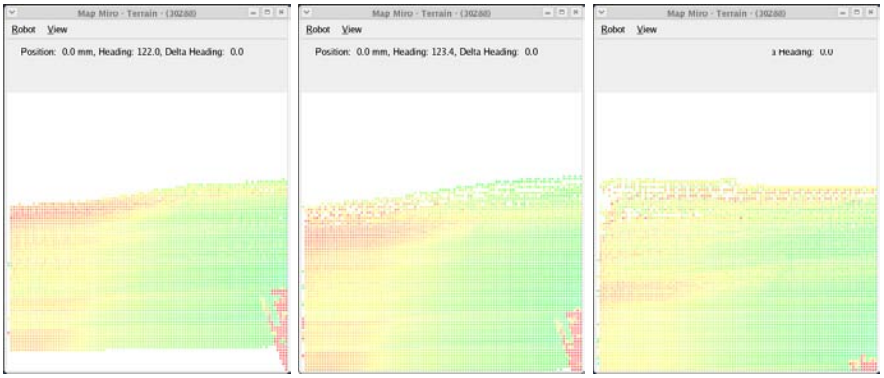
Position: 0.0 mm, Heading: -77.2, Delta Heading: 0.0

Figure 16: *Typical Terrain Map*

Figure 17 illustrates the terrain map's wrapping capabilities.

The initial map, built while the Raptor was stationary, has a white region across the bottom of the map, which denotes the *dead-zone*. An obstacle[8], denoted in red, is

[8] This obstacle is relatively tall since the nodding laser scans it in its peripheral vision.


Map Miro - Terrain - (30288)
Robot
View
Map Miro - Terrain - (30288)
Robot
View
Map Miro - Terrain - (30288)
Robot
View

Position: 0.0 mm, Heading: 122.0, Delta Heading: 0.0

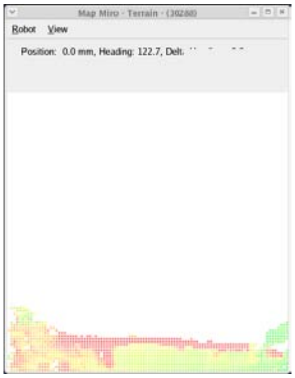
Position: 0.0 mm, Heading: 123.4, Delta Heading: 0.0

a Heading: 0.0

Figure 17: *Wrapping Terrain Map*

located in the botton right corner of the map. The map tracks the Raptor's position as it moves and the scanned terrain migrates towards the bottom of the map. This tracking is evident in Figure 17 where the *dead-zone* disappears and the obstacle moves downwards, out of the mapping region.

Figure 18 shows another illustrative terrain map. In this Figure the Raptor vehicle is positioned directly in front of a wall. The wall, denoted in red, almost completely obscures the map's field of view. Thus, the majority of the terrain map is tagged as unknown and displayed in white.

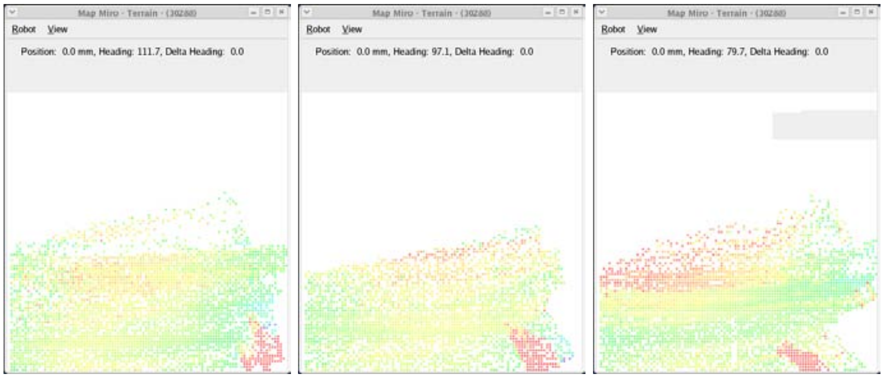


Position: 0.0 mm, Heading: 122.7, Delt.

Figure 18: *Obstructed Terrain Map*

Figure 19 shows the Raptor vehicle performing a left turn. The obstacle, in the bottom right of the figure, shuffles across the map as the Raptor makes the turn. The right side of the map reverts to unknown status when it no longer has valid data. But as the nodding SICK laser scans the unknown region, the map is fused with new 3-D data, and the terrain once again has a valid representation.

This figure also reveals that the map's density becomes sparser as a result of the turn. As the vehicle rotates it quickly faces unknown terrain and invalidates previously



Position: 0.0 mm, Heading: 111.7, Delta Heading: 0.0

Position: 0.0 mm, Heading: 97.1, Delta Heading: 0.0

Position: 0.0 mm, Heading: 79.7, Delta Heading: 0.0

Figure 19: *Rotating Terrain Map*

scanned regions. The nodding SICK laser, even in the tandem configuration where there are two forward looking lasers, does not have the data rates that allow the map to maintain its density.

The terrain maps, generated for the ALS demonstration, relied exclusively on nodding SICK laser 3-D data. Although the terrain map implementation handled 3-D data from the Digiclops camera, this data was not used. Early field trials revealed the Digiclop's 3-D data was especially prone to fringe effects where the extremities of its field of view yielded erroneous and unreliable data.

As described earlier in this report, terrain maps enable autonomous manoeuvres. Terrain maps are analyzed for traversibility creating a Traversibility Map, as shown Figure 20.

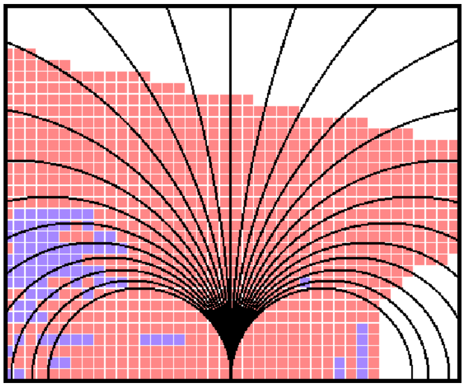


Figure 20: *A Traversibility Map overlaid with 25 Candidate Arcs, Obstacles: Blue, Traversable: Ted*

The traversibility map labels terrain as traversable, obstacles or unknown. The path planner tries numerous candidate arcs and marks an arc as invalid if it encounters an obstacle, as can be seen in Figure 20.

8. Conclusions

A prerequisite of autonomous navigation is an adequate world representation. Using a world representation, traversibility analysis, and obstacle avoidance behaviours, a UxV can safely navigate within its environment. This report researched the current state-of-the-art for world representation techniques before concluding outdoor, unstructured environments are best represented by the terrain map technique.

Many terrain map implementations rely upon *single sample* perception devices; thus, the map's data fusion is relatively insensitive to pose errors. 3-D ranging techniques such as stereo vision, or ”stop and scan” techniques are also insensitive to pose errors. DRDC's continuous motion mapping, using an inexpensive nodding SICK laser, generates multiple terrain samples; thus DRDC's terrain map implementation is pose sensitive. To minimize pose error effects DRDC proposes the variance weighted statistical technique to fuse new 3-D data into the terrain map. A detailed analysis of pose error sources [8] revealed roll, pitch and yaw as the dominant error source. Using the 3-D data's estimated positional error, the variance weighted technique optimally fuses new data into the terrain map.

Experiments, conducted under static conditions, revealed the variance weighted technique performed slightly better than a classical statistics implementation. Simulations, representing the dynamic conditions of a moving platform, disclosed the performance of the variance weighted technique as superior to classical statistics.

A wrappable, rotating, variance weighted terrain map was implemented under the MIRO framework. The MIRO framework allowed the map to simultaneously consume 3-D data from multiple data sources using the vehicle's pose to maintain the map's egocentricity. A demonstration, conducted in the fall of 2005, exercised the map's capabilities. During these trials DRDC's terrain map successfully and adequately represented the world. The map remained egocentric while the Raptor performed translational and rotational manoeuvres. The terrain maps, once processed for traversibility, supported local manoeuvres and allowed the Raptor to successfully avoid obstacles and hazards. Hard turn manoeuvres would often result in a sparse map . The roots of this problem were traced a lack of data, stemming from the nodding SICK laser's slow sample rate. To help ameliorate this condition, dual, forward looking lasers were mounted on the Raptor. The dual lasers helped increase the 3-D data densities but the map fringes still suffered from sparsity problems.

The terrain map's efficient implementation consumed relatively small amounts of computing resources. It could easily handle multiple nodding SICK laser devices, as well a 320x240 stereo vision frames twice per second. This excess overhead means this terrain map implementation can be expanded to meet future requirements.

The current terrain map implementation maintained its egocentricity relative to the Raptor's front bumper. This reference point often proved problematic since the obstacle ceased to exist once the vehicle's front bumper passed by it. This led to a behaviour where the Raptor, believing the obstacle had been avoided, would turn sharply and side swipe the obstacle. Future terrain maps will move the reference point to the rear bumper and eliminate the root cause of this problem.

References

1. Herbert, M. and Krotkov, E. (1993). Local Perception for Mobile Robot Navigation in Natural Terrain: Two Approaches. In *Workshop on Computer Vision for Space Applications*, pp. 24–31.

2. Kweon, S. and Kanade, T. (1992). High-Resolution Terrain Map from Multiple Sensor Data. *IEEE Transactions on Pattern Analysis and Machine Vision*, *14*(2), 278–292.

3. Bellutta, P., Manduchi, R., Matthies, L., Owens, K., and Rankin, A. (2000). Terrain Perception for DEMO III. In *Proceedings of the 2000 Intelligent Vehicles Conference*.

4. Goldberg, S., Maimone, M., and Matthies, L. (2002). Stereo Vision and Rover Navigation Software for Planetary Exploration. *IEEE Aerospace Conference Proceedings*.

5. Lacroix, S., Mallet, A., and Bonnafous, D. (2000). Autonomous Rover Navigation on Unknown Terrains Demonstrations in the Space Museum ”Cite de l’Espace” at Toulouse Automation, Albuquerque, USA, 1997. In *7th International Symp. on Experimental Robotics*, pp. 669–683. Honolulu, HI.

6. Kelly, A. (1997). Intelligent Unmanned Ground Vehicles: Autonomous Navigation Research at Carnegie Mellon, Ch. RANGER: Feedforward Control Approach to Autonomous Navigation, pp. 105–144. Kluwer Academic Publishers.

7. Herbert, M. (2000). Active and Passive Range Sensing for Robotics. In *IEEE International Conference on Robotics and Automation*, pp. 102–110. IEEE. San Franscisco, CA.

8. Broten, G. and Collier, J. (2005). The Characterization of an Inexpensive Nodding Laser. (DRDC Suffield TR 2005-232). Defense R&D Canada – Suffield. Medicine Hat, Alberta.

9. Broten, G., Monckton, S., Giesbrecht, J., Verret, S., Collier, J., and Digney, B. (2004). Towards Distributed Intelligence. (DRDC Suffield TR 2004-287). Defence R&D Canada – Suffield. Medicine Hat, Alberta.

10. Nilsson, Nils J. (1984). Shakey The Robot. (Technical Report 323). AI Center, SRI International. 333 Ravenswood Ave., Menlo Park, CA 94025.

11. Brooks, R.A. (1991). Artificial IntelligenceMemo No. 1293: Intelligence without Reason, Massachusetts Institute of Technology.

12. Brooks, Rodney A. (1986). A Robust Layered Control System for a Mobile Robot. *IEEE Journal of Robotics and Automation*, *RA-2*(1), 14–23.

13. Kelly, A. J. (1995). Predictive Control Approach to the High-Speed Cross-Country Autonomous Navigation Problem. Ph.D. thesis. Carnegie Mellon University. Pittsburg, Pa.

14. Kelly, A. J. (1997). An Approach to Rough Terrain Autonomous Mobility. In *International Conference on Mobile Planetary Robots*, Santa Monica.

15. Broten, G. and Monckton, S. (2005). Frameworks and Middleware for Unmanned Ground Vehicles. In *Proceedings of SPIE Symposium on Mobile Robots*.

16. Broten, G., Moncton, S., Giesbrecht, J., and Collier, J. (2006). Software Engineering for Experimental Robotics, Ch. UxV Software Systems, An Applied Research Perspective. Springer Tracts in Advanced Robotics.

17. Broten, G., Monckton, S., Giesbrecht, J., and Collier, J. (2006). Software Sysetms for Robotics, An Applied Research Perspective. *International Journal of Advanced Robotic Systems*.

18. Utz, H., Sablatnog, S., Enderle, S., and Kraetzschmar, G. (2002). Miro - MIddleware for Mobile Robot Applications. *IEEE Transactions on Robotics and Automation*.

19. Monckton S., Broten G., Vincent I. (2006). A Prototype Vehicle Geometry Server: Design and development of the ModelServer CORBA Service. (DRDC Suffield TR 2005-240). Defence R&D Canada – Suffield. Medicine Hat, Alberta.

20. Henning, M. and Vinoski, S. (1999). Advanced CORBA Programming with C++, Addison-Wesley.

21. Bolton, F. (2002). Pure CORBA: A code intensive premium reference, SAMS.

DOCUMENT CONTROL DATA

(Security classification of title, body of abstract and indexing annotation must be entered when document is classified)

1. ORIGINATOR (the name and address of the organization preparing the document. Organizations for whom the document was prepared, e.g. Centre sponsoring a contractor's report, or tasking agency, are entered in section 8.)

Defence R&D Canada – Suffield
PO Box 4000, Medicine Hat, AB, Canada T1A 8K6

2. SECURITY CLASSIFICATION (overall security classification of the document including special warning terms if applicable).

UNCLASSIFIED

3. TITLE (the complete document title as indicated on the title page. Its classification should be indicated by the appropriate abbreviation (S,C,R or U) in parentheses after the title).

World Representation Using Terrain Maps (U)

4. AUTHORS (Last name, first name, middle initial. If military, show rank, e.g. Doe, Maj. John E.)

G.S. Broten, J.L. Giesbrecht, S.P. Monckton

5. DATE OF PUBLICATION (month and year of publication of document)

December 2005

6a. NO. OF PAGES (total containing information. Include Annexes, Appendices, etc).

40

6b. NO. OF REFS (total cited in document)

21

7. DESCRIPTIVE NOTES (the category of the document, e.g. technical report, technical note or memorandum. If appropriate, enter the type of report, e.g. interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered).

Technical Report

8. SPONSORING ACTIVITY (the name of the department project office or laboratory sponsoring the research and development. Include address).

Defence R&D Canada – Suffield
PO Box 4000, Medicine Hat, AB, Canada T1A 8K6

9a. PROJECT OR GRANT NO. (if appropriate, the applicable research and development project or grant number under which the document was written. Specify whether project or grant).

9b. CONTRACT NO. (if appropriate, the applicable number under which the document was written).

10a. ORIGINATOR'S DOCUMENT NUMBER (the official document number by which the document is identified by the originating activity. This number must be unique.)

DRDC Suffield TR 2005-248

10b. OTHER DOCUMENT NOs. (Any other numbers which may be assigned this document either by the originator or by the sponsor.)

11. DOCUMENT AVAILABILITY (any limitations on further dissemination of the document, other than those imposed by security classification)

(X) Unlimited distribution
(   ) Defence departments and defence contractors; further distribution only as approved
(   ) Defence departments and Canadian defence contractors; further distribution only as approved
(   ) Government departments and agencies; further distribution only as approved
(   ) Defence departments; further distribution only as approved
(   ) Other (please specify):

12. DOCUMENT ANNOUNCEMENT (any limitation to the bibliographic announcement of this document. This will normally correspond to the Document Availability (11). However, where further distribution beyond the audience specified in (11) is possible, a wider announcement audience may be selected).

13. ABSTRACT (a brief and factual summary of the document. It may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall begin with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (S), (C), (R), or (U). It is not necessary to include here abstracts in both official languages unless the text is bilingual).

This report introduces a new terrain map technique for representing unstructured environments. Using an inexpensive nodding 2-D laser rangefinder this technique captures multiple terrain data sets and optimally fuses the data into a wrappable terrain map. Traditional terrain map implementations scan a terrain patch only once, thus integrate a single data point into the map. The nodding laser, allowing multiple scans of a given terrain patch, must fuse multiple data sets into the terrain map. A moving vehicle, with a limited pose accuracy, demands a data fusion technique that minimizes the effects of pose errors. This research investigated and tested a variance weighted statistical technique to optimally fuse terrain data sets into a terrain map. Simulations and experiments were conducted, which demonstrated the performance of the variance weighted technique as superior to classical statistical methods. DRDC implemented a variance weighted terrain map under the Miro architecture and tested its performance during the Autonomous Land Systems (ALS) demonstration. This terrain map technique performed admirably and its accurate terrain representation allowed the Raptor Unmanned Ground Vehicle to successfully navigate while avoiding obstacles and hazards.

14. KEYWORDS, DESCRIPTORS or IDENTIFIERS (technically meaningful terms or short phrases that characterize a document and could be helpful in cataloguing the document. They should be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location may also be included. If possible keywords should be selected from a published thesaurus. e.g. Thesaurus of Engineering and Scientific Terms (TEST) and that thesaurus-identified. If it not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title).

terrain maps, digital elevation maps, $2\frac{1}{2}$ D maps, world representation, data fusion