Terrain Classification and Segmentation using Non-Semantic Range Data

Michael Woods, José Guivant and Jayantha Katupitiya
School of Mechanical and Manufacturing Engineering,
The University of New South Wales, Sydney NSW
{m.woods, j.guivant, j.katupitiya}@unsw.edu.au

Abstract

Terrain classification is an important component for path planning and control of Autonomous Ground Vehicles that operate in unstructured and uneven terrain. Current terrain classification has been predominantly achieved through the use of visual and semantic spatial data. This paper presents a method, for terrain type classification and scene segmentation, as well as a confidence measure in classification, based on non-semantic range data. The features used in the inference process are calculated at the pixel level, from the high-frequency components of the terrain depth images. The extracted features are then transformed using Multiple Discriminant Analysis (MDA), with the Bhattacharyya distance used in terrain classification for the different typical terrain types; grass, artificial turf, gravel, tile and concrete. The effects of relevant parameters, such as the size of the region of interest, on the performance of the approach (i.e. classification, boundary detection and computation time) are also investigated. A classification accuracy of 85.1% was achieved using non-semantic range data with a confidence threshold of 40%.

1. Introduction

Autonomous Farming is becoming an important aspect of the mechanisation of farming, with agricultural Autonomous Ground Vehicles (AGVs) being required to operate within these harsh and remote contexts where the terrain is more unstructured and uneven, leading to a greater amount of uncertainty in system behaviour. It is necessary, therefore, to identify the frictional and geometric characteristics of the terrain in order to reduce the uncertainties in the system model so that AGV operation can be carried out safely. The terrain frictional coefficients are dependent on the type of terrain, allowing those coefficients to be determined through terrain type classification [Angelova *et al.*, 2006].

Generally, the approaches that are applied in terrain classification can be grouped into three main categories; vision only [Brooks and Iagnemma, 2007], range only [Wang *et al.*, 2012; Kragh *et al.*, 2015] and a fusion of both [Yi *et al.*, 2014; Zhao *et al.*, 2014]. The boundaries between different terrain types are of great importance to be identified, as the greatest amount of uncertainty is present in the terrain model at these boundaries [Lin *et al.*, 2008; Cho *et al.*, 2010].

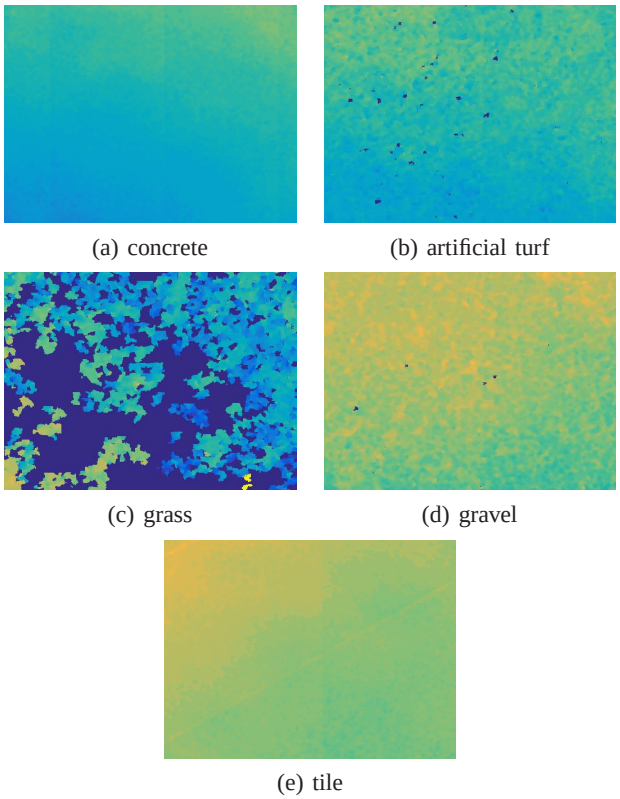
Terrain classification methods usually estimate boundaries between terrain types through the use of visual or geometric changes in the scene. These methods provide good estimation of these boundaries under most conditions, however, there are conditions where these methods provide poor classification and can result in misclassified terrain types. Terrain classification and segmentation could also be accomplished through the use of non-semantic range data, which does not attempt to classify an object based on geometry and appearance. Currently, non-semantic range data based implementations [Walas, 2013] provide only moderately good classification results. A method that is able to provide reliable terrain classification using non-semantic range data can be utilised during situations where visual or semantic spatial information is degraded due to environmental conditions.

This paper introduces an extension to the work in [Woods *et al.*, 2013], which is able to provide terrain type classification and to segment the scene based on the boundaries between multiple terrain types. A classification confidence is also provided for each region, allowing for areas in the scene which do not have a high classification confidence to be reported as unclassified, preventing misclassification and reducing the uncertainty about the terrain conditions. There are four main steps that are involved in the proposed terrain classification and segmentation algorithm: range data pre-processing, feature space dimensionality reduction, terrain classification and finally classification confidence measurement. These steps are described and explained further in the sections below with experimental classification results of the method presented and discussed.

2. Terrain Classification

2.1. Range Data Pre-processing

A key issue of the previously developed method [Woods *et al.*, 2013], was that the algorithm segmented a scene based only on geometry with no regard to the underlying terrain types, meaning that in the absence of a geometric change, boundaries between terrain types would not be discovered and classified correctly. To overcome this limitation a method of pre-processing the raw range data to provide an adequate feature map for terrain classification and segmentation is proposed within this paper. Each range data point in a depth image is assumed to be the center of a fixed size region of interest, utilising the range data within this region to generate features for each data point.



(a) concrete (b) artificial turf

(c) grass (d) gravel

(e) tile

Fig. 1. Typical depth image of the different terrain types

For validating the approach, a platform retrofitted with 3D perception capabilities was employed. The method can be utilised with many different 3D sensors, with the sensor used for this experiment being a structured light 3D camera. The range images of the different terrain types; grass, artificial turf, gravel, tile and concrete, shown in Fig. 1, were captured with the 3D camera facing downwards at a shallow angle, at a distance of 500mm from the ground, with the 3D camera, a PrimeSense Carmine 1.09, mounted on the robotic platform, as shown in Fig. 2. The range images were taken while the robot platform was stationary, and in areas that were not in direct sunlight as this is a limitation of the sensor used with the lighting conditions being held constant for all data sets.

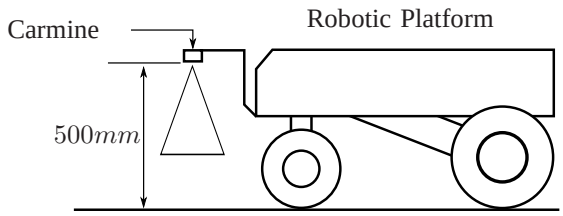


Fig. 2. Schematic of the experimental setup showing the structured light camera mounted on a moving platform

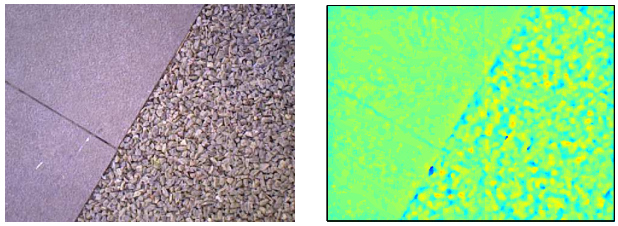
In order to generate a feature image of the scene, an important step is to generate a residual image from the range data. A high-pass spatial filter is applied to the range data as one of the stages in the range data pre-processing. The high-pass spatial filter response needs to be tuned for the expected spatial frequencies that are encountered. This is achieved by fitting the spatial filter response to the output of a Piece-Wise Multi-Linear (PWML) surface estimation [Robledo *et al.*, 2011]. A typical residual image that is generated can be seen in Fig. 3, where the filter response at the boundary between the terrain types can be seen to be a smooth transition.

Gray-Level Co-occurrence Matrix Features

The Gray-Level Co-occurrence Matrix (GLCM) method that was developed by Haralick [Haralick *et al.*, 1973], operates by recording the jump between different pixel values in directions and scales that can be specified. The gray-level co-occurrence matrix as defined in (1), which is used for all subsequent features and is the definition originally used in [Haralick *et al.*, 1973], where I is the input residual image of the terrain, with either spatial or visual information.

$$p_{\Delta x,\Delta y}(i,j) = \sum_{k=1}^{n}\sum_{l=1}^{n} \begin{cases} 1 & \text{if } I(k,l)=i \text{ and} \\ & I(k+\Delta x, l+\Delta y)=j \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

where Δx and Δy are used to determine the direction and step size that the GLCM is based on, and can also be expressed in a combination of angles θ and step distances d as $p_{\theta,d}(i,j)$. The four primary orientation combinations



(a) ground truth (b) 3D residual image

Fig. 3. Images showing the generated depth residual

of Δx and Δy with a step distance d of 1 grid gives the angles θ as 0°, 45°, 90° and 135°, with symmetric features being recorded so that other orientations, i.e. 180°, are also included. The different step directions and sizes are depicted in Fig. 4.

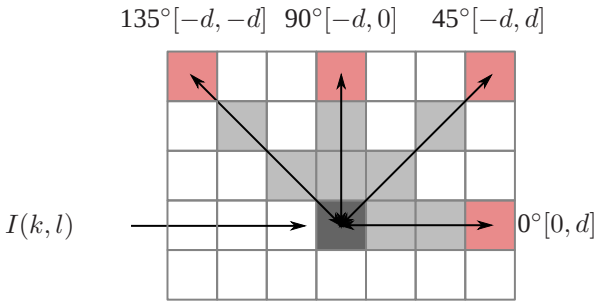


Fig. 4. GLCM cell evaluation showing orientation and step size

There are a number of different texture features that can be extracted from the GLCM matrix $p_{\theta,d}$, with a list of features proposed in [Haralick *et al.*, 1973], the features that are utilised within this method include: energy, contrast, correlation, homogeneity, information measure of correlation 1 and information measure of correlation 2 as well as maximum probability, which have previously been used in [Woods *et al.*, 2013]. The features are expressed as a vector for each pixel with the form $f_{\theta,d}(k)$, where θ is the angle and d is the step distance for the GLCM feature and $k = 1 \cdots 7$, one for each GLCM feature.

To group the related features, the matrix $t_d(k, a)$ is defined as given in (2).

$$t_d(k, a) = f_{\theta(a),d}(k) \qquad \theta(a) = 0°, 45°, 90°, 135° \qquad a = 1, 2, 3, 4 \tag{2}$$

The GLCM features are generated for each of the four directions θ that are used in the GLCM generation. There exists a number of different methods [Kruger *et al.*, 1974; Soh and Tsatsoulis, 1999; Chai *et al.*, 2011] for reducing the dimensionality of data in order to make classification less expensive. The methods focus on metrics that compare the different θ angles for each of the different texture features, the metrics that are utilised are the mean, variance and span as described in equations (3), (4) and (5), where $t_d(k, a)$ represents the vector of features k, of the step distance d in the ath direction as before.

$$\bar{t}_d(k) = \frac{1}{4} \sum_{a=1}^{4} t_d(k, a) \tag{3}$$

$$v_d(k) = \frac{1}{4} \sum_{a=1}^{4} \left(t_d(k, a) - \bar{t}_d(k)\right)^2 \tag{4}$$

$$s_d(k) = \max\{t_d(k, a)\} - \min\{t_d(k, a)\} \tag{5}$$

The spatial feature image is developed from the range residual image and consists of GLCM texture features, in addition to the standard deviation of the patch over the region of interest, which is described in (6).

$$\bar{\sigma} = \sqrt{\frac{1}{W^2} \sum_{i=1}^{W} \sum_{j=1}^{W} (R(i - \frac{W}{2}, j - \frac{W}{2}))^2} \tag{6}$$

The GLCM texture features that are generated use the range residual image for a region of interest of fixed size, quantised into 32 different levels. These levels are chosen so that the maximum and minimum levels correspond to 5mm above the dominant surface and 5mm below the dominant surface. The spatial feature image consists of the following feature vectors for each pixel,

$$\begin{aligned} X_T = [&\bar{\sigma}, \\ &\bar{t}_1(k), v_1(k), s_1(k), \\ &\bar{t}_2(k), v_2(k), s_2(k)] \\ &k = 1 \ldots 7 \end{aligned} \tag{7}$$

Gabor Filter Features

The Gabor filter bank operate by looking at the different orientations and scales for the image and taking the filtered response for image data. Gabor filter based textures are developed by convolving the scene with a variety of different Gabor filters and then taking the filtered responses as features in order to distinguish different textures from a scene. A number of different Gabor filters can be seen in Fig. 5, where the different filters correspond to different orientations and wavelengths of the filter.

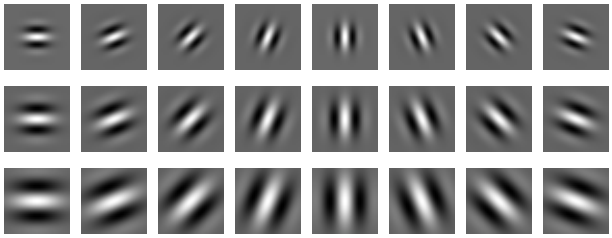


Fig. 5. Gabor Filter Bank with 8 different orientations and 3 different scales

The filter can be applied over the image as a whole or on a segment of the image as a window filter. The gabor filter response on a windowed segment contains a large amount of information and so to reduce the amount of data the response of a region can be reduced to the mean, standard deviation and span between the maximum and minimum of the filter response, the gabor filtered features can be seen in (8).

$$g_\mu(a, k) = \frac{1}{W^2} \sum_{i=1}^{W} \sum_{j=1}^{W} g_{a,k}(i, j) \tag{8a}$$

$$g_\sigma(a,k) = \frac{1}{W^2}\sum_{i=1}^{W}\sum_{j=1}^{W}\left(g_{a,k}(i,j) - g_\mu(a,k)\right)^2 \quad (8b)$$

$$g_{span}(a,k) = \max\{g_{a,k}(i,j)\} - \min\{g_{a,k}(i,j)\} \quad (8c)$$

with $g_{a,k}$ being the response of the gabor filter using orientation a and scale k, local coordinates in i and j and W being the width of the square region that the filter is applied to.

$$X_G = [\bar{\sigma}, g_\mu(a,k), g_\sigma(a,k), g_{span}(a,k)]$$
$$k = 1\ldots3, a = 1\ldots8 \quad (9)$$

Radial Power Spectral Density Features

The fourier transform features for a surface are straightforward to evaluate; however, the responses are in 2 dimensions. This means that there are a large number of features, with the effect of orientation dependent frequency responses. Power Spectral Density (PSD) is most often used to form features from a fourier transformed signal, and was used in the terrain classification method proposed by Wang [Wang *et al.*, 2012]. For comparison to previous work, it is therefore necessary to be able to generate PSD based features but from the 2D fourier transform. This can be achieved through the use of a radially averaged PSD, where the 2D image is converted into polar coordinates with bins at different radii being used to reduce the dimension of the fourier transform output.

$$P(k) = \frac{1}{N_k}\sum_{r_k<i<r_k+\delta r}\hat{I}(\xi = i)^2 \quad (10)$$

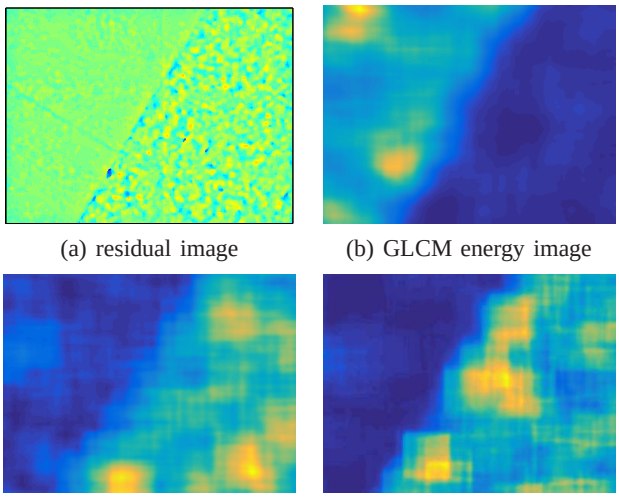
where $\hat{I}(\xi)$ is the fourier transform of the input 2D matrix, N_k is the number of elements i that are within the boundaries r_k and $r_k + \delta r$ for each set k.

$$X_P = [\bar{\sigma}, P(1), P(2), \cdots, P(16)] \quad (11)$$

The resulting feature images from each of the three different techniques can be seen in Fig. 6, where a typical residual image is shown as well as the typical output residual images from the GLCM textures, the Gabor filter responses and the Radial Power Spectral Density features. The feature images that are shown are only indicative of only one of the feature dimensions.

2.2. Dimensionality Reduction

The feature space vectors that are generated from the range data: Haralick textures, Gabor Filter and Radial Power Spectral Density descriptors is of a very high dimension. This high dimensionality in the feature space means that generating the mean and covariance for each region in order to classify it becomes costly. In order to both increase the reliability of the classification results and to reduce the computational time, dimensionality reduction should be carried out.



(a) residual image (b) GLCM energy image

(c) Gabor $g_\mu(1,1)$ image (d) RPSD $P(2)$ image

Fig. 6. Images showing a typical residual and the corresponding non-semantic feature images

The method of Multiple Discriminant Analysis (MDA) [Murtagh and Heck, 2000] is used to reduce the dimensionality and increase the variance between the classes. MDA is utilised in favour of Principle Component Analysis (PCA) as results that were obtained in comparison showed MDA outperforming PCA by a margin of 10% in the classification accuracy. The output of MDA is a basis vector of rank $n-1$, where n is the number of classes that is present in the dataset.

2.3. Classification

There are a number of different statistical distances that can be used in providing a classifier metric. There are metrics that measure the distance directly, such as the Bhattacharyya distance, whereas other metrics measure the divergence between two distributions, such as the symmetric and asymmetric Kullback-Leibler divergences and the Jensen-Shannon divergence.

The Bhattacharyya distance shown in (12) is used in this application, which is taken from [Kailath, 1967]. The Bhattacharyya distance is used in preference to other statistical distance metrics as a comparative analysis between the aforementioned statistical distances showed that all four had similar classification accuracies. The Bhattacharrya distance is chosen so as to reduce the computational complexity. The equation depends on the means of the distribution P and a target distribution Q, $\boldsymbol{\mu}_P$ and $\boldsymbol{\mu}_Q$, as well as the covariances of the distributions $\boldsymbol{\Sigma}_P$ and $\boldsymbol{\Sigma}_Q$,

$$d_b = \frac{1}{8}\left(\boldsymbol{\mu}_Q - \boldsymbol{\mu}_P\right)^{\boldsymbol{T}} \boldsymbol{\Sigma}^{-1}\left(\boldsymbol{\mu}_Q - \boldsymbol{\mu}_P\right) + \frac{1}{2} ln\left(\frac{|\boldsymbol{\Sigma}|}{\sqrt{|\boldsymbol{\Sigma}_P|.|\boldsymbol{\Sigma}_Q|}}\right) \quad (12)$$

where $\boldsymbol{\Sigma} = \frac{1}{2}\left(\boldsymbol{\Sigma}_P + \boldsymbol{\Sigma}_Q\right)$, which represents the combination of the covariances of the distributions P and Q.

2.4. Confidence Metric

In order to provide an accurate representation of the terrain type that is being classified, an additional component of information that is generated is how confident the classifier is in terrain classification for a particular region.

There are a number of different confidence metrics that have been considered; relative inverse distance, weighted confidence metric as well as a comparative distance [Smith *et al.*, 1994]. The inverse distance is meant to give a confidence metric that relates to the classified class and how close that classified terrain is to the other terrain types. This metric can be calculated using (13).

$$\chi = \frac{\sum_i^N d_b(i)}{min(\boldsymbol{d}_b)}, \quad (13)$$

where $\boldsymbol{d}_b$ is a vector of Bhattacharyya distances (12) to all current terrain classes N.

2.5. Algorithm

The terrain classification method utilises the transformed feature space images that were generated using the MDA transformation method. The feature space image is then used to calculate the distances between a subregion of the image and the available terrain classes. The subregions are classified through the use of the Bhattacharyya distance, with the classifier selecting the class with the lowest distance for that region.

The minimum size of the segments used for classification is 8×8, as this amount provides enough data so that accuracy in the mean and covariance can still be maintained [Schäfer and Strimmer, 2005]. If the size of the segments are too small, then there is only a small amount of data to draw from to create an accurate mean vector $\boldsymbol{\mu}_k$ and covariance matrix $\boldsymbol{\Sigma}_k$. This can be seen in the algorithm as limiting the parent node to a size of no smaller than 16×16, which guarantees that the children of that node will be at least 8×8.

The proposed Algorithm 1, shown below, is the method by which the feature space image is segmented and classified. The resulting terrain model T contains the classification results from the implementation of this algorithm. The method is applied over the generated feature space image, where the frame of the feature space is used to constrict the splitting of the resulting quad-tree.

For each classified parent and children subregions the classification confidence is calculated. This confidence metric is then used to identify the terrain areas that were classified with high confidence. Areas where the classification confidence is less than a specified threshold χ_{th} are left unclassified.

Algorithm 1 Segment and Classify the Feature Space Image

Calculate the feature space image $F(^*)$
set stack queue Q to $\emptyset$
Push F onto Q
while Q is not empty **do**
  Pop parent subregion n from Q
  Calculate children subregion vector c from n
  if n is partially in frame **then**
    if n is not completely in frame and $n_{\text{width}} > 16$ **then**
      Push children subregion c on Q
    else
      Calculate $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ for n and c
      for all Class types **do**
        Evaluate the distance d_b for n and c
      end for
      Classify n and c using d_b to each class
      if $n_{\text{class}} \neq c_{\text{class}}$ or $|c_{\text{class}}| > 1$ **then**
        Push children subregion c vector onto Q
      else
        calculate confidence χ
        Set $T_{class}\forall(i,j) \in n$ to n_{class}
        Set $T_{confidence}\forall(i,j) \in n$ to χ
      end if
    end if
  end if
end while

* computationally expensive calculation

3. Results

3.1. Terrain Classification

The results of the terrain classification through the use of the ERTA method for the three different descriptors can be seen in Fig. 7, the classification accuracies that are presented are taken with a zero confidence threshold. The relationship between the classification accuracy and the region of interest size seems to indicate a slight increase as the region of interest size increases, however this effect varies. The best classification accuracy for X_T is 82.7 % for a region of interest size of $W = 72$, for X_P a classification accuracy of 80.6% at $W = 64$ is best and for X_G the best classification accuracy of 76.7% is achieved at $W = 32$.

This result can be further enhanced by applying different confidence threshold values, given the relationship between the confidence threshold and the classification accuracy for

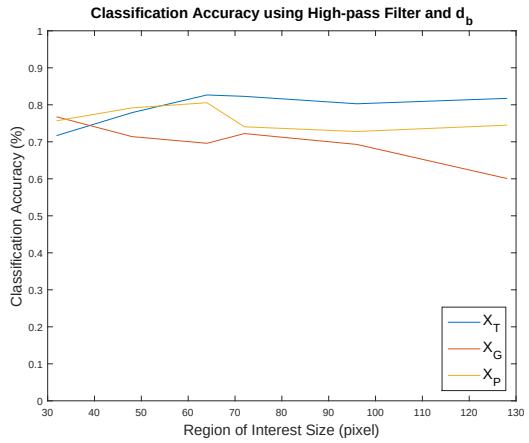


Fig. 7. Classification results for the three different descriptors using the high-pass filter residual and the Bhattacharyya distance metric

X_T, as shown in Fig. 8. The figure also shows the relationship between the confidence threshold and the percentage of the scene that is classified. As the confidence threshold becomes more restrictive the classification accuracy approaches 100%, however the majority of the scene is left unclassified. A compromise between the classification accuracy and the scene classification is shown in Table I, where a confidence threshold $\chi_{th} = 40\%$ is used, this compromise results in a classification accuracy of 85.1% with a scene coverage of 89.4% for a region of interest size of 72.

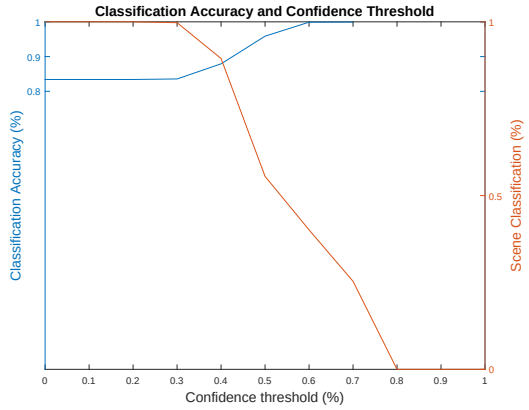


Fig. 8. Classification accuracy and percentage scene classification as a function of confidence threshold for the X_T descriptor

The misclassification of the different terrain types can be seen to consist of terrain types that have similar features, for example, the tiled and concrete surfaces, the artificial turf and gravel surfaces all consist of similar sized features and hence lower the success rates. The grass terrain type was the most varied type and so there were a large number of false positives, and a few false negatives, this is due to the large variability that exists within the grass terrain type surface.

The outcome of this method can be seen in Fig. 9, which has the same configuration as the data shown in the confusion matrix in Table I, with a descriptor vector X_T and with a confidence threshold of 40%.

These results show that the algorithm is able to adequately classify and segment the terrain such that an unknown boundary between different terrain types can be extracted. The environment that is present in these figures consists of concrete on the left and artificial turf on the right with a boundary between the two that splits the image into two distinct sections.

3.2. Terrain Segmentation

The classification of terrain near a boundary between different terrain types was investigated to observe whether a significant relationship exists between the size of the region of interest and the classification accuracy near to a boundary. To extract the relationship from the classification results it was necessary to be able to calculate the discretised distance from a boundary in each of the scenes, which was counted in pixels. Then the total number of elements for each discretised distance value was determined, as well as the total amount of correctly classified elements was calculated.

The terrain classification around the boundary between different terrain types can be seen in Fig. 10. Note that there is no clear relationship between the distance from a boundary and the classification accuracy. All region of interest sizes have similar improvements in their classification accuracies as the distance from the boundary increases.

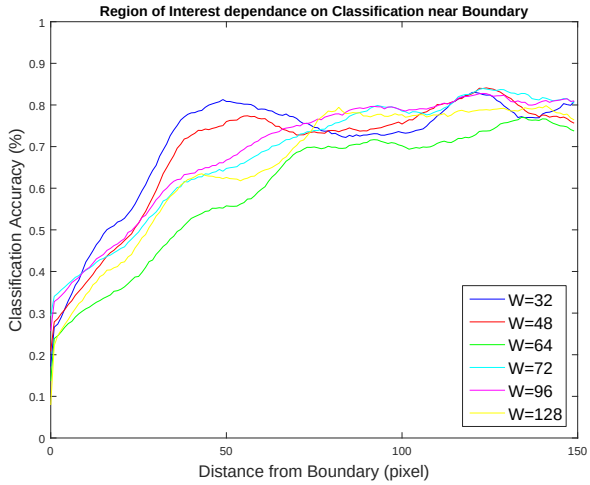


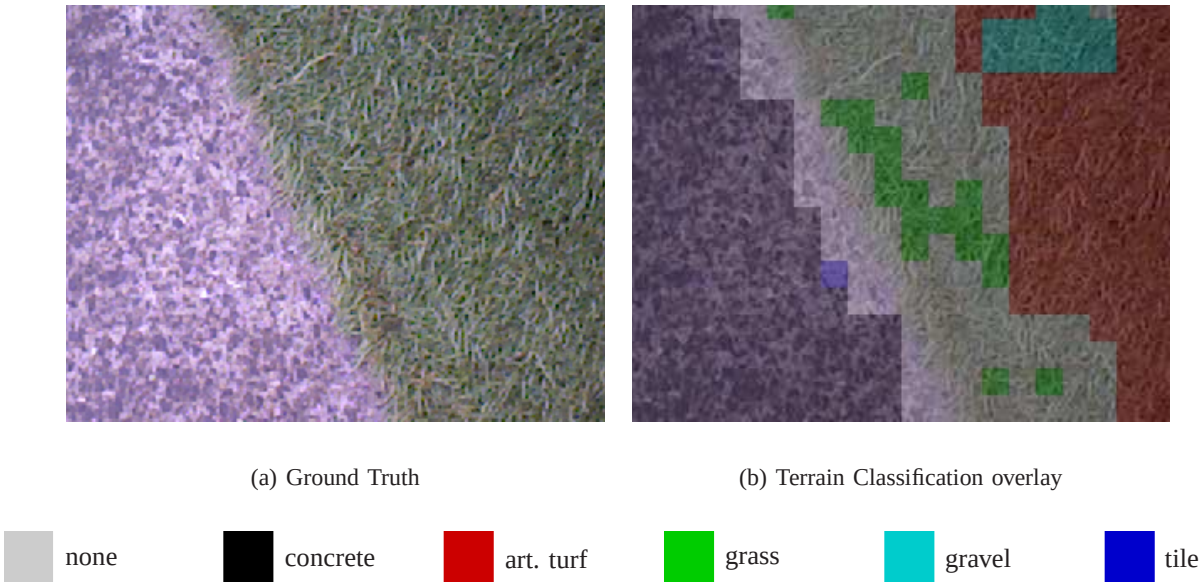
Fig. 10. Classification Accuracy from X_T descriptor as the distance from a terrain Boundary for the different region of interest sizes

3.3. Computation Time

The computation time of this method is of significant importance as the time taken to evaluate each image impacts directly on the applicability of the method. The computation

	concrete	fakeGrass	grass	gravel	tile	precision
concrete	0.630	0.000	0.000	0.000	0.123	0.837
fakeGrass	0.000	0.922	0.000	0.072	0.000	0.928
grass	0.105	0.101	1.000	0.000	0.000	0.829
gravel	0.000	0.004	0.000	0.931	0.012	0.983
tile	0.357	0.000	0.000	0.000	0.843	0.703
recall	0.630	0.922	1.000	0.931	0.843	0.851

TABLE I
CONFUSION MATRIX FOR $X_T, W = 72$ WITH $\chi_{th} = 40\%$



(a) Ground Truth

(b) Terrain Classification overlay

Fig. 9. Example terrain classification of X_T descriptor with a confidence threshold of $\chi_{th} = 40\%$ with two different terrain types. On the left is concrete and on the right is artificial turf, with the colours representing gray: not classified, black: concrete, red: artificial turf, green: grass, cyan: gravel, blue: tile.

time results are shown in Table II as a function of the region of interest window size. The implementation for the GLCM texture features are performed entirely within C implementation. The computer that was used was an Intel i5-2500 CPU with a clock speed of 3.3GHz running Windows 7 64bit. The timing is found by recording the time it takes for 10000 calculations and then finding the average computation time.

The overall computation time result showed that the method is unable to work in real-time, with an input image of 320x240 the time required to calculate the feature image would be 7.6 seconds for the GLCM textures and 12.4 seconds for the Radial PSD features, both with a region of interest size of 72. The Gabor filter implementation is an adapted implementation provided from [Haghighat *et al.*, 2015], which has a significantly greater computation time than the other two methods, as it is necessary to apply the 24 separate filters to the image.

As would be expected, as the region of interest size increases the computation time of both methods increases. As the region of interest size increases, more data is included in the feature extraction and so it increases the amount of time required to evaluate that pixel. The difference in computation time between the different non-semantic features are due to the data size and processing required for each method.

4. Conclusion

In this paper we have proposed a novel method using non-semantic range data to perform terrain classification and terrain type segmentation, together with a measure of classification confidence. The introduction of a classification confidence metric provides a useful tool in assessing the amount of knowledge a system has on a particular surface, with the behaviour of an AGV able to adapt to the confidence level of each surface. With the proposed method, terrain classification can be achieved and is demonstrated in this paper for the entire scene irrespective of geometry with terrain segmentation dependant on the terrain texture information.

	32	48	64	72	96	128
GLCM	0.054	0.068	0.089	0.10	0.16	0.23
Gabor	3.54	6.04	8.05	9.87	17.0	37.4
RPSD	0.0988	0.117	0.144	0.161	0.204	0.317

TABLE II

PIXEL COMPUTATION TIME IN MILLISECONDS (ms) AS A FUNCTION OF THE REGION OF INTEREST SIZE

The inclusion of the classification confidence metric allows for the classification accuracy of the scene to be improved by discarding ambiguous regions. It is shown that the proposed method using range-only data enables changes in terrain types to be identified irrespective of the visual appearance or semantic spatial information, with the non-semantic method providing a classification accuracy of 85.1% over the dataset with a confidence threshold of 40%.

Additional work is planned that will enable the method to run in real-time through the use of hardware acceleration from a Graphics Processing Unit as well as optimisations in serial calculations of the GLCM texture features. The method would also benefit from an improved confidence metric that can better distinguish the available terrain types.

The dataset that was used in this paper can be found at the following web address www.agile-robotics.com/data/

References

[Angelova *et al.*, 2006] Anelia Angelova, Larry Matthies, Daniel Helmick, and Pietro Perona. Slip prediction using visual information. *In Proceedings of Robotics Science and Systems*, 12:15, 2006.

[Brooks and Iagnemma, 2007] C.A. Brooks and K.D. Iagnemma. Self-supervised classification for planetary rover terrain sensing. In *Aerospace Conference, 2007 IEEE*, pages 1–9, 2007.

[Chai *et al.*, 2011] Hum Y Chai, Lai K Wee, Tan T Swee, Sh-Hussain Salleh, AK Ariff, et al. Gray-level co-occurrence matrix bone fracture detection. *American Journal of Applied Sciences*, 8(1):26, 2011.

[Cho *et al.*, 2010] Wanki Cho, Jangyeol Yoon, Seongjin Yim, Bongyeong Koo, and Kyongsu Yi. Estimation of tire forces for application to vehicle stability control. *Vehicular Technology, IEEE Transactions on*, 59(2):638–649, Feb 2010.

[Haghighat *et al.*, 2015] Mohammad Haghighat, Saman Zonouz, and Mohamed Abdel-Mottaleb. Cloudid: Trustworthy cloud-based and cross-enterprise biometric identification. *Expert Systems with Applications*, 42(21):7905 – 7916, 2015.

[Haralick *et al.*, 1973] R.M. Haralick, K. Shanmugam, and Its'Hak Dinstein. Textural features for image classification. *Systems, Man and Cybernetics, IEEE Transactions on*, SMC-3(6):610–621, 1973.

[Kailath, 1967] T. Kailath. The divergence and bhattacharyya distance measures in signal selection. *Communication Technology, IEEE Transactions on*, 15(1):52–60, February 1967.

[Kragh *et al.*, 2015] Mikkel Kragh, RasmusN. Jrgensen, and Henrik Pedersen. Object detection and terrain classification in agricultural fields using 3d lidar data. In Lazaros Nalpantidis, Volker Krger, Jan-Olof Eklundh, and Antonios Gasteratos, editors, *Computer Vision Systems*, volume 9163 of *Lecture Notes in Computer Science*, pages 188–197. Springer International Publishing, 2015.

[Kruger *et al.*, 1974] R.P. Kruger, William B. Thompson, and A Franklin Turner. Computer diagnosis of pneumoconiosis. *Systems, Man and Cybernetics, IEEE Transactions on*, SMC-4(1):40–49, Jan 1974.

[Lin *et al.*, 2008] P.P. Lin, Maosheng Ye, and Kuo-Ming Lee. Intelligent observer-based road surface condition detection and identification. In *Systems, Man and Cybernetics, 2008. SMC 2008. IEEE International Conference on*, pages 2465–2470, Oct 2008.

[Murtagh and Heck, 2000] Fionn Murtagh and A Heck. Multivariate data analysis with fortran, c and java code. *Northern Ireland: Queens University Belfast, Astronomical Observatory Strasbourg*, page 272, 2000.

[Robledo *et al.*, 2011] Alicia Robledo, Stephen Cossell, and Jose Guivant. Outdoor ride: Data fusion of a 3d kinect camera installed in a bicycle. In *Proceedings of the 2011 Australasian Conference on Robotics & Automation, Melbourne, Australia*, 2011.

[Schäfer and Strimmer, 2005] Juliane Schäfer and Korbinian Strimmer. A shrinkage approach to large-scale covariance matrix estimation and implications for functional genomics. *Statistical applications in genetics and molecular biology*, 4(1), 2005.

[Smith *et al.*, 1994] S.J. Smith, M.O. Bourgoin, K. Sims, and H.L. Voorhees. Handwritten character classification using nearest neighbor in large databases. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 16(9):915–919, Sep 1994.

[Soh and Tsatsoulis, 1999] L.-K. Soh and C. Tsatsoulis. Texture analysis of sar sea ice imagery using gray level co-occurrence matrices. *Geoscience and Remote Sensing, IEEE Transactions on*, 37(2):780–795, Mar 1999.

[Walas, 2013] Krzysztof Walas. Terrain classification using vision, depth and tactile perception. *Proceedings of the 2013 RGB-D: Advanced Reasoning with Depth Cameras in Conjunction with RSS, Berlin, Germany*, 27, 2013.

[Wang *et al.*, 2012] Shifeng Wang, Sarath Kodagoda, and Ravindra Ranasinghe. Road terrain type classification based on laser measurement system data. *Proceedings of the 2012 Australasian Conference on Robotics & Automation*, 2012.

[Woods *et al.*, 2013] Michael Woods, José Guivant, and Jayantha Katupitiya. Terrain classification using depth texture features. In *Proceedings of Australasian Conference on Robotics and Automation*, 2013.

[Yi *et al.*, 2014] Chuho Yi, Donghui Song, and Jungwon Cho. Self-supervised sensor learning and its application: Building 3d semantic maps using terrain classification. *International Journal of Distributed Sensor Networks*, 2014, 2014.

[Zhao *et al.*, 2014] Gangqiang Zhao, Xuhong Xiao, Junsong Yuan, and Gee Wah Ng. Fusion of 3D-LIDAR and camera data for scene parsing. *Journal of Visual Communication and Image Representation*, 25(1):165–183, 2014.