International Journal of Advanced Research in Education & Technology (IJARET)

Vol. 4, Issue 2 (April - June 2017)

ISSN : 2394-2975 (Online)
ISSN : 2394-6814 (Print)

Detection of Drivable Path in LIDAR image using Gray Level Neighbours Matrix and K- Nearest Neighbour

[I]Biji Yohannan, [II]D.Abraham Chandy, [III]A.Hepzibah Christinal

[I]PG Scholar, Dept. of Electrical Technology, Karunya University, Coimbatore, India.

[II,III]Associate Professor, Dept. of Electrical Technology, Karunya University, Coimbatore, India.

Abstract

Autonomous vehicles are used greatly for range of task like automated highway driving, transporting work in process etc. This work presents algorithms for terrain classification which includes different stages like Gray level conversion, feature extraction and Classification process. The inputs to the process is acquired from rotating LIDAR sensors. The objective of this proposed method is to detect the drivable path using feature extraction through GLNM (Gray Level Neighbour Matrix) techniques and classified using k-NN classifier. The method jointly determines the ground surface and segments individual objects into cells including overhanging structures.

Keywords

Gray Level Neighbour Matrix, k – Nearest Neighbour, LIDAR sensors, Autonomous navigation, Unmanned Ground Vehicles.

I. Introduction

The most significant challenges faced by Unmanned Ground vehicles is that the road has unstructured and harsh environment. These challenges can only be overcome by the autonomous navigation in an unstructured environment that will enable the commercial exploitation of robotics. Autonomous Navigation System(ANS) was the combat vehicles that has the capability to upgrade the manned vehicle to unmanned vehicles. The unstructured environment includes obstacles like tree, bush, bumps, other vehicles, buildings etc. So that it is important to detect the drivable path through these obstacles. The autonomous robot applied for military applications, industrial applications, etc. The significance of this work concentrates to detect the drivable path for Unmanned Ground Vehicles(UGV) through image fusion, feature extraction and classification techniques by classifying the path whether drivable or not.

Based on the literature made, according to [1] Dominik Steinhauser et.al., continues with the preprocessing steps to discard all the non-reliable points and will classify the path from an obstacle and finally the ego-motion detected using RANSAC algorithm. It results in Data processing: 0.55s Drivable road estimation: 0.49s. In [2], Frank Moosmann et.al., says the 3D LIDAR get segmented using graph based approach with Local convexity region. The algorithm can successfully merge the road segment. The processing time for normal it takes 0.352s for Segmentation 0.25s and finally for classification it takes 0.021s. According to [3], F. Samadzadegana et.al., present a road extraction method from both intensity and height information from LIDAR data and Fuse the result of classifiers and get more accurate result. It classifies the path using Multiplier Classifier Systems and Optimum Classifier Selection. In [4], A.K. Aijazi et.al., compares the result with two approach. The first approach follows the segmentation process using Watershed approach followed by classification through SVM classifier. The second approach segmented the results using Super-voxel approach and classified the path using local descriptors. Finally, by comparing the results. The mathematical morphology based method, constrained by the generated profile, also fails to segment out 3D ground points directly under the motorcycles and car. This is not an issue for the super-voxel based method relying on local descriptors i.e. colour, reflectance intensity and surface normal. The results show that the building and ground are much better in mathematical morphology based method while the detection quality performance for cars, poles, and other road furniture is much more superior for the super-voxel based method. According to [15], D. Abraham Chandy et.al., proposes a new approach for feature extraction technique called Gray Level Neighbour Matrix (GLNM) for the application of mammogram retrieval. In this method, the results acquired from GLNM techniques compared with the Gabor, CDF 9/7 and Db4 wavelets, GLAM and GLCM based texture extraction methods and found that GLNM provides accurate results.

The objective of this paper is to detect the drivable path and classify the results through different stages. The input is from 3D LIDAR Sensor (Velodyne sensor), the image is then converted into gray level image followed by Feature extraction through GLNM techniques and finally classified the results through k-NN classifiers.

This paper organized as follows: In Section 2, it explains the methodology flow used in this project. In section 3, the results and discussion have been demonstrated and finally concluded the project in Section 4.

II. Methodology

The overview of methodology to detect the drivable path from the obstacles for Unmanned Ground Vehicle is shown in Fig. 1, which includes the important stages like Feature extraction and classification.

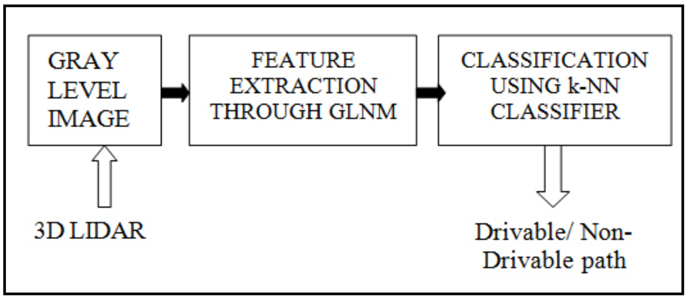

Fig.1: Overview for Path detection.

1. Gray Level Image

The input to the system is from 3D LIDAR Sensor which was acquired from Velodyne 64 HDL-E. The data is then converted into gray scale image. The LIDAR image is shown in Fig. 2.

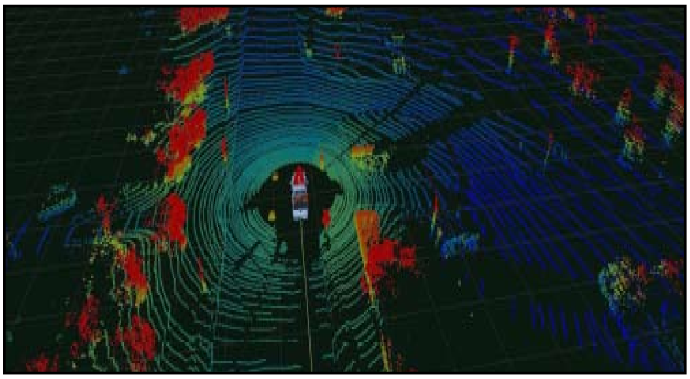

Fig. 2: 3D LIDAR image

2. Feature Extraction

The fused image is represented into 16 X 16 cells. From each cell the feature is extracted through GLNM technique. The GLNM is the Statistical textural representation methods which characterize the texture of an image by the statistical distribution of the image intensity.

The GLNM technique can extract all the textural information in an image which includes the size information of texture elements and one important fact is that it concentrates on the gray level neighbours which occur within the specified neighbourhood. It extracts mainly four texture features are uniformity (F1), gray level proportionality (F2), rough textures (F3) and texture pattern (F4). The matrix element (i, j) of the GLNM in Eq (1) is the 'j' number of neighbours within the given neighbourhood which having the intensity 'i', is defined as,

$$G(i,j) = \#\{(x,y)|\ I(x,y) = I, \#[(p,q)|N_{xy}(p,q) = i] = j\} \quad (1)$$

where, # denotes the number of elements in the total set and Nxy (p, q) is the neighbourhood which defined in the image.

(a) FEATURE (F1): The uniformity of an image is measured by using Eq. (2). A uniform image will contain few gray levels, giving a GLNM with only a few but with high values of G (i, j), which results to give high value for F1. The expression for F1 is given below,

$$F1 = \sum_{i=1}^{N_g} \sum_{j=1}^{N_n} \{G(i,j)\}^2 \quad (2)$$

where, Ng is considered as the maximum number of gray levels in an image and Nn is considered as the maximum number of gray level neighbours for the specified neighbourhood.

(b) FEATURE(F2): The F2 provides a gray level proportionality feature by using Eq. (3). This feature will capture the proportionality between the number of gray level neighbours and gray levels in an image.

$$F_2 = \sum_{n=0}^{N_g-1} n^2 \left(\sum_{|i-j|=n} G(i,j)\right) \quad (3)$$

(c) FEATURE(F3): The F3 provides rough textures in an image and calculated using Eq. (4). It is to be noted that the term (i−j) in Eqs. (3) and (4) serves the purpose of selecting specific entries of GLNM.

$$F_3 = \sum_{i=1}^{N_g} \sum_{j=1}^{N_n} \frac{G(i,j)}{1+(i-j)^2} \quad (4)$$

(d) FEATURE(F4): The F4 provides textural pattern of an image by Eq. (5). Also, the textural pattern is dominant in an image.

$$F4 = \sum_{i=1}^{N_g} \sum_{j=1}^{N_n} (min_j - \mu)^2 G(i,j) \quad (5)$$

Using these formulas, the feature is extracted using Gray Level Neighbour Matrix.

3. Classification

The k-NN is an algorithm that will store all available cases and classifies new cases depending on a similarity measure. It is conceptually simple but the only disadvantage is that it is time consuming process. The classification process starts by initializing and defining the value of k (k = Kernel size; k = '3' or '5'). The training and test samples are given to the classifier from the features extracted from GLNM (Gray Level Neighbour Matrix) technique, thus the classifier will compute the distance between input and the training sample. Then, the system will sort the distance so that it can able to find the K nearest neighbors. Finally, it will apply the simple majority to find the system belongs to which class. For k-NN classification "knnclassify" command used in MATLAB.

III. Results And Discussions

The experimentation for this work conducted with the dataset collected from Ford campus vision dataset. The whole process is simulated using MATLAB.

The image is then represented into cells of 16 × 16. This representation will make us to verify the path and the obstacles clearly. Each block will represent either path or obstacles. So that it makes ease the classification process. The cell division is shown in Fig. 3.

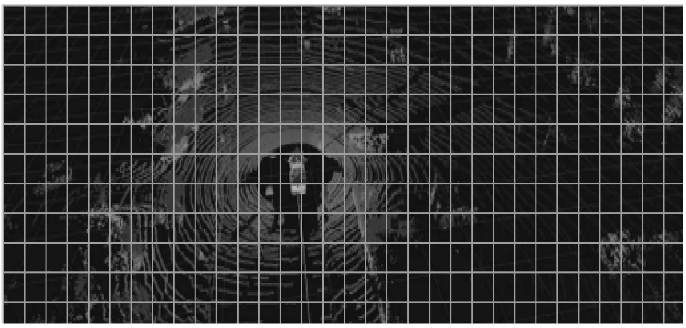

Fig. 3: Cell division

Now, the feature is extracted from each block of an image using Gray Level Neighbour Matrix (GLNM). It will provide for textural features from various block like car, path, building, tree. It will extract four features like uniformity (F1), gray level proportionality (F2), rough textures (F3) and texture pattern (F4) from an image. The extracted features are shown in Table. 1.

Table 1. GLNM Feature extraction.

Cell. No	Object type	Features			
		Uniformity	Gray level proportionality	Rough textures	Textural pattern
10	TREE	2348	2115	983	75
96		2904	2309	944	72
45	ROAD	3216	1477	691	99
46		3197	1566	867	93
207	CAR	21237	8355	2723	21
175		21759	8055	2639	27
50	PILE	3099	1928	837	76
47		3617	1204	789	104

Finally, the result gets classified using k-NN classifier by using the function "knnclassify" in MATLAB. It classified into two classes such as obstacles as class 1 and path as class 2. The drivable path get faded by the green color. The vehicle can navigate through the drivable region. The classified results are shown in Fig. 4.

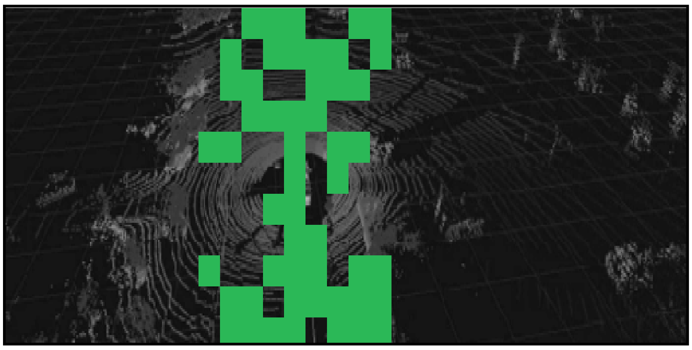

Fig. 4: Classified image using k -NN classifier.

IV. Conclusion and Future Scope

Thus, the drivable path get detected from the LIDAR image. The GLNM technique for feature extraction extracted four main textural features from road, car, building, tree and found to have convincing results. Finally, the path get detected and classified using k -NN classifier into drivable or non-drivable region.

Acknowledgement

The authors would like to thank Karunya University for providing necessary facilities to carry out the experimentation.

References

[1]. *Dominik Steinhauser, Oliver Ruepp, and Darius Burschka,"Motion Segmentation and Scene Classification from 3D LIDAR Data".IEEE, 398 – 403, 2008.*

[2]. *Frank Moosmann, Oliver Pink and Christoph Stiller," Segmentation of 3D Lidar Data in non-flat Urban Environments using a Local Convexity Criterion", IEEE,215-220, 2009.*

[3]. *F. Samadzadegana, B. Bigdelia, M. Hahnb"Automatic road extraction from lidar data based on classifier fusion in urban area", conference paper, 2009.*

[4]. *A.K. Aijazi, A. Serna, B. Marcotegui, P. Checchin and L. Trassoudaine," Segmentation and Classification of 3D Urban Point Clouds: Comparison and Combination of Two Approaches" 10th Conference on Field and Service Robotics, 2015*

[5]. *Biji Yohannan, D. Abraham Chandy, A. Hepzibah 'A Novel Approach for Fusing LIDAR and Visual Camera images of an Unstructured Environment' International Conference on Advanced Computing and Communication Systems(ICACCS), 2017.*

[6]. *Nikolaos Mitianoudis and Tania stathaki, "Adaptive image fusion using ICA Bases", Imperial College London, Exhibition Road, SW72AZ London, UK ,2006, pp. 829-832.*

[7]. *M Häselich, M Arends, D Lang, D Paulus, Terrain Classification with Markov Random Fields on fused Camera and 3D Laser Range Data. ECMR, 153-158, 2011.*

[8]. *Andrew Mastin, Jeremy Kepner, John W. Fisher, "Automatic Registration of LIDAR and Optical Images of Urban Scenes" CVPR, 2009.*

[9]. *Muralidhar, Ashwini, Saripalli, Srikanth, Papandreou-Suppappola, Antonia, et al. Augmented Image Classification using Image Registration Techniques, ASU Electronic Dissertations and Theses, 2011.*

[10]. *Robert M. Haralick, K. Shanmugam, and Its'hak Dinstein, "Textural Features for Image Classification", IEEE Transactions on Systems, Man, and Cybernetics, SMC-3 (6): 610–621,1973.*

[11]. *Nikolaos Mitianoudis and Tania stathaki, "Adaptive image fusion using ICA Bases", Imperial College London, Exhibition Road, SW72AZ London, UK, pp. 829-832, 2006.*

[12]. *VH Mistry, R Makwana "Survey: Vision based Road Detection Techniques" International Journal Computer Science and Information Technology, 2014, Volume 5, 4741-4747.*

[13]. *D. Puig, M. A. Garcia, "Automatic Texture Feature Selection for Image Pixel Classification," Pattern Recognition 39, 2006, pp. 1996– 2009.*

[14]. *Kolski, Sascha, Mario Bellino, and Roland Siegwart "Autonomous driving in structured and unstructured environments," IEEE Intelligent Vehicles Symposium, 2006.*

[15]. *D. Abraham Chandy, J. Stanly Johnson and S. Easter Selvan "Texture feature extraction using gray level statistical matrix for content-based mammogram retrieval", Springer, Multimedia Tools and Applications. pp. 2011-2024, 2014.*