An Improved Fast Ground Segmentation Algorithm for 3D Point Cloud

Zhixin Leng[1], Shu Li[2], Xin Li[1], Bingzhao Gao[1*]

1. State Key Laboratory of Automotive Simulation and Control, Jilin University, Changchun, 130025
E-mail: lengzx18@mails.jlu.edu.cn, lixinnjupt@126.com, gaobz@jlu.edu.cn

2. Aviation University of Air Force, Changchun, 130022
E-mail: maryannli@163.com

Abstract: In this paper, we propose an improved algorithm to divide the point cloud collected from Lidar into ground points and non-ground points. The primary purpose of our method is to improve the accuracy and guarantee data processing speed. In every frame of the point cloud, we divide the point cloud into different blocks basing on the vertical lines, and threshold points and ground initial points are then analyzed in each block. According to the priori approach, we complement two more restrictions to pick the threshold points and the ground initial points, which are height threshold and vertical distance. These conditions take the vertical extreme distance information into account. After the threshold points and ground initial points are labelled, all points between ground initial points and threshold points are ground points, and the other points are non-ground points. The whole process is completed after all points are labelled. We do some experiments based on our proposed method, and the algorithm can reach 22 frames per second. The results prove that our method has higher accuracy and the efficiency meets the real-time requirements.

Key Words: 3D point cloud, ground segmentation, threshold point, ground point, non-ground point

1 INTRODUCTION

Internet of things (IoT) is developing quickly all over the world [1-3]. The intelligent vehicle is one part of IoT systems. Several sensors are utilized to collect data from surrounding areas, such as cameras, radars, global positioning system (GPS), inertial measurement unit (IMU) and light detection and ranging (Lidar). The reliable perception of the local environment is one of the pivotal missions for intelligent vehicles. On the one hand, we need to improve the measurement accuracy of sensors, on the other hand, we continuously make the processing algorithm better to meet the need for real-time autonomous moving systems. The segmentation of the ground point and non-ground point is the first step that needs to be done no matter what cluster algorithms or other recognizing approaches are deployed. Since ground points segmentation pre-processing is the first step before detection classification and tracking, its quality affects the performances of the nest subsequent perceptual steps.

In recent years, many scholars and researchers have proposed different methods to separate the ground points and the non-ground points. Hernanez and Marcotegui raised an approach that works well in a flat urban environment [4], but the method is not very accurate in sloping terrain. The method mentioned in paper [5] proposes fan-shaped grid cell to separate the ground and non-ground point and cluster the non-ground points to be the obstacles based on the Euclidean point distances. [6] proposes a new method named Ground Plane Fitting, which utilizes some seed points and two thresholds to estimate the ground fitting equation and calculate the distance from points to the fitting equation and get the result of the ground points and non-ground points. This method has high precision but time complexity is also expensive. [7] proposes a new approach to segment ground and non-ground points, and gets a good result, its propose is to improve the efficiency and ensure the accuracy. But this method considers just three conditions to limit the selection of the threshold points. Apparently, more necessary conditions should be considered. Basing on the method proposed in [7], we add more limitations to this algorithm and test our improved approach on KITTI Dataset [8]. The Lidar data in KITTI dataset collected from the Velodyne HDL-64E contains about 120,000 points and we choose the Point Cloud Library (PCL) [9] for the visualization. In the case that LIDAR sensors have the abilities to capture 360 degrees of information, the data in dataset is organized in layers, and the points in each layer form a ring, which can reduce hardware resources for more computationally demanding processes in the whole autonomous driving systems [10-14].

Generally speaking, the Lidar signals are processed frame by frame, and the segmentation procedure contains two steps, one is extracting ground points, and the other is clustering the remaining non-ground points. Ground point separation is an important part of the obstacles recognition. So it is necessary to improve the accuracy, reduce the time complexity for later processes and research one new method to separate the ground points and non-ground points. Basing on one algorithm mentioned before [7], we utilize threshold points and ground initial points as technical means to

*Corresponding author: Bingzhao Gao

This work is supported in part by National Nature Science Foundation of China under grant No.61790564, in part by the China Automobile Industry Innovation and Development Joint Fund under Grant U1664257, in part by the National Natural Science Foundation of China under Grant 61520106008, in part by Program for JLU Science and Technology Innovative Research Team under grant 2017TD-20

978-1-7281-5855-6/20/$31.00 ©2020 IEEE

distinguish ground points and non-ground points. All points between ground initial points and threshold points are ground points, and the other points are non-ground points. In our research, two more restrictions are added to select the threshold points and the ground initial points. These two restrictions are named height threshold and vertical distance. These conditions take the vertical extreme distance information into account. Finally, we do some experiments on KITTI dataset, the algorithm can reach 22 frames per second and the final results show that our improved method has higher accuracy and the efficiency is also enough for the real-time performance.

2 SEGMENTATION METHODOLOGY

The following paragraph describes the processing flow of our improved fast ground segmentation algorithm. Above all, the overview of the ground segmentation algorithm is as followed. First of all, we need to collect data points from Lidar sensors, such as Velodyne sensors. Then according to the timestamp, we divide all collected points frame by frame. Here, we call one frame point cloud "one frame", the data structure of the point cloud is "xyzir", which x, y, z respectively represents the three-dimensional coordinates, "i" is intensity and "r" is the ring number. Theoretically speaking, the number of the points in one ring is fixed. However, because of the existence of "NaN", the number of points in one ring may be floating. The reason why "NaN" exists is that the laser does not return or other reasons. The next step is to extract useful points and get vertical lines. The number of the vertical lines is calculated by the physical parameters of the Lidar, specifically the horizontal resolution. After getting vertical lines, according to algorithm rules, the ground initial points and threshold points are then obtained. Finally, all points are labelled and the process starts again for a new loop until all points in one frame are labelled. The fundamental loop is shown in Fig. 1.

2.1 Calculate the vertical line

According to the horizontal resolution of the Lidar which is called r_{hor}, the number of the vertical lines calculated using formula below (1).

$$n_{ver} = \frac{2\pi}{r_{hor}} \times 180° \tag{1}$$

where n_{ver} is the count of the vertical lines. r_{hor} is the horizontal resolution of the Lidar.

We assume that the number of the points in each vertical line is fixed, which is equal to the number of laser rays. The vertical line in side view is shown in Fig. 2. "Point 0" is a point that the algorithm needs to specify, in fact it is an imaginary point in each vertical line. We assume that the first scan point (point 1) is the ground point, and the point mentioned before (point 0) is the reference point. It is obvious that point 0 and 1 are ground points, point 3, 4 and 5 are non-ground points, point 2 cannot be determined. Therefore, it is necessary to distinguish each uncertain point.

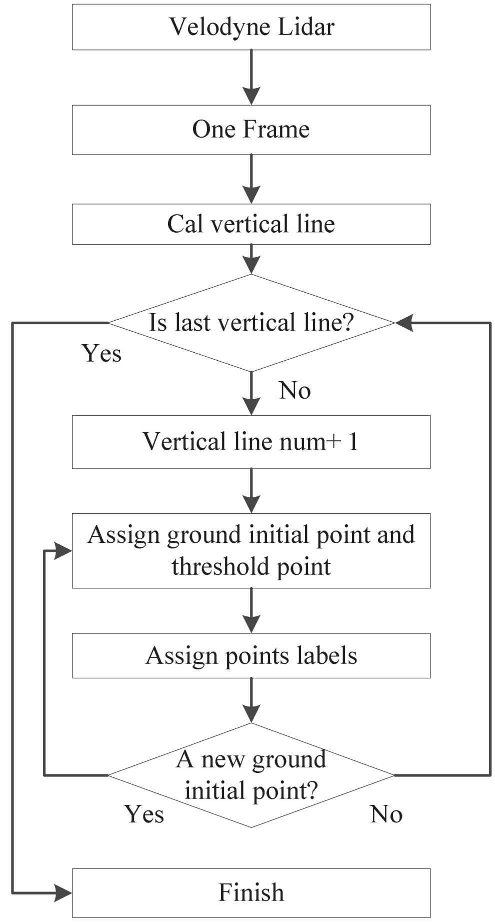

Fig. 1. The processing flow of the improved algorithm

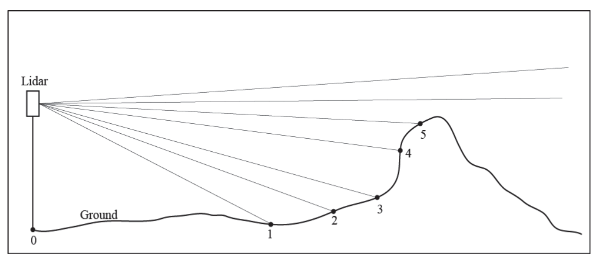

Fig. 2. A vertical line in side view

2.2 Improved Method to Find Threshold Points

The approach mentioned before mainly considers three features, which are gradient, lost threshold points, and abnormalities in the distance between the Lidar sensor and a particular threshold point [7]. In this section, basing on the method proposed before, we improve the algorithm performance by adding two more restrictions, which are called height threshold and vertical distance. Misjudgments exist due to benchmark factors in the judgment process. Height threshold and vertical distance are proposed to reduce misjudgments of threshold points. Height threshold is utilized as an auxiliary judgment method for gradient in order to improve the accuracy of threshold points judgment. Similarly, vertical distance is designed for abnormalities.

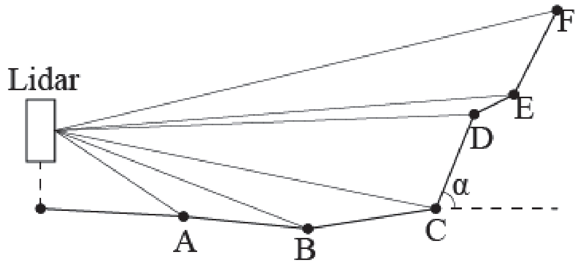

Fig. 3. Gradient which exceeds the limit value α

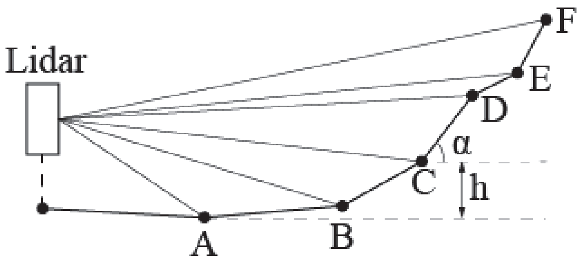

Fig. 4. Gradient which exceeds the limits value α and h

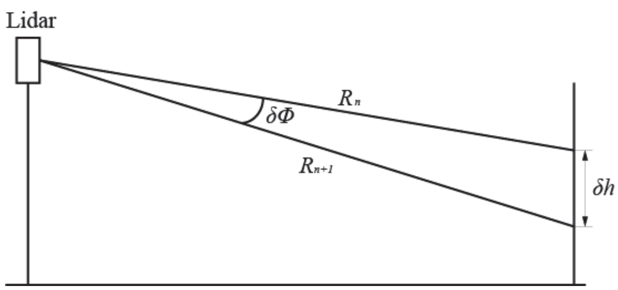

Fig. 5. Adjacent lasers in the radial direction

In first case, introducing height threshold Th_{hei} is introduced based on the gradient value α between two consecutive points. The fundamental method is shown in Fig. 3 and Fig. 4. The formulas for calculation are below (2) and (3).

$$\alpha = \arcsin\left(h/d\right) \tag{2}$$

$$h = h_c - h_a \tag{3}$$

In Fig. 3, basing on geometric relationship, if $\alpha > \alpha_{\max}$, we consider that point C is the threshold point, and point C is the ground point and point D is non-ground point. But in Fig. 4, it is obvious that the point C is not ground point. Proposed restriction height threshold is utilized here, calculating the vertical distance from point C to point A, if $h > h_{\max}$ and $\alpha > \alpha_{\max}$, the threshold point is point B, obviously, point C and D are non-ground points. In Fig. 5 and equation (4) (5) (6), getting the formula for calculating the value of $h_{\max}$.

$$\delta h = \sqrt{R_n^{\ 2} + R_{n+1}^{\ 2} - 2R_n R_{n+1}\cos(\delta\phi)} \tag{4}$$

where R_n is the distance from one Lidar point (assuming point n) to the laser geometric center, and R_{n+1} is the distance between the nearest point to point n and laser geometric center, $\delta\phi$ refers to the angle between R_n and R_{n+1}, which is determined by the parameters of the Lidar, δh is the vertical distance between point n and point (n+1). Considering the computational complexity, assuming that $R_n = R_{n+1}$, so equation (4) can be simplified to equation (5).

$$\delta h = R_n\sqrt{2(1-\cos(\delta\phi))} \tag{5}$$

$$h_{\max} = \delta h\left(l_c - l_a\right) \tag{6}$$

where δh is calculated by equation (5). In equation (6), l_c and l_a mean the label value of point a and point c in one vertical line. For example, like point 1 and point 3 in Fig. 2, the value of $h_{\max}$ is 2.

In order to distinguish whether the point C in Fig. 4 is a ground point or not, calculating $h_{\max}$ using δh calculated in equation (5), then calculating actual vertical distance h between point A and point C. If $h < h_{\max}$, point C is threshold point, on the contrary, point C is not threshold point. δh can be utilized to judge if the neighboring point belongs to one obstacle, but this approach cannot play a role here, which can be applied in clustering functions.

In second case, we introduce vertical distance basing on the abnormalities between two consecutive points. In Fig. 6 we show the improved algorithm. Calculating the vertical distance between point D and point A, if the vertical distance is shorter than the threshold calculated basing on equation (6), we consider that the point D is a threshold point, instead if the vertical distance is longer than threshold, in this case, we still need to analyze point C. According to equation (6), if the equation $h_c - h_a < h_{\max}$ is established, then we can conclude point C is a threshold point. The approach which utilizes two thresholds to find ground initial points just follows the method in paper [7], so we do not repeat that process again.

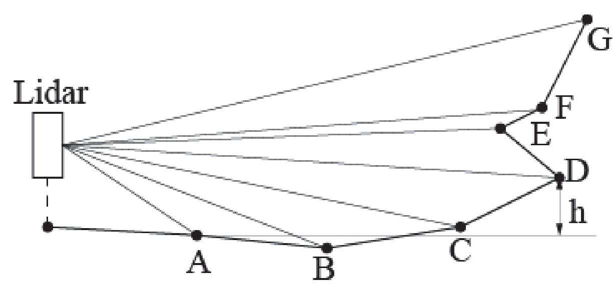

Fig. 6. Abnormality which exceeds the limits value h

3 Experiments and Analysis

In this section, some experiments are done to show the performance of our algorithm. In our experiments and analysis, we utilize KITTI dataset which was collected from a Velodyne HDL-64E sensor. According to our algorithm, a computer equipped with Inter Core i7-8700 3.2 GHz CPU and 8 GB RAM is applied. In our experiments making use of the original KITTI datasets, the ground points and the non-ground points are not fixed. In our experiments we choose $\delta\phi$=0.08° basing on the Velodyne sensors, and $\alpha_{\max}$=45° basing on practical testing. For different Lidar sensors or different application scenario, the value of the parameters can be different.

Fig. 7 to Fig. 9 is the separation result of one frame signal in KITTI dataset, in these three figures, the red parts are non-ground points and the white parts are ground points, and Fig. 10 to Fig. 12 is another frame. From the results of the segmentation, our improved algorithm for 3D point cloud has a certain effect, the road is well segmented and defined as ground, and the walls, trees, cars and pedestrians are defined as non-ground. Fig. 8 and fig. 12 show the side view of an inclined road respectively, proving that our improved approach is effective on the inclined roads. From the other figures, the other obstacles can be separated from the whole point cloud.

Every frame in dataset contains about 110,000~120,000 points. The average processing time of our improved algorithm is about 22ms. Thus, our method can process approximately 45 frames per second (45fps). Furthermore, the Velodyne Lidar sensors return only 10 frames per second (10fps). As a result, our processing system works well in real-time as it operates 4.5 times faster than Lidar sensors. Our method is contrasted with some different algorithms in Table 1, traditional method Random Sample Consensus (RANSAC) consumes 262ms on average for one frame point cloud, and Ground Plane Fitting (GPF) approach consumes 150ms on average for one frame. The result shows that our method produces favorable results at higher speed than the previous methods.

Finally, we attempt the down-sample approach to reduce the points in the dataset, and compare it with RANSAC method and GPF algorithm, making use of the similar scenarios, the separation results are similar and our approach performs better. The results are shown in Fig. 13. It is obvious that the computational complexity of our algorithm is smaller than traditional functions while the number of points increases.

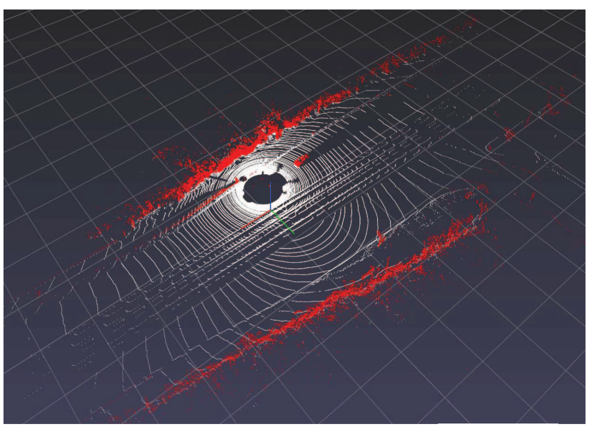

Fig. 7. Result of frame one segmentation in oblique view

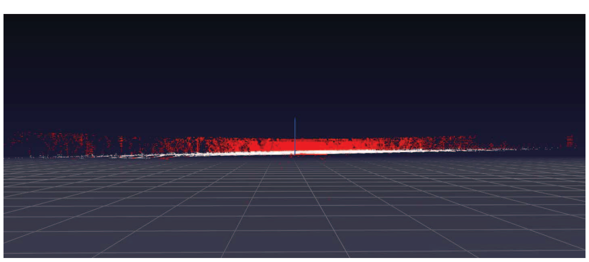

Fig. 8. Result of frame one segmentation in side view

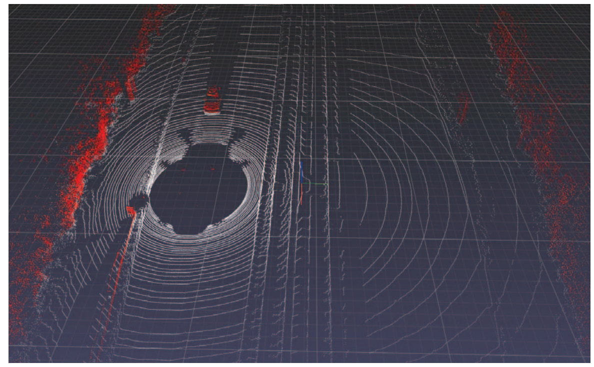

Fig. 9. Result of frame one segmentation in perspective view

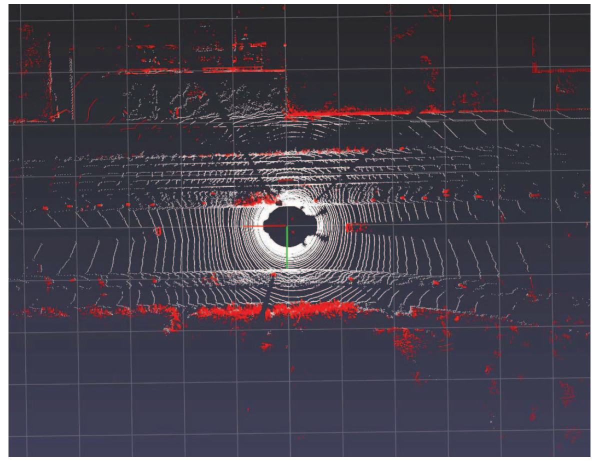

Fig. 10. Result of frame two segmentation in top view

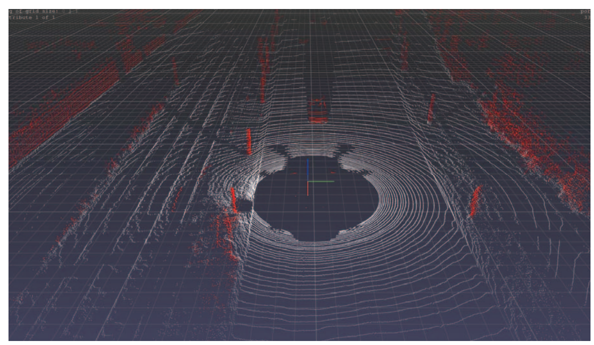

Fig. 11. Result of frame two segmentation in perspective view

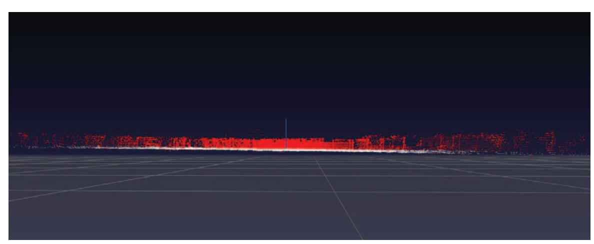

Fig. 12. Result of frame two segmentation in side view

Table 1. Processing time comparison for different method

Method	Average processing time per frame (ms)
RANSAC	262
GPF	150
Our method	22

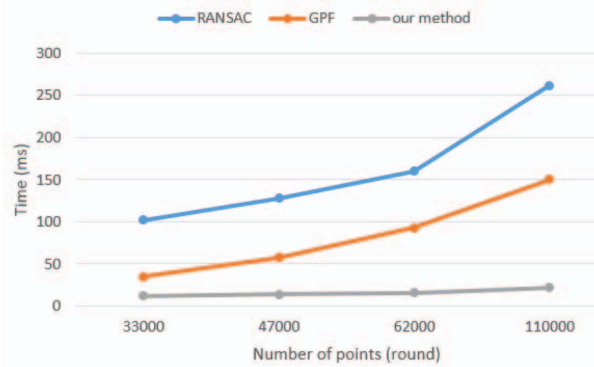

Fig. 13. Processing time comparison for different number of points

4 CONCLUSION

In this paper, an improved fast ground segmentation algorithm for 3D point cloud is proposed, basing on the previous three limitation factors, we add two limits named height threshold and vertical distance into this algorithm. These two conditions improve the effect of the gradient and abnormalities conditions. The algorithm has been successfully tested on KITTI dataset, and our improved algorithm takes just 22ms per frame. the visualization results show that our method has a certain effect, as the time complexity increases, the accuracy improves, and still meets the real-time requirements. The next step is to improve the selection principle of the ground initial points in order to increase the accuracy and reduce the time complexity. For this purpose, we plan to propose more complex segmentation models and furthermore consider a more effective and efficient cluster method.

REFERENCES

[1] Kang, W. M., Moon, S. Y., Park, J. H. An enhanced security framework for home appliances in smart home, Human-centric Computing and Information Sciences, Vol.7, No.1, 6., 2017.

[2] Gubbi, J., Buyya, R., Marusic, S., Palaniswami, M., Internet of Things (IoT): A vision, architectural elements, and future directions, Future generation computer systems, Vol.29, No.7,1645-1660, 2013.

[3] Xiao, B., Wang, Z., Liu, Q., Liu, X., SMK-means: an improved mini batch k-means algorithm based on mapreduce with big data, Comput. Mater. Continua, Vol.56, No.3, 365-379, 2018.

[4] Hernández, J., Marcotegui, B., Point cloud segmentation towards urban ground modeling, 2009 Joint Urban Remote Sensing Event, 1-5, 2009.

[5] Moosmann, F., Pink, O., Stiller, C., Segmentation of 3D lidar data in non-flat urban environments using a local convexity criterion, IEEE Intelligent Vehicles Symposium, 215-220, 2009.

[6] Zermas D., Izzat I., Papanikolopoulos N., Fast segmentation of 3d point clouds: A paradigm on lidar data for autonomous vehicle applications, IEEE International Conference on Robotics and Automation (ICRA), 5067-5073, 2017.

[7] Chu, P., Cho, S., Sim, S., Kwak, K., Cho, K., A Fast Ground Segmentation Method for 3D Point Cloud, JIPS, Vol.13, No.3, 491-499, 2017.

[8] Geiger, A., Lenz, P., Stiller, C., Urtasun, R., Vision meets robotics: The kitti dataset, The International Journal of Robotics Research, Vol.32, No.11, 1231-1237, 2013.

[9] Rusu, R. B., Cousins, S., 3d is here: Point cloud library (pcl), IEEE international conference on robotics and automation, 1-4, 2011.

[10] Douillard, B., Underwood, J., Kuntz, N., Vlaskine, V., Quadros, A., Morton, P., Frenkel, A., On the segmentation of 3D LIDAR point clouds. IEEE International Conference on Robotics and Automation, 2797-2805, 2011.

[11] Wen, M., Cho, S., Chae, J., Sung, Y., Cho, K., Range image-based density-based spatial clustering of application with noise clustering method of three-dimensional point clouds, International Journal of Advanced Robotic Systems, Vol.15, No.2, 1729881418762302, 2018.

[12] Jo, H., Yoon, Y. I., Intelligent smart home energy efficiency model using artificial TensorFlow engine, Human-centric Computing and Information Sciences, Vol.8, No.1, 1-18, 2018.

[13] Xiao, B., Wang, Z., Liu, Q., Liu, X. SMK-means: an improved mini batch k-means algorithm based on mapreduce with big data, Comput. Mater. Continua, Vol.56, No.3, 365-379, 2018.

[14] Guo, H., Liu, F., Yu, R., Sun, Z., Chen, H. Regional path moving horizon tracking controller design for autonomous ground vehicles, Science China Information Sciences, Vol.60, No.1, 013201, 2017.