ICIC Express Letters
Part B: Applications
Volume **14**, Number **9**, September **2023**

ICIC International ©2023 ISSN 2185-2766
pp. **977–984**

EXTRACTION OF LOADED CARGO AND VOLUME CALCULATION PCD FOR CARGO VEHICLES

Eun Sung Choi[1], Ji Yeon Kim[2], Young-Jin Kang[2], Young Joon Jeon[1,*] and Seok Chan Jeong[1,2,3]

[1]Convergence of IT Devices Institute Busan
[2]AI Grand ICT Research Center
[3]Department of e-Business
Dong-Eui University
176 Eomgwang-ro, Busanjin-gu, Busan 47340, Korea
{ desty; 15073; scjeong }@deu.ac.kr; jnsp0907@naver.com
*Corresponding author: biggood@deu.ac.kr

Received January 2023; accepted April 2023

Abstract. *The current cargo measurement requires a lot of time and manpower to perform, but automating measurement makes the process more efficient. LiDAR sensors were used in a previous study to get a 2D data array of appearance of the non-standard cargo such as vehicles. The system calculates the length, the height, the width, and the volume and then transforms 2D data into PCD, and structured 3D model with PCD. However, when the trucks are loaded with cargo, the system cannot generate a PCD for the goods only. Therefore, we propose the loaded cargo extraction and volume calculation of PCD of cargo vehicles. We provide two cargo extraction algorithms and a comparative analysis these algorithms in this paper.*
Keywords: PCD, Loaded cargo extraction, Volume calculation, 3D modeling

1. **Introduction.** Recently, with the rapid increase in global cargo volume and digitalization of ports, genetic algorithm techniques for automating cargo shipment or container placement have been proposed [1]. Genetic algorithm is an algorithm that mimics how an organism finds optimized methods while adapting to a special environment, which allow them to present optimized solution to irregular problems that cannot be clearly defined mathematically. However, since only optimizing the transport method for uniformly sized cargo is performed, there is no study to automatically measure non-standard. This is because the current non-standard cargo measurement requires a lot of time and manpower to be performed by humans, thereby delaying efficient cargo shipment processing.

Prior study has proposed the method of non-standard cargo volume calculation based on LiDAR (Light Detection And Ranging) sensor for cargo loading optimization [2]. The coordinates of the moving non-standard cargo were obtained with 2D LiDAR sensors, and the length, height, width, and volume were automatically measured, and 3D modeling was performed to calculate the volume of the cargo for cargo shipment. However, when a cargo vehicle is loaded with cargo, there is a disadvantage in that the volume of only the cargo cannot be calculated.

Therefore, in this paper, by extending the prior study, we propose a measurement technology that extracts the cargo loaded on the cargo vehicle to calculate volume and perform 3D modeling [3]. The volume of cargo can be automatically measured without unloading cargo, which will increase the efficiency of cargo management in the logistics industry. We present calculation of cargo volume and 3D modeling by extracting only the cargo area from the cargo through the PCD (Point Cloud Data) [4] obtained using

DOI: 10.24507/icicelb.14.09.977

LiDAR sensors [2] and the vehicle license plate recognition system that can identify the vehicle [5] to explore the PCD the empty cargo vehicle required to extract the cargo area.

The structure of this paper is as follows. Section 2 describes the proposed algorithms. Section 3 compares and analyzes the two algorithms used for cargo area extraction, and Section 4 concludes and provides future study plans.

2. The Proposed Method.

The complete flow of extraction of loaded cargo and cargo volume calculation PCD for vehicles is depicted in Figure 1. This study was performed with Python 3.9, where the cargo vehicle's side of PCD was presented in 2D using matplotlib [6] and the cargo area is extracted as a 3D model by Open3D [3].

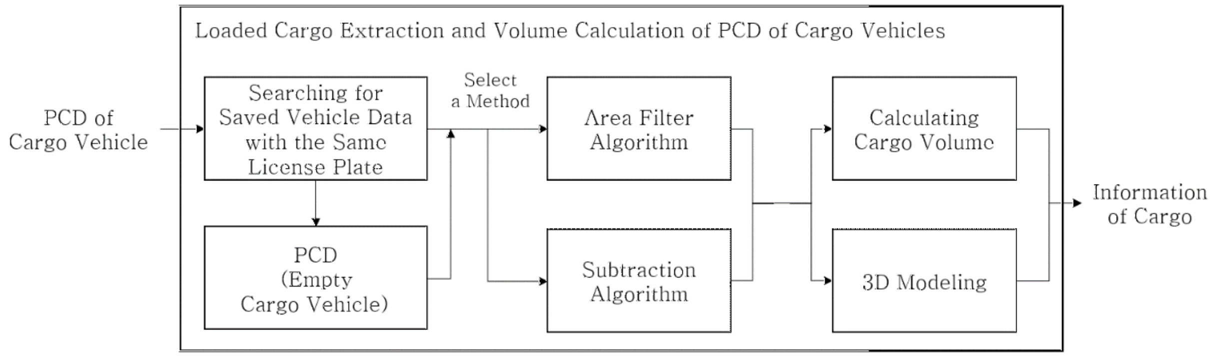


FIGURE 1. Block diagram of the proposed method

2.1. Storing and exploring PCD of empty cargo vehicles.

The PCD of the empty cargo vehicle is necessary to extract a cargo area from the cargo vehicle data constituted of the point cloud. The PCD is calculated by measuring empty cargo trucks in advance and classifying the data using license plate recognition numbers [5]. After that, using the PCD and license plate information of a vehicle laden with goods, data from an empty cargo truck with the same license plate is investigated, and the cargo area is separated using the chosen loaded cargo extraction algorithm.

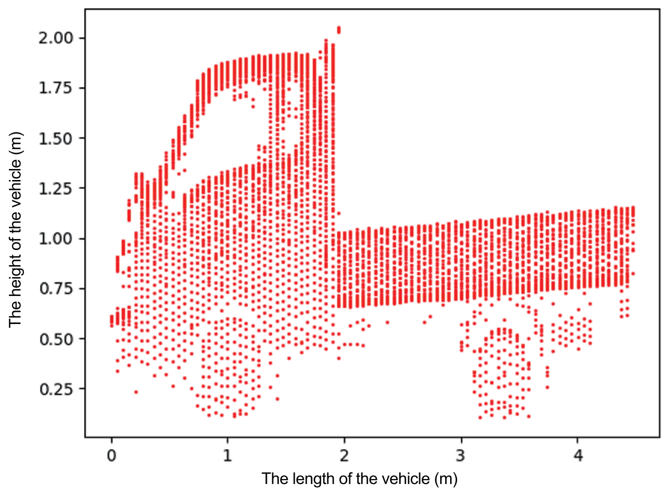


FIGURE 2. Side of the PCD of the cargo vehicle

According to prior study [2], the width of the vehicle was set to x-axis, the height was set to y-axis, and in the length was set to z-axis, and in this study, the y-axis and z-axis were used to represent the side of the vehicle that can specify the cargo area in 2D.

2.2. **Area filter algorithm.** We propose an algorithm for identifying the cargo area of a cargo vehicle to create an area filter and then extracting a cargo area. To do so, it is necessary to search the coordinates of the loader. Since the input PCD is data measured by the LiDAR sensor, there is no data provided except for the coordinates. Therefore, the feature points should be explored with only coordinates.

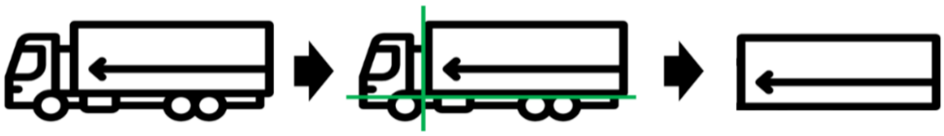
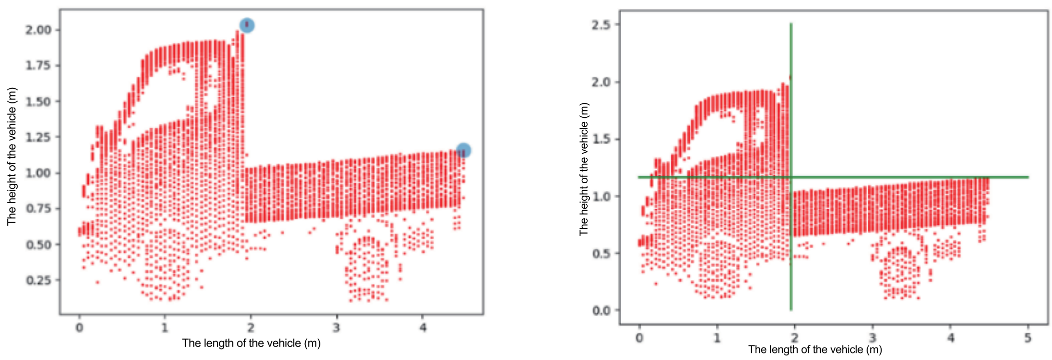


FIGURE 3. Brief description of area filter algorithm

In Figure 4(a), the points that can specify the cargo area are the rightmost point at the top of the PCD and the uppermost point on the right. To search these two points, find the rightmost coordinate among the top coordinates by searching for the coordinate with the maximum value of z, the length coordinate of truck, among the coordinates in the range below 0.03 m from the maximum value of y, the height coordinate of truck, find the uppermost coordinate among the right coordinates by searching for the coordinate with the maximum value of y among the coordinates in the range below 0.03 m from the maximum value of z. At these two points, the algorithm creates two filters parallel to the y-axis and the z-axis. However, Figure 4(b) shows that the intersection of the two filters and the concave vertex of the loader do not match. Since the loader of the cargo vehicle is not necessarily perpendicular, a cargo area filter with an added slope is required and the concave vertex coordinate of the loader has to be explored.

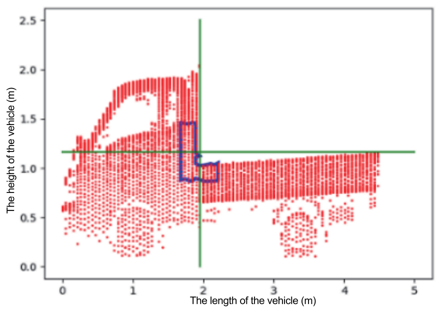


(a) Feature points of the cargo vehicle (b) Area filter without slope

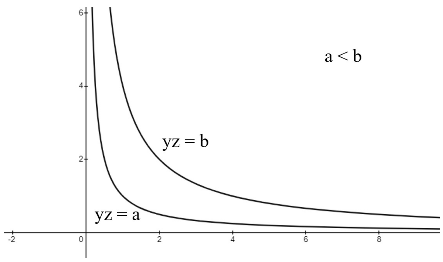
FIGURE 4. Area filter created with feature points

As shown in Figure 5(a), since it can be estimated that the concave vertex of the loader is close to the intersection of the two previously created filters, a rectangle 0.3 m away from the intersection is created to divide the region and apply the concave hull algorithm [7] for region to implement the polygon. Afterwards, a rectangle 0.25 m away from the intersection is created to remove the outer coordinates of polygon, and the y-value and z-value of the remaining coordinates are multiplied to find the coordinates closest to the origin of the minimum value of the coordinate and derive the concave vertex as shown in Figure 5(c). The coordinate closest to the origin can be obtained using Pythagorean theorem, but since calculation requires a lot of computation, the positive part of the hyperbolic function, $yz = k$, is used to determine the coordinates with the smallest k as close to the origin as shown in Figure 5(b). Using Equation (1) of the line passing through two different points is as follows:

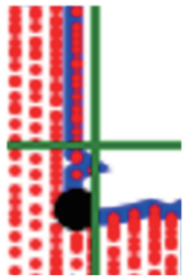
$$y - y_1 = \frac{y_2 - y_1}{x_2 - x_1}(x - x_1) \tag{1}$$



(a) Apply concave hull algorithm to partitioned area



(b) Graph to find a point close to the origin

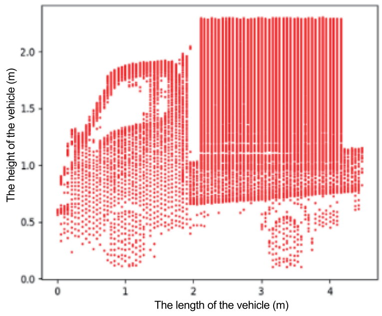


(c) Concave vertex search result

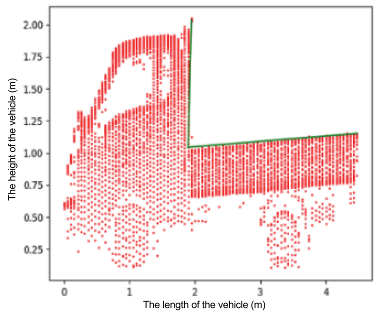
FIGURE 5. Exploring the concave vertex of the cargo area

The equations of each straight line connecting the concave vertex coordinates of the loader and the two previously obtained points was calculated to generate a cargo area filter with slope. If the slope is 0 or infinity, the filter parallel to the axis originally generated can be applied as it is.

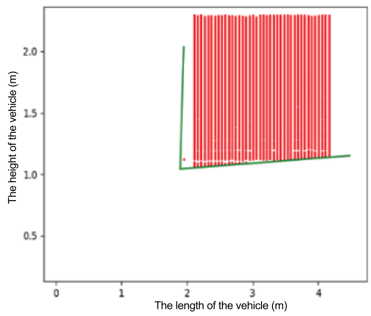
As shown in Figure 6(a), area filter algorithm was applied by loading the simulated cargo data on the cargo vehicle. It was confirmed that the cargo area and the vehicle were separated as shown in Figure 6(b), and the cargo area was extracted as shown in Figure 6(c) by converting the two equations used in the area filter into inequality.



(a) Loading simulated cargo into a cargo vehicle



(b) Create area filter



(c) Cargo area extraction

FIGURE 6. Result of the cargo area extraction with area filter

2.3. **Subtraction algorithm.** We propose an algorithm to extract the cargo area by removing the non-cargo area using subtraction algorithm. The non-cargo area removal algorithm using subtraction algorithm is a way that extracts the cargo area by subtracting the previously stored PCD of an empty cargo vehicle from the PCD of the vehicle loaded with the cargo.

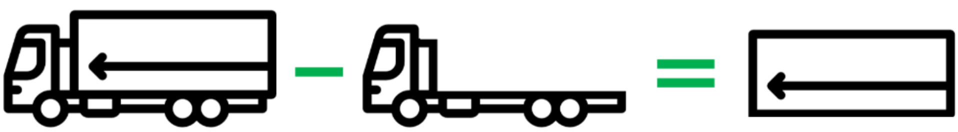
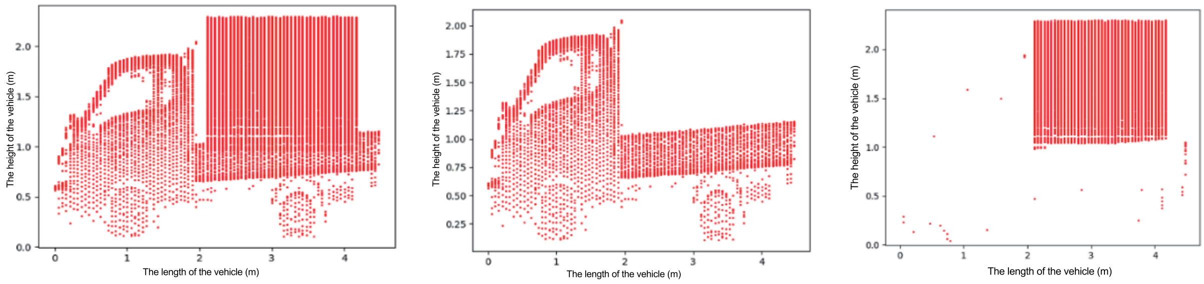


FIGURE 7. Brief description of subtraction algorithm

The PCD of a loaded cargo vehicle and the PCD of an empty cargo vehicle are data from a previous research that measured a moving vehicle using a LiDAR sensor, therefore, the two coordinates do not perfectly match, and the z-axis spacing is different depending on the measurement environment. Since it is impossible to subtract the matching coordinates that match, subtraction has to be performed by area. At first, the PCD coordinates are moved so that the minimum value of the y coordinate and the minimum value of the z coordinate becomes 0, and the two data are aligned according to the origin. Then calculate using the Pythagorean theorem the distance between coordinates of PCD of empty cargo vehicle and coordinates of the PCD of the vehicle loaded with cargo, one by one. The area to be subtracted is determined by the radius of the circle, and if there are coordinates inside the circle delete the coordinate and leave it in the opposite case.

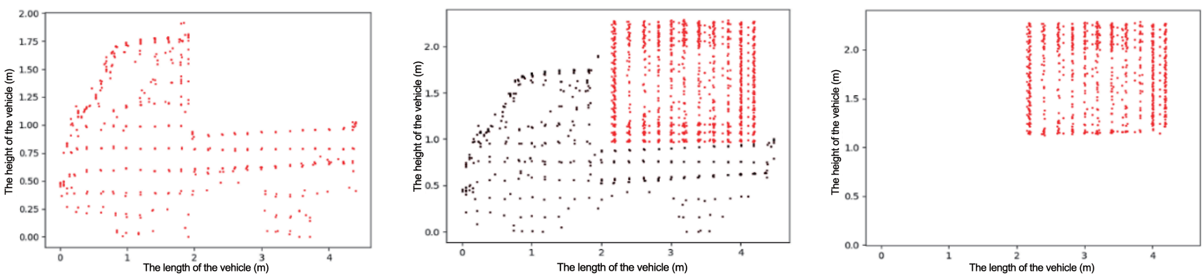
As shown in Figure 8(a), subtraction algorithm was applied by loading simulated cargo data to the cargo vehicle. Figure 8(c) is the result of extracting the cargo area by applying subtraction algorithm to the PCD in Figure 8(a) and Figure 8(b). As a result of applying subtraction algorithm, it took 115.70882 seconds in the processing time. This is because it takes a long time to determine the subtraction area due to a lot of PCD coordinates. In addition, it can be seen that the cargo area could not be completely extracted because the shape of PCD was different even for the same vehicle due to an error according to vehicle measurement. To improve these two problems, down sampling and statistical outlier removal [8] were applied.



(a) Loading simulated cargo into a cargo vehicle (b) Empty cargo vehicle (c) Subtraction

FIGURE 8. Subtraction empty cargo vehicle from vehicle loaded with cargo

First, voxel grid [3] was applied as a way of down-sampling to the PCD of empty cargo vehicles and the PCD of vehicles carrying cargo. Through Figure 8(a) and Figure 9(a), it can be seen that the number of coordinates after down-sampling is significantly decreased compared to before down-sampling. When performing voxel grid, the applied voxel size becomes the radius of the circle to be subtracted by designating the interval of coordinates in the shape of a tetrahedron. When the cargo was extracted by subtraction algorithm using voxel grid, the processing time was 0.37001 seconds, which was significantly reduced from the previous one. Second, to remove nose, a statistical outlier removal was used, the



(a) Empty cargo vehicle down-sampled (b) Loading cargo vehicle down-sampled (c) Subtraction

FIGURE 9. Subtracting PCDs from down-sampled

average distance of 20 neighboring points was calculated, and the coordinates outside the standard deviation of ±0.95 were identified as noise through Gaussian distribution, and the cargo area was neatly extracted as shown in Figure 9(c).

2.4. **Volume calculation.** The width, height, and length are obtained by calculating the difference between the maximum and minimum x-coordinate, y-coordinate, and z-coordinate values of the cargo's PCD. Then, for volume calculation, the PCD is made into a 2D data array by forming coordinates with the same z value into one frame. Equation (2) used to obtain the area of 2D data is the shoelace formula and is as follows:

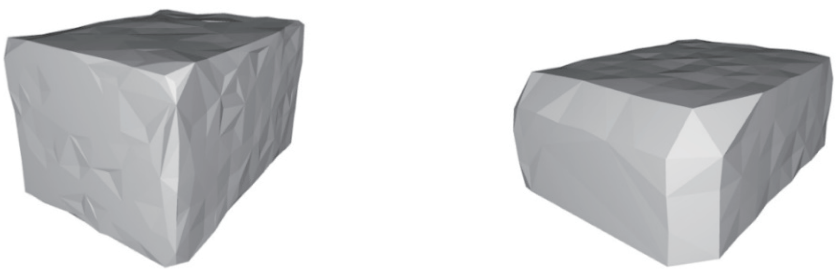
$$A = \frac{1}{2}\left|(x_1y_2 + x_2y_3 + \cdots + x_{n-1}y_n + x_ny_1) - (x_2y_1 + x_3y_2 + \cdots + x_ny_{n-1} + x_1y_n)\right| \quad (2)$$

where n is the number of the coordinates in the frame, and A is the area of 2D data in the frame. The volume is calculated by integrating for length. Since the integration takes a long time to create a function according to the array of area data, we use Riemann sum. Equation (3) is an expression to find the area as the intermediate value of the left Riemann sum and the right Riemann sum to reduce error, and is as follows:

$$\frac{1}{2}\left(\sum_{k=0}^{m-1} A_k + \sum_{k=1}^{m} A_k\right)\frac{l}{m-1} = \left(\frac{A_0 + A_m}{2} + \sum_{k=1}^{m-1} A_k\right)\frac{l}{m-1} \quad (3)$$

m is the number of 2D data in 2D data array, and l is the measured length. If two sigmas are used as the left side of Equation (3), two for-loop also have to be used in computer. In order to reduce processing time of the algorithm, sigmas were reduced to one as shown on the right side of Equation (3).

2.5. **3D modeling.** 3D modeling is performed with the PCD of cargo to visualize the data. Data is down-sampled with voxel grid to reduce processing time. Since the subtraction algorithm already applied voxel grid to PCD, voxel grid is applied only to area filter algorithm. Then, create Triangle Mesh with the Alpha shape [9], and calculate the surface normal vector for each mesh to determine the surface direction of the model. Complete the 3D modeling work by covering the surface according to the normal vector and the result is as shown in Figure 10.



(a) 3D model with area filter (b) 3D model with subtraction

FIGURE 10. Results of 3D modeling with cargo extraction

3. **Comparative Analysis.** A simulated cargo with a length of 2.1 m, a height of 1.3 m, and a width of 1.56 m was loaded on a cargo vehicle, and the cargo area was extracted using the area filter algorithm and the subtraction algorithm. The result of the area filter algorithm was 2.059 m in length, 1.321 m in height, and 1.569 m in width, and the result of the subtraction algorithm was 1.959 m in length, 1.009 m in height, and 1.503 m in width. As a result of comparing the area filter algorithm and the subtraction algorithm as shown in Table 1, the area filter algorithm is better for in down-sampling required, processing

Table 1. Area filter vs subtraction

Content	Area filter	Subtraction
Down-sampling required	Not required	Required
Processing time	0.16957 s 0.01795 s (down-sampled)	115.70882 s 0.37001 s (down-sampled)
Extraction accuracy	Length: 98.05% Height: 98.38% Width: 98.78%	Length: 93.29% Height: 77.62% Width: 96.35%
Generalization	Each type of cargo vehicle needs different algorithms for area filter	Generalizable
Algorithm complexity	Complex	Simple

time, and extraction accuracy, but the subtraction algorithm is better for generalization and algorithm complexity.

Since the subtraction algorithm has to compare each coordinate, the more coordinates there are, the longer the processing time is square. Down-sampling is required to reduce time, but the area filter algorithm creates an area filter by identifying only feature points to remove non-cargo area, so there is no need for down-sampling due to short processing time. The subtraction algorithm has data loss at the boundary between the cargo vehicle and the cargo due to voxel grid and area subtraction, but the area filter algorithm extracts the area by drawing straight lines with the feature points, thereby reducing the data loss a lot.

However, if a car is loaded on the roof or a trailer is added behind the vehicle, the algorithm for creating the area filter is different, so the existing area filter cannot extract cargo. Since the subtraction algorithm simply subtracts the empty vehicle from the vehicle loaded with cargo, it can be extracted regardless of the type of cargo vehicle, and the algorithm structure is simpler than the area filter algorithm with complex algorithms, making it easier to maintain and less errors.

4. **Conclusions.** This paper implements loaded cargo extraction and volume calculation of PCD of cargo vehicles. Cargo area extraction stores PCD of empty cargo vehicles measured by LiDAR sensors in advance and distinguishes the PCD of the same cargo vehicle by license plate recognition. After that, if an empty cargo vehicle PCD with the same license plate as the PCD of the vehicle loaded with cargo is stored, the cargo area is extracted by applying the area filter algorithm or the subtraction algorithm. The area filter algorithm finds the feature points and the concave vertex of the empty cargo vehicles and creates area filters to extract the cargo. In the subtraction algorithm, a vehicle loaded with cargo and an empty cargo vehicle are down-sampled, and then an overlapping area is determined and subtracted, and the cargo area is extracted by removing noise through a statistical outlier removal. The width, height, and length of the cargo are calculated from the extracted PCD of cargo. After calculating the PCD is transformed into a 2D data array, and the area of 2D data is calculated using the shoelace formula, the volume is obtained by applying the Riemann sum of the area to the length. Finally, a 3D model was created through Triangle Mesh and surface normal vectors and in the case of the area filter algorithm, this was performed after down-sampling by voxel grid.

We plan to verify the validity of implemented algorithms of loaded cargo extraction and volume calculation of PCD of cargo vehicles to the PCD of the cargo vehicle loading the actual cargo rather than randomly generated simulated cargo.

In the next study, the implemented algorithm will be applied to the vehicle PCD loading the actual cargo rather than randomly generated simulated cargo to verify the validity of the cargo area extraction and volume calculation method algorithm, and the algorithm will be supplemented with reference to the verification result, and the algorithms will be supplemented by referring to the verification results.

Acknowledgment. This research was financially supported by Ministry of Trade, Industry and Energy of the Republic of Korea and Korea Institute for Advancement of Technology (KIAT) through the Regional Innovation Cluster R&D Program (Grant Number: P0015330). And it also supported by the MSIT (Ministry of Science and ICT), Korea, under the Grand Information Technology Research Center support program (IITP-2022-2020-0-01791) supervised by the IITP (Institute for Information & communications Technology Planning & Evaluation).

REFERENCES

[1] J. Y. Kim, W. J. Cho, Y. J. Jeon and S. C. Jeong, A design of cargo optimal arrangement algorithm for shipment planning, *Proc. of the Korea Multimedia Society Conference*, vol.24, no.2, pp.41-42, 2021.

[2] Y. J. Jeon, Y. S. Kim, S. K. Ahn and S. C. Jeong, A study on the method of non-standard cargo volume calculation based on LiDAR sensor for cargo loading optimization, *Journal of Korea Multimedia Society*, vol.25, no.4, pp.559-567, 2022.

[3] Q.-Y. Zhou, J. Park and V. Koltun, Open3D: A modern library for 3D data processing, *arXiv.org*, arXiv: 1801.09847, 2018.

[4] D. Holz, A. E. Ichim, F. Tombari, R. B. Rusu and S. Behnke, Registration with the point cloud library: A modular framework for aligning in 3-D, *IEEE Robotics and Automation Magazine*, vol.22, no.4, 2015.

[5] S. H. Ha, S. C. Jeong, Y. J. Jeon and M. S. Jang, A study on vehicle license plate recognition system through fake license plate generator in YOLOv5, *Journal of the Korean Society of Industry Convergence*, vol.24, no.6, pp.699-706, 2021.

[6] P. Barrett, J. Hunter, J. T. Miller, J.-C. Hsu and P. Greenfield, Matplotlib – A portable python plotting package, *ASP Conference Series*, vol.347, pp.91-95, 2005.

[7] A. Moreira and M. Y. Santos, Concave hull: A k-nearest neighbours approach for the computation of the region occupied by a set of point, *Proc. of International Conference on Computer Graphics Theory and Applications*, Barcelona, Spain, pp.61-68, 2007.

[8] H. Balta, J. Velagic, W. Bosschaerts, G. De Cubber and B. Siciliano, Fast statistical outlier removal based method for large 3D point clouds of outdoor environments, *IFAC-PapersOnLine*, vol.51, pp.348-353, 2018.

[9] S. Park and S. Lee, A surface reconstruction algorithm using weighted alpha shapes, *Korean Journal of Society for Computational Design and Enginerring*, pp.267-275, 2005.