

LiDAR 센서기반 근접물체 탐지계측 알고리즘

Algorithm on Detection and Measurement for Proximity Object based on the LiDAR Sensor

정 종택¹ · 최 조천^{2*}

¹카네비컴(주)

²목포해양대학교 해양컴퓨터공학과

Jong-teak Jeong¹ · Jo-cheon Choi^{2*}

¹Carnavicom.Co.,Ltd., Incheon, 21984, Korea

²Department of Marine Computer Engineering, National Mokpo Maritime University, Jeollanam-do, 58628, Korea

[요 약]

최근 운송수단의 안전운행 및 사고방지를 목표로 하는 자율운행 관련 기술이 적극적으로 연구되고 있다. 현재 자율운행에서 장애물 탐지를 위하여 레이더 및 카메라 기술이 사용되고 있으나, 근접한 물체의 탐지 및 이격거리의 정밀계측에는 LiDAR (light detection and ranging) 센서를 사용하는 방법이 가장 적합하다. LiDAR 센서는 레이저 펄스빔을 발사하고 물체로부터 반사되어 온 반사빔과의 시간차를 취득하여 이것으로 정밀한 거리를 계산하는 측정기로, 광을 이용하기 때문에 대기환경에서 물체의 인식이 감소할 수 있는 단점이 있다. 본 논문은 LiDAR 센서의 raw 데이터에 대한 신뢰성 향상과 이를 기반으로 실시간 주변물체에 대한 탐지 및 이격거리 계측에서 오차를 개선하기 위하여 삼각함수에 의한 포인트 cloud를 추출하고, 선형회귀 모델을 이용하여 계측 알고리즘을 구현하였으며, Python 라이브러리를 활용하여 물체탐지의 오차범위를 개선할 수 있음을 검증하였다.

[Abstract]

Recently, the technologies related to autonomous drive has studying the goal for safe operation and prevent accidents of vehicles. There is radar and camera technologies has used to detect obstacles in these autonomous vehicle research. Now a day, the method for using LiDAR sensor has considering to detect nearby objects and accurately measure the separation distance in the autonomous navigation. It is calculates the distance by recognizing the time differences between the reflected beams and it allows precise distance measurements. But it also has the disadvantage that the recognition rate of object in the atmospheric environment can be reduced. In this paper, point cloud data by triangular functions and Line Regression model are used to implement measurement algorithm, that has improved detecting objects in real time and reduce the error of measuring separation distances based on improved reliability of raw data from LiDAR sensor. It has verified that the range of object detection errors can be improved by using the Python imaging library.

Key word : LiDAR sensor, Line regression model, Distance measurement algorithm, Object detection algorithm, Raw data.

<https://doi.org/10.12673/jant.2020.24.3.192>



This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Received 3 June 2020; Revised 4 June 2020

Accepted (Publication) 27 June 2020 (30 June 2020)

*Corresponding Author; Dong-soon Hong

Tel: +82-2-2209-3671

E-mail: biscoprop@naver.com

I. 서론

최근 4차 산업혁명의 추진과 함께 자율주행자동차 분야에서는 운송 패러다임의 변화가 가속되고 있으며, 해상분야에서도 자율운항선박 및 무인선 (USV; unmanned surface vessel) 등의 기술개발이 본격적으로 이루어지고 있다. 이러한 자율화, 무인화를 위해서 가장 먼저 선행되어야 할 요소는 주변물체의 탐지 및 모니터링 기술이라고 말할 수 있다. 해상에서는 기존에 주변환경의 인식을 위하여 레이더와 카메라 기술이 사용되고 있으나, 자율운항 및 USV의 기본적 요소기술은 근접거리의 물체 탐지, 이격거리 및 방위에 대한 정밀한 정보수집이 우선적으로 요구되고 있고, 이러한 문제를 완전하게 해결하는 방법으로 LiDAR 센서의 활용이 고려되고 있다[1]. LiDAR 센서는 레이저 빔을 펄스형태로 발사하고 발사된 빔이 물체에서 반사되어 되돌아오는 반사빔과의 시간차를 인지하여 거리를 계산하는 센서로서 그 원리는 그림 1과 같다. 본 논문에서는 LiDAR 센서를 기반으로 실시간 raw 데이터를 분석하여 2차원 공간으로 표시하며, 각 점들의 위치데이터 행렬을 활용하여 선형회귀 모델을 추출한 후, 주변물체를 탐지하여 최소의 이격거리를 측정하는 알고리즘을 설계하였다.

II. 물체탐지 알고리즘

LiDAR 센서로 측정된 데이터에서 객체를 구분하기 위해서는 불필요한 데이터를 제거하고, 측정데이터의 연속성을 정확하게 판단해야 한다. 우선 LiDAR 센서를 통하여 취득된 3D 포인트 cloud는 데이터의 양이 너무 많으므로 실시간 처리 및 객체분류 처리에는 부적합하다. 이러한 문제를 보완하기 위하여 그림 2와 같이 3D로 취득된 거리, 각도 등의 데이터에서 각 점들의 X, Y 좌표가 포함된 2D 데이터로 변환하는 방법으로 고속 신호처리를 가능하게 하였다.

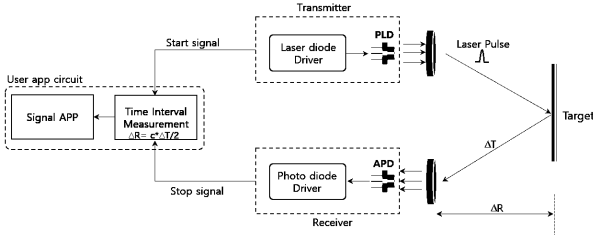


그림 1. 라이다 센서의 원리
Fig. 1. Principle of LiDAR sensor

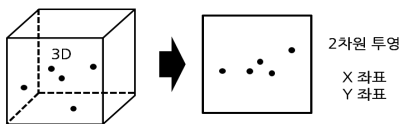


그림 2. 3D 데이터에서 2D 데이터로 변환
Fig. 2. Conversion to 2D data from 3D data

특히, 3D로 입력된 거리, 각도 등의 데이터를 직접 처리하려는 경우, LiDAR 센서에 사용되는 임베디드의 성능대비 raw 데이터에 대한 분석과 판단의 처리과정에서 실시간 보장은 어렵게 된다. 따라서 이를 개선하기 위하여 삼각함수에 의한 포인트 cloud 사이의 거리분류 및 해당 계측부분의 연속성을 판단하는 알고리즘으로 문제를 정립하였다[2],[3]. 본 연구에 적용한 3채널 LiDAR 센서는 120°의 HFOV(horizontal field of view) 범위에서 채널당 750개의 포인트 cloud를 계측할 수 있으며, 포인트 cloud 사이의 각도는 0.16°로 식(1)을 이용하여 거리와 각도에서 포인트 cloud 사이의 거리를 계측하고 구분하였다.

$$D_{ij} = \sqrt{(D_j \times \sin(\angle_{ij}))^2 + (D_i - D_j \times \cos(\angle_{ij}))^2} \quad (1)$$

D_i = Distance from LiDAR to Point i
 D_{ij} = Distance from Point i to Point j
 \angle_{ij} = Angle between Point i, j

계산된 포인트 cloud의 거리가 일정거리 내에서 존재할 경우, 연속성을 가진다고 판단하여 연속된 포인트 cloud의 행렬에 추가하였다. 또한 연속성의 판단에서 LiDAR 센싱 데이터의 노이즈에 의한 영향을 고려하여 특정 횟수까지는 2개 포인트 cloud 사이의 거리가 조건을 만족하지 않으면, 연속성 판단에서 지연을 적용하여 이전 포인트 cloud와 지연된 포인트 cloud의 거리를 비교하였다. 이러한 과정을 거쳐 작성된 연속 포인트 cloud의 행렬로부터 복수의 행렬이 존재할 경우, 물체탐지의 목적에 따라 연속된 행렬에서 노이즈를 최소화하였고, 연속성을 검증하기 위하여 선형회귀 모델을 이용하였다.

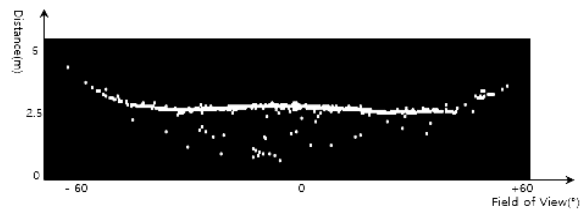


그림 3. 선형잡음 모델
Fig. 3. Model of line noise

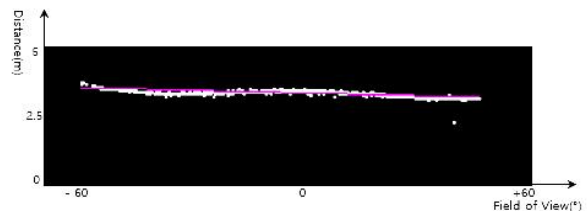


그림 4. 선형회귀 모델
Fig. 4. Model of line regression

그림 3에서 실제로 계측된 LiDAR 센서의 데이터에는 산발적인 노이즈가 많이 포함됨을 알 수 있다. 여기에서 python의 pygame 라이브러리를 이용하여 노이즈가 포함된 데이터에서 연속성을 판단하고, 선형회귀를 적용하여 시각화한 결과가 그림 4 이며, 이와 같이 연속된 포인트 행렬을 추출하므로써 선형회귀 모델을 확인하고 검증하였다. 선형회귀를 구하기 위해서는 데이터군의 각 데이터에서 오차가 최소가 되는 선이 필요하게 되며, 선형회귀의 선은 $y = ax + b$ 와 같은 선형함수를 기반으로 각각 특정한 데이터의 X, Y 좌표들을 고려하여 연립방정식을 얻을 수 있고, 이 연립방정식을 행렬로 표현하여 $Ax = b$ 를 최소화하는 x 값의 산출이 가능하게 된다.

$$Ax \cong b \text{ where } A = \begin{bmatrix} 50 & 1 \\ 100 & 1 \\ 200 & 1 \\ \dots & \dots \end{bmatrix}, x = \begin{bmatrix} x_1 \\ x_0 \end{bmatrix}, b = \begin{bmatrix} 10 \\ 20 \\ 40 \\ \dots \end{bmatrix} \quad (2)$$

식(2)에서 직교행렬(orthogonal matrix)과 상삼각행렬(upper triangular matrix)의 성질을 이용하면, $Q^T A = R$ 과 같은 식을 얻을 수 있으므로 이를 만족하는 Q를 구하면, $A = QR$ 으로 되는 Q와 R을 산출할 수 있다.

$$Q^T A = R \rightarrow QQ^T A = QR \rightarrow A = QR \quad (3)$$

$$Ax \cong b \rightarrow QRx \cong b \rightarrow Rx \cong Q^T b$$

단: Q = Orthogonal Matrix
R = Upper Triangle Matrix

여기에 householder 변환법을 이용하여 Q와 R을 산출하고, householder 변환에서 반복적으로 직교행렬과 상삼각행렬의 성질을 이용하면, 식(3)과 같이 $A = QR$ 이 되는 Q와 R을 구할 수 있으므로 이 과정은 python을 사용하여 선형회귀 모델로 구현하였다[4],[5].

III. 이격거리 계측알고리즘

LiDAR 센서의 ToF(time of flight)를 이용한 거리측정 원리는 레이저의 start 펄스후 echo 펄스 도달까지의 주행시간을 측정하여 거리를 계산하는 원리이다. 이것은 LiDAR 센서에서 광의 경로에 대하여 편차가 없는 일정수준이 유지될 때, 이상적인 거리측정의 조건이 된다.

$$L(m) = \frac{c \cdot \Delta t}{2} = \frac{3 \times 10^8 m/s \cdot \Delta t}{2} \quad (4)$$

그러나 H/W의 온도 및 노화 현상에 따라 변동될 수 있으며, 이러한 오차는 정기적으로 교정하면서 사용할 수도 있지만, 센서의 설계단계에서 최소화시키는 것이 중요하다. 또한, 대기밀

도에 따라 레이저의 주행시간이 달라지므로 이것도 오차의 원인이 될 수 있다. 즉, 대기의 압력, 습도 및 온도에 따라 다르게 나타날 수 있으나, 본 연구에서는 극히 짧은 거리의 측정이므로 대기에 대한 오차는 충분히 무시할 수 있으므로 이에 대한 문제는 고려하지 않았다. 그리고 물표의 매질이나 물표와 수신부의 거리에 따라 반사되는 펄스의 크기는 작아지므로 threshold 점에 의하여 거리측정의 오차가 발생하는데 이것을 walk 에러라고 한다. 이러한 walk 에러를 최소화시키려는 방법으로 CFD(constant fraction discriminator)를 설계하여 적용하였다.

그림 5는 CFD의 기본동작을 나타낸 것으로 그림 5의 a)와 같이 입력펄스와 반사펄스는 threshold 선에서 2개의 교차점을 갖는 일정분율 신호로 된다. 일정분율 신호에서 교차점은 입력신호의 크기에 거의 독립적이기 때문에 그림 5의 b)와 같은 방법으로 교차점을 검출하게 되면 물표의 매질이나 거리에 따라 달라진 반사펄스의 크기 변화에 기인하는 거리오차를 최소화할 수 있다[6]. 그림 5의 c)는 CFD의 회로도이며, 광대역 필터에서 지연을 발생시키고, 입력력 저항으로 감쇄정도의 조절과 파형의 분석에 의하여 walk 에러를 최소화할 수 있도록 설계하였다.

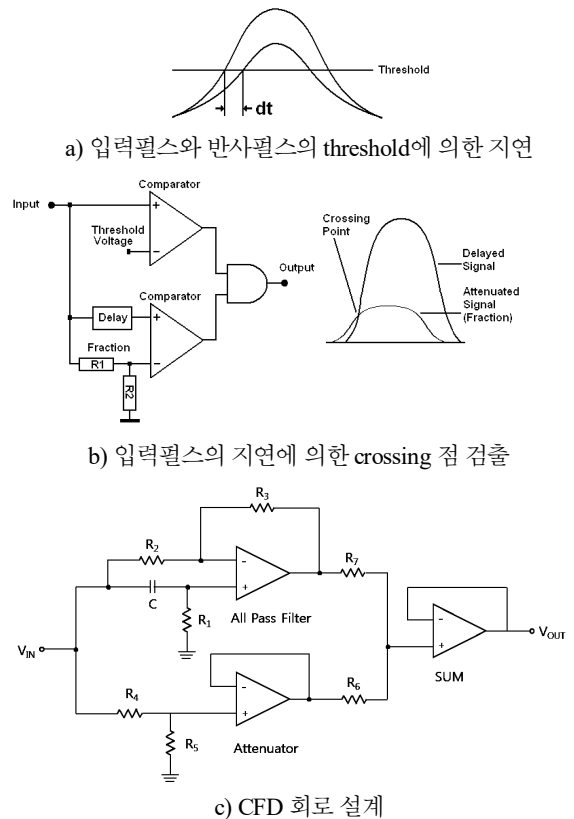


그림 5. CFD의 기본동작 및 설계
Fig. 5. Basic operation and circuit design of CFD

다음으로 레이저의 ToF에 대한 데이터를 얻기 위하여 TDC(time to digital converter)를 사용하며, 레이저 빔은 1ns 에서 왕복 150mm, 100ps 에서 왕복 15mm를 주행하므로 이러한 초고

속 측정을 위해서는 ps의 시간을 측정하는 기술이 필요하다. 즉, 거리분해능의 향상을 위하여 25 ps 시간단위의 고정도 측정이 가능한 TI 제품의 TDC7200 모듈을 사용하였다. 그림 6은 TDC 모듈의 블록도이며, 그림 7은 레이저의 주행시간을 산출하는 프로그램의 순서도이다[7],[8].

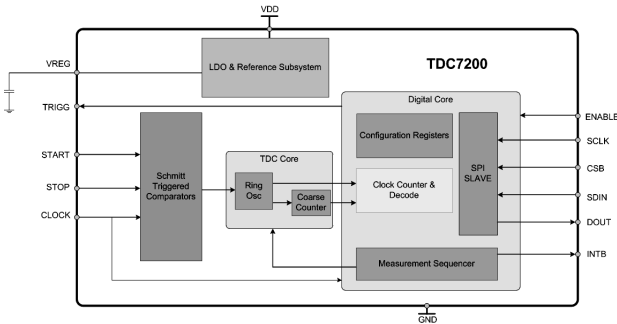


그림 6. TDC 모듈의 블록다이어그램
Fig. 6. Block diagram of TDC module

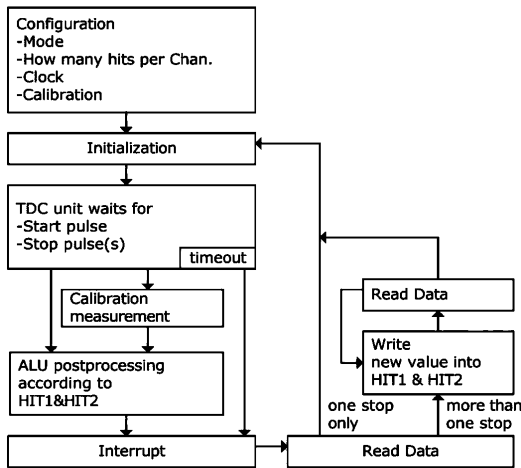


그림 7. TDC 시간측정 순서도
Fig. 7. TDC time measuring flowchart

TDC 측정은 start 와 stop 펄스의 적용까지 op code로 초기화되면서 start 와 stop 펄스를 대기한다. 처리부는 시간데이터를 준비하며, stop 펄스가 도착하면 인터럽트를 발생한다. 인터럽트의 발생여부에 따라 TDC 데이터를 읽고, 시간차를 계산하여 거리값을 얻는다. TDC7200 모듈의 분해능은 55 ps 이며, 표준 편차는 35 ps, 측정범위는 mode 1 에서 12~500 ns, mode 2 에서 250 ns~8 ms 의 특성을 갖는다. 본 연구는 mode 2 에서 고정도를 위하여 stop 펄스의 계수를 ps 단위로 취득하도록 설계하였다. LiDAR 센서에서 raw 데이터를 취득한 후, 공간좌표 상에서 이격거리를 측정하기 위해서는 각도 분해능의 산출과 좌표계의 변환이 필요하다. 센서에서 취득된 거리와 방향데이터는 구면좌표계로 표현되므로 이러한 3차원의 데이터를 가지화하고 이격거리 측정을 위하여 식(5)의 직교좌표계로 변환하였다.

각도 분해능은 최대의 수평시야각에 대하여 송수신된 레이

저 펄스의 갯수로 나누어 계산되고, 포인트 cloud 의 θ_i 는 각도 분해능의 합으로 얻을 수 있다.

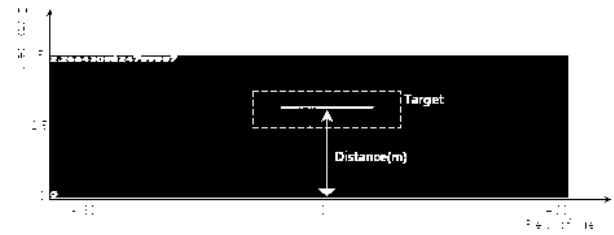


그림 8. 실시간 계측
Fig. 8. Real time measurement

$$x = r \cos(\theta), y = r \sin(\theta), \theta_i = \sum_{i=1}^N \Delta\theta_i \quad (5)$$

식(5)를 기반으로 LiDAR 센서로부터 물체와의 이격거리 계측을 위하여 II장에서 언급한 선형회귀를 통해 얻어진 선형회귀 모델과 연속된 포인트 cloud 행렬을 비교하여 노이즈를 경감시키면, 선형회귀 모델과 일정오차 이내의 연속성의 행렬이 얻을 수 있었다. 그리고 노이즈의 제거로 선형의 행렬에서 계측점과 LiDAR 센서와 거리가 최소치로 측정되는 포인트 cloud를 산출하고, 그 점과의 거리를 계산하도록 구현하면 그림 8과 같이 LiDAR 센서와 물체와의 이격거리를 실시간 계측할 수 있다.

IV. 실험 및 분석

설계된 알고리즘의 검증을 위하여 표 1과 같은 실험환경을 구성하여 이것을 기반으로 측정실험을 하였다. 측정데이터는 멀티프로세싱 기법을 사용하여 취득하였고, 그림 9는 각각의 처리기 사이에서 데이터전송의 내용을 도식화 한 것이다.

각각의 처리기는 개별적인 함수에 의하여 운용되도록 설계하였고, 처리기 간에 서로 참조해야 하는 데이터는 queue로 구성하여 실시간 교환이 가능하도록 하였다. LiDAR 센서 데이터와 메인 PC 명령코드의 교환에는 2개의 전용 queue를 설계하였고, LiDAR 센서 데이터의 수신, 객체인식 프로세스 그리고 메인 PC 명령통신 등의 처리기에는 데이터를 실시간으로 제공할 수 있는 각각의 queue를 설계하였다[9].

표 1. 실험환경
Table 1. Experimental environment

Program / Library	Version	Build
Python	3.5.6	-
Pygame	1.9.4	pip
Numpy	1.15.2	py35ha559c80_0
Anaconda	4.2.0	np111py35_0
mkl	2018.0.3	1

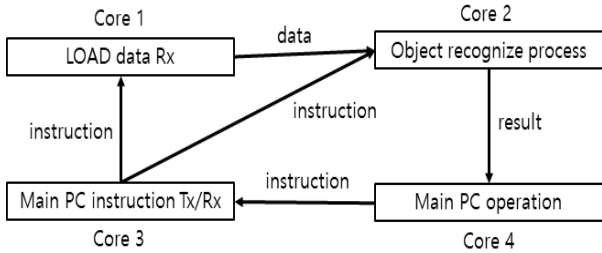


그림 9. 멀티프로세싱 프로그램 구조도
Fig. 9. Multiprocessing program structure

LiDAR 센서의 raw 데이터에서 보다 정확한 거리정보를 얻기 위해서는 TDC의 기본오차와 신호처리 H/W의 지연을 고려한 LiDAR 센서의 교정이 필요하다. LiDAR 센서는 레이저를 이용하여 거리를 측정하므로 광속은 일정하다는 가정하에 신호처리 H/W에서 발생하는 지연과 TDC의 ToF 시간을 더하여 실제 거리와 비교하는 교정실험으로 진행하였다. 본 연구에서 제작한 신호처리 H/W와 TDC의 정밀도에서는 mm 단위의 보정으로는 신뢰성을 보장할 수 없으므로 cm 단위로 보정실험을 진행하였다. 거리측정의 실험환경에서 교정작업을 위하여 그림10과 같은 거리측정 장치를 사용하였고, 교정방법은 1~7m까지의 거리에서 0.1m 간격으로 기준 타겟을 이동시키며, LiDAR 센서로 측정된 거리와 실제 거리를 비교하며 이것으로 보정하여 기준 데이터를 얻었다.

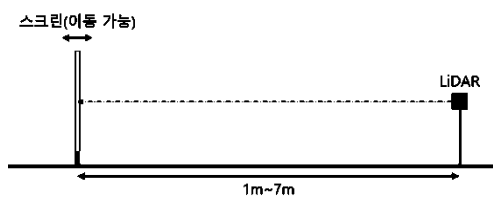


그림 10. LiDAR 센서의 교정 실험
Fig. 10. Calibration experiment of LiDAR sensor

다채널 LiDAR 센서는 채널별로 각각의 보정이 필요하므로 채널당 3번씩 반복 교정으로 정확도를 향상시켰으며, 수행한 결과로 그림 11과 같이 2채널 모두 직선으로 정밀한 특성을 유지하는 교정치를 얻었다. 그림 12는 교정실험으로 물체를 인식하고, 연속된 포인트 cloud 행렬로 얻어진 선형회귀 모델을 확인하는 결과를 나타낸 것이다. LiDAR 센서를 고정시킨 뒤, 실시간으로 확인할 수 있는 뷰어를 활용하여 LiDAR 센서에서 특정물체를 탐지하고, 계측된 최소거리를 측정하여 ruller 와 실제 거리를 비교하였다. 여기에서 포인트 cloud 들의 선형성 향상은 물론 노이즈도 저감됨을 확인하였다. 표 2는 실험을 통하여 LiDAR 센서의 측정치와 실제로 설정된 거리를 비교한 결과를 보인 것이다. 실험결과와 분석에서 LiDAR 센서로 계측된 거리는 실제거리와 비교하여 오차가 크게 개선됨을 확인하였고, 동시에 연구한 알고리즘의 실효성도 검증되었다고 하겠다.

또한, 광속은 일정하고, 실험에서 구현한 H/W의 지연이 cm

단위 이내에서 일정하게 나타나므로 30m, 50m에서 측정하더라도 동일한 cm 단위의 오차를 가지게 됨을 알 수 있다.

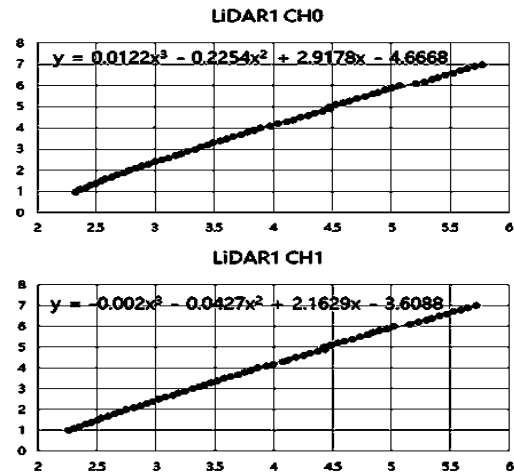


그림 11. 2채널 각각의 교정 결과
Fig. 11. Calibration result of 2 channel respectively

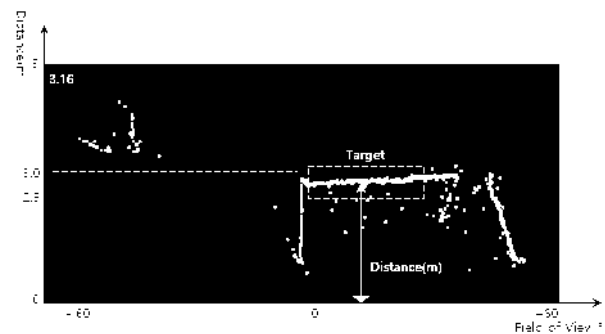


그림 12. LiDAR 센서의 계측
Fig. 12. Measurement of LiDAR sensor

표 2. 거리계측의 비교(평균)

Table 2. Comparison of distance measurement

Measurement Method	Average Distance (m)/3회
LiDAR	3.16
Ruller	3.23

V. 결론

정밀한 근거리계측이 가능한 LiDAR 센서는 자율주행자동차, 주행로봇 등 많은 분야에 적용되고 있다. 그러나 해상의 자율운항 선박에 대한 연구에서 근거리 탐지 충돌예방 센서는 매우 중요한 요구사항이지만, 해상환경의 열악한 조건으로 인하

여 이에 대한 적당한 해법은 아직도 미비한 상태이다,

LiDAR 센서는 레이저 빔을 이용하는 장치이므로 기상조건과 이동체의 동요환경에 대하여 많은 제약을 받는다. 본 연구에서는 LiDAR 센서의 이러한 단점을 보완하기 위하여 raw 데이터의 신뢰도를 향상시키고자 선형회귀 모델을 적용한 물체탐지 및 이격거리 계측알고리즘을 구현하였고, 실험을 통하여 물체탐지 및 거리계측의 오차가 개선됨을 실험적으로 검증하였다. 연구결과를 기반으로 해상 of 자율운항선박, USV 등의 분야에서 자선 주변의 회피가 가능한 근거리에 대한 물체탐지와 거리계측 센서로 적용해도 충분히 가능할 것으로 사료된다.

부가적으로 해상의 선박항해 조건에서 선박동요 환경을 보상해 주는 수평제어 마운트의 기술을 접목하고, 기상 to 따른 외란광(태양광)에 의한 레이저의 감쇠와 노이즈 보완 등을 병행하면 더욱 신뢰도를 향상시키는 시스템이 될 것으로 판단된다.

References

- [1] H. S. Park and H. R. Park, "A study on the policy directions related to the introduction of smart maritime autonomous surface ship (MASS)," in *Conference Proceedings of Navigation and Port Research(19)*, Busan: Korea, pp. 234-235, 2019
- [2] S. P. Choi, J. H. Jo, and J. S. Kim, "An filtering automatic technique of LiDAR data by multiple linear regression analysis," *Journal of the Korean Society for Geospatial Information Science*, Vol. 19, No. 4, pp. 109-118. 2011
- [3] F. Castaño, G. Beruvides, A. Villalonga, and RE. Haber, "Self-tuning method for increased obstacle detection reliability based on internet of things LiDAR sensor models," *Sensors Journal of Multidisciplinary Digital Publishing Institute*, Vol. 18, No. 5, pp. 1508-1523, May 2018.
- [4] A. Ferraz, S. Saatchi, C. Mallet, and V. Meyer, "Lidar detection of individual tree size in tropical forests," *Journal of Remote Sensing of Environment*, Vol. 183, pp. 318-333. Sep. 2016.
- [5] J. Wijesingha, T. Moeckel, F. Hensgen, and M. Wachendorf, "Evaluation of 3D point cloud based models for the prediction of grassland biomass," *International Journal of Applied Earth Observation and Geoinformation*, Vol. 78, pp. 352-359. June 2019.
- [6] Sense Light, 2-Channel. USB-Controlled Constant Fraction Discriminator [Internet]. Available : http://www.sensl.com/downloads/ds/DS-110522_CFD_v1p1.pdf.
- [7] Texas Instrument. TDC7200 Time-to-Digital Converter for Time-of-Flight Applications in LIDAR [Internet]. Available : <http://www.ti.com/lit/ds/symlink/tdc7200.pdf>.
- [8] J.-P. Jansson, A. Mantyniemi, and J. Kostamovaara, "A CMOS time-to-digital converter with better than 10 ps single-shot precision," *IEEE Journal of Solid-State Circuits*, Vol. 41, No. 6, pp. 1286-1296, May 2006.
- [9] J. J. Yun, An implementation of LiDAR sensor supporting autonomous navigation of unmanned vehicles, Ph. D. dissertation, Mokpo National Maritime University, Korea, Feb. 2017.



정 종택 (Jong-Teak Jeong)

2014년 2월 : 한국해양대학교 경제산업학부 (학사)

2016년 2월 : 목포해양대학교 대학원 해양전자통신컴퓨터공학과 (공학석사)

2020년 현재 : 목포해양대학교 대학원 해양전자통신컴퓨터공학과 (박사과정), 카네비컴(주) 대표

※관심분야 : 자율주행, 웨이브통신, 해양전자통신



최 조 천 (Jo-Cheon Choi)

1987년 2월 : 서울과학기술대학교 전자공학과 (공학사)

1990년 2월 : 조선대학교 대학원 컴퓨터공학과 (공학석사)

1998년 8월 : 한국해양대학교 대학교 전자통신공학과 (공학박사)

2020년 현재 : 목포해양대학교 해양컴퓨터공학과 교수

※관심분야 : 해양전자통신시스템, 해상통신정책