Development and evaluation of a ROS package to publish stereo vision sensing data as a LiDAR type message.

Samir E. Silva[1], Rafael P. D. Vivacqua[1], Marco A. S. L. Cuadros[1]

[1]*PROPECAUT, Federal Institute of Espírito Santo*
ES-010 - Km 6,5, Zip Code 29173-087, Serra/ES, Brazil
ehlertsamir@gmail.com, {rafsat, marcoantonio}@ifes.edu.br

Abstract. The ability to obtain a representation of the environment is essential in applications that require knowledge of the arrangement of objects that make up a particular work space. The collection of environmental data through the use of Light Detection And Ranging (LiDAR), cameras or a combination of the two methods are widely studied for several purposes. However, sensors of the LiDAR type still present significatively higher financial costs when compared to alternatives that employ cameras. Currently, these types of sensing have been widely used in autonomous mobile robots and vehicles due to the ability of these sensors to provide good detailing of the scene within its angle of view allowing the objects surrounding the autonomous system to be detected, which permit it to take necessary actions for its safe motion within the environment. The Robot Operating System (ROS) is an open-source framework that helps researchers and developers to build and reuse code for robotic applications. This work proposes the development of a ROS package that uses stereo cameras for the publication of messages characteristic of a LiDAR system and, hence, be able to evaluate the replacement of a LiDAR by stereo cameras with minimum changes to architecture of the system.

Keywords: computer vision, stereo vision, ROS.

1 Introduction

An autonomous robot operating in an environment requires knowledge about relevant objects present within its workspace in order to perform its task. As the task or the environment become more complex usually more information is needed to safely achieve its goal. For operation in unstructured environments, where object positions may be unknow and subject to changes over time, it becomes imperative for autonomous agents to be able to detect objects within the ambient and adapt to situations to complete its task safely.

To acquire ample information of the disposition of obstacles present in a workspace, Light Detection And Ranging (LiDAR) or cameras can be used as sensors. Nowadays, these two sensor types are being broadly used by autonomous mobile robots and vehicles. Some examples are the development of self-driving cars by Tesla [1] and Waymo [2].

There are several researches regarding the use of these two types of sensors on mobile robots and vehicles. D. Hutabarat, et al [3] make use of LiDAR to detect obstacles and enable the navigation of a differential drive autonomous mobile robot to navigate the environment; C. Pang, et al [4] suggest the use of a 2D downward-looking LiDAR for the detection of roads and obstacles by unmanned ground vehicles; S. Gehrig, et al [5] propose the use of prior scenes information in stereo algorithms to improve the detection of obstacles on adverse weather scenarios, presenting good results with a probabilistic approach of the Semi-Global Matchmaking and the Graph Cut algorithms; G. Gil, et al [6] evaluate the usability of stereo vision systems for an Advanced Driver Assistance System for tilting vehicles; Y. WANG, et al [7] argue that the gap between image- and LiDAR based 3D object detection on convolutional neural networks may occur simply due to the 3D data representation used; V. De Silva, R. Jamie, and K. Ahmet [8] propose a framework for the data fusion of LiDAR and wide-angle monocular image sensor.

The Robot Operating System (ROS) is an open-source framework that helps researchers and developers build and reuse code for robotic applications. Its use allows faster development due to the possibility of modulization of a robot's software and thus the same code can be reused for different applications [9].

Nowadays LiDAR sensors still present significatively higher financial costs when compared to alternatives that uses cameras. This work proposes the development of a ROS node to publish partial data from the object detection obtained through the use of stereo vision as a message from a planar laser range-finder. The use of the proposed technique allows the measurement to be interpreted as a planar LiDAR, and permit its use with nodes that offer support to this type of message. This work presents comparative practical results between a system using stereo cameras and a LiDAR on the ROS environment. To do so, different scenarios with known geometries are used to carry out the evaluation of the stereo system and compare it with the results of a SICK TIM551-2050001 LiDAR.

2 Methodology

2.1 Stereo vison

OpenCV and python are utilized on the development of this work. The implementation of the algorithms to obtain the positions of the detected obstacles using stereo vision were done taking reference from StereoPi's blog articles [10] [11], Bradski and Kaehler's book [12] and the OpenCV's documentation [13]. To acquire the positions of the obstacles using stereo vision, first it was necessary to calibrate the stereo pair in order to be able to receive images and convert it into a rectified images to be utilized to calculate the disparity, though the use of block matching algorithm. Then the obtained disparity and the parameters attained during calibration are used to calculate the positions of the obstacles detected on the tridimensional space. The size of the block used by the block matching algorithm in this work was defined as 21. OpenCV library functions were used on the whole process.

2.2 Angle adjustment

To conform the data to a "sensor_msgs/LaserScan" message it is necessary to inform the maximum and minimum measured angles, as well as the angle increments between each measurement. Furthermore, the measurements must be converted to the ranges that reflect the obstacles detected for each angle. In order to verify the relation between the pixels of the image and the pitch and yaw angles observed on the detected obstacles those angles are calculated and displayed graphically on Figure 1.

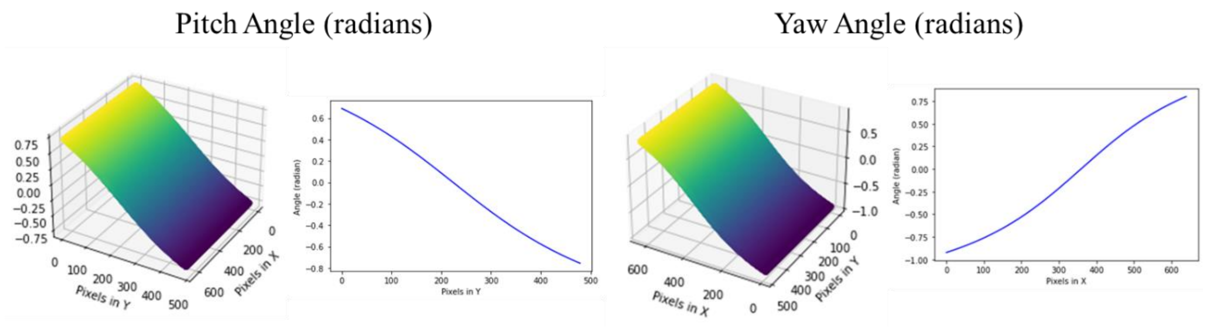

Figure 1: Graphs showing the correlation between pixel position and the angles of pitch and yaw of the point correspondent on the tridimensional space.

The results show that the pitch angle depends only on the pixel positioning in the direction of the image height (y) and the yaw angle depends only on the positioning of the pixel in the direction of the image width (x). As the conversion depends only on the angle of yaw, it is defined a routine to map the closest yaw angle to that of desired angle, this approach was taken in order to reduce computational cost. Based on the configuration and characteristics of the stereo vision system it was determined the use of a maximum angle of 0,79412 radians, minimum angle of -0,83776 radians and increments of 0,00873 radians.

2.3 Distance calculation

In order to facilitate the positioning of the camera, a parameter was implemented to specify the horizontal line in the image to be used when measuring the distance of obstacles. Using the angle mapping described on 2.2, the distance to the objects observed on the selected line for the specified angles are then computed. However, as the disparity depends on the ability to stablish correlation between the images of both cameras, it is possible to get some bad results due to various conditions. To try to obtain better measurement for each of the angles, some of the elements that surround each central point in the vertical are analyzed, this is done by evaluating whether these values within the defined search range are possible, and then use those to estimate the distance using one of the defined methods, the methods proposed for this work are the maximum, minimum, mean and median of the conceivable values detected within the interval. In this work the search range was defined as 9, which means that each conversion may consider the central value and the 4 values below and above it when estimating the distance.

Since the relevant distances for generating the conversion can be reduced to the ones that surround the designated horizontal line within a specific height, as to reduce the computational cost, as done on [11], an image that represents only the Region of Interest (ROI) can be utilized to calculate the disparity. To do so this ROI must be centered on the desired line, have the same width of the rectified image and its height must be at least the size of the block utilized by the block match algorithm plus the number of elements that are examined on the vertical. By repeating this process to both of the obtained rectified images and feeding them to the block match algorithm to calculate the disparity, it is possible to restrict the calculations to only the necessary values, and thus, reduce the processing time.

3 Experimental Setup

In order to acquire the stereo images a StereoPi was used, it is a Raspberry Pi Compute Module compatible board that provides connectivity for two 15-pin CSI-2 cameras, an HDMI output, two USB and one RJ45 connectors 40 GPIO pins [14]. With the intention of evaluating the conversion of the detected obstacles, through the use of stereo vison, to a ROS “sensor_msgs/LaserScan” message. A SICK TIM551-2050001 LiDAR is used to compare the results of the conversion with those of an actual LiDAR sensor.

So as to compare the results of the stereo system and the LiDAR measurements, both sensors were arranged side by side viewing the same scene. In order to evaluate the results, the comparison to the objects placed over a desk, which were measured in order to allow an evaluation of the dimensions returned by both sensors. The sensors data were published trough ROS, so as to allow visualization of the results in real time and ensure that both sensors are viewing the same objects. Both sensors are utilized at the same time and the lateral distance between sensors is considered by providing the transformation between their reference frames. This allows the data provided by the two sensors to be related to know reference frames; the Figure 2 shows the sensors data viewed using Rviz, which stands for ROS visualization and is a powerful visualization tool, allowing the users to visualize the robot model and messages being published by the nodes and thus see what the robotic system is planning and perceiving helping in the debugging of robotic applications [15].

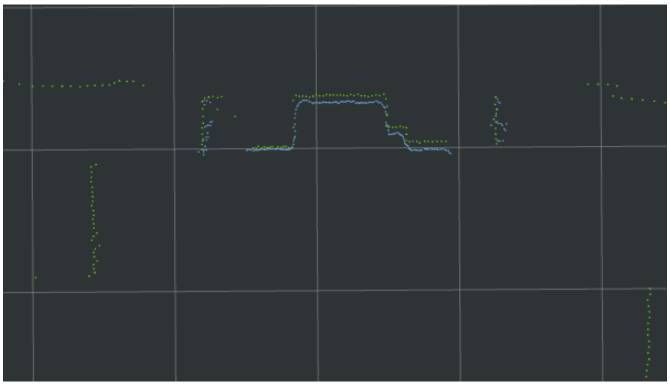

Figure 2:Visualization of both “sensor_msgs/LaserScan” messages on Rviz, LiDAR measurements are displayed in green and the stereo ones in blue.

Three scenes were used when collecting the sensors data, Figure 3 shows the image acquired by the left camera of the StereoPi for each scene. Here it is defined the leftmost scene as scene 1, the one to the right as scene 2 and the lower one as scene 3.

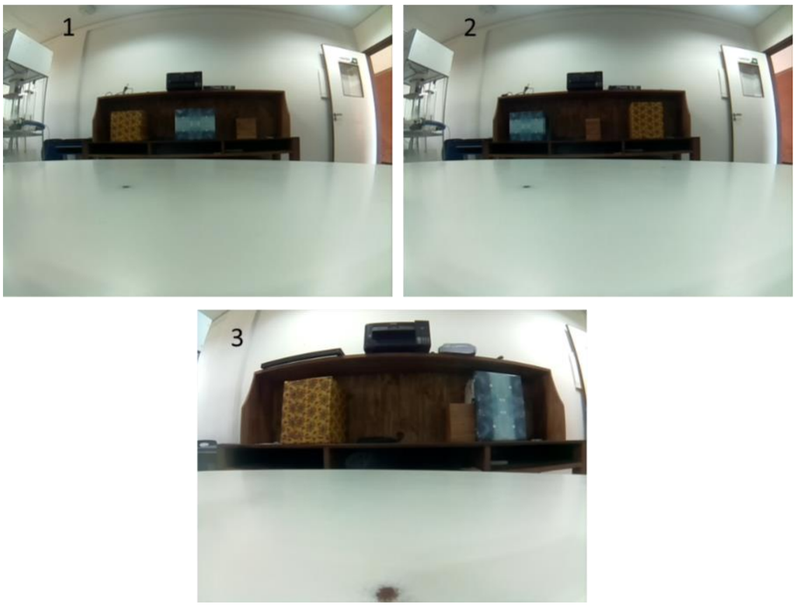

Figure 3: Left camera image from the stereo pair showing views of the three proposed scenes.

The disposition of the objects of interest on the scenes were measured using tape measure to be used as ground truth when comparing the results attained from the sensors, Figure 4 shows graphs generated using the information of the measurement for each scene.

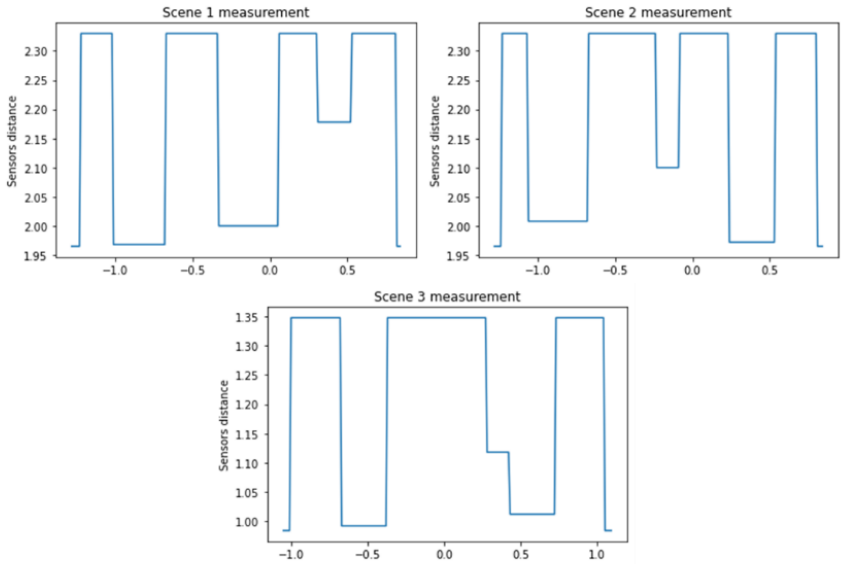

Figure 4: Graphs of the measurements of the proposed scenes.

On the first and second scene the sensors are placed at about two meters from the objects and on the third one the sensors are place at about one meter from the objects.

4 Results

When gathering the data for each method, both the distances returned by the LiDAR and the converted information from the stereo system are collected. The data is then compared between the measurement of the scene, the LiDAR sensing and the result obtained from the correspondent method of conversion. A graph of the positions of the detected objects considering the measurement of the scene, the LiDAR sensing and the results of a conversion using median is presented on Figure 5, where the green line represents the measured dimensions, the red line the LiDAR sensing and the blue line the conversion using median of the calculated values.

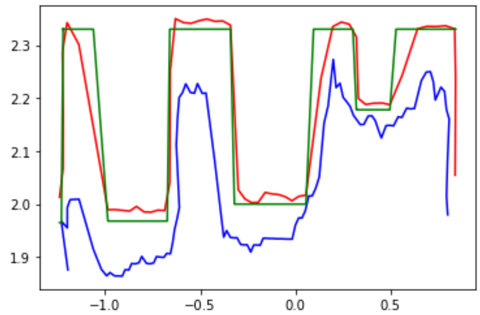

Figure 5: graph showing an example of the measurement that correspond to the scene. Presenting the measured values(green), the LiDAR detection (red) and the values of the conversion from stereo using the median (blue).

With the purpose of analyzing the error on the measurements it is used boxplot to show de variation in the distance when comparing two of the measurement procedures. The results of comparing the LiDAR to each of the proposed methods of conversion for the three proposed scenes are shown on Figure 6.

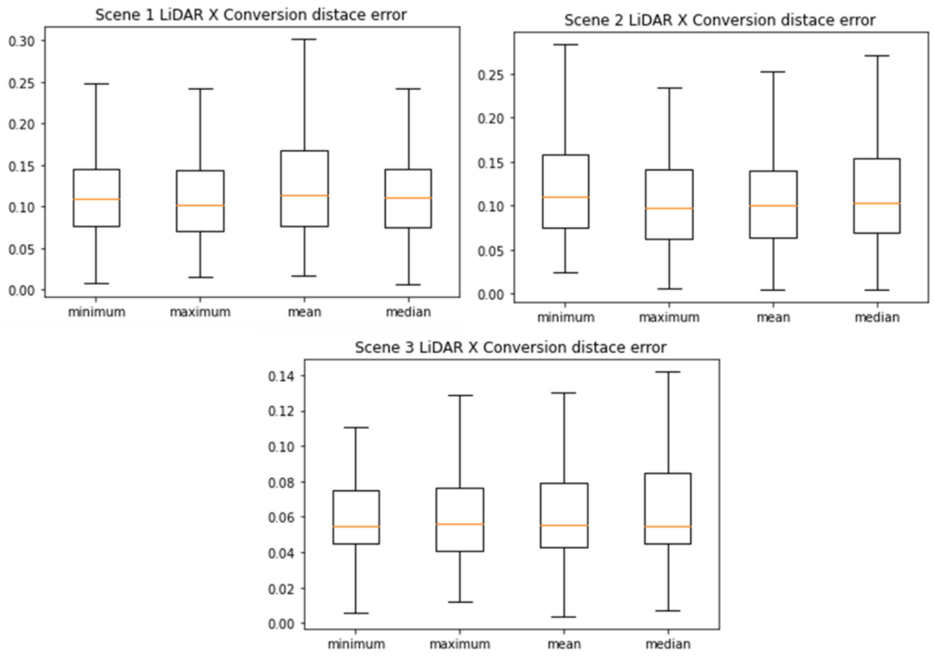

Figure 6: boxplot of the distance between the LiDAR detection of obstacles and each of the methods for calculating the conversion of the detection using the stereo images.

From the data presented in the Figure 6, when observing scene 1 and 2 it is clear that there is a tendency of the difference of the measurements between the two sensors to concentrate at about 10cm. On the third scene there is tendency of values to concentrate at about 6cm.

The results when comparing the measured values from each scene with the distances obtained from the sensors are shown in Figure 7.

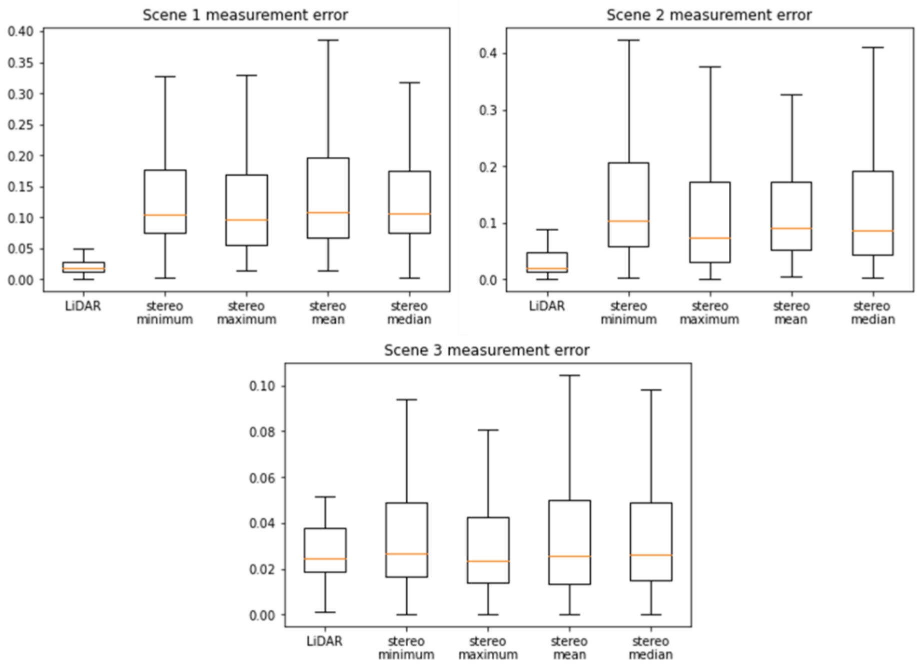

Figure 7: boxplot of the distance between the measurements from the scenes and the detection of obstacles from the LiDAR and each of the methods of calculating the conversion of the detection using the stereo images.

From the Figure 7 can be noticed that the LiDAR detection tends to maintain distance error under 5cm on all the proposed scenes. As for the conversion attained from the stereo system, the distance error tends to fluctuate around 10cm for all method no the first and second scene. The third scene shows a result closer to the LiDAR performance but still present significant amplitude variation of the error comparatively.

5 Conclusion

From the evaluation of the data of the experiments it is observed that the comparison with the LiDAR data can be used as a method to evaluate the accuracy of the measurement provided by the use of stereo vision. As for de performance of the conversion, the analyses show inferior results when compared with an actual LiDAR, however for distances up to 2,33m in most cases its error is kept under 20% and tends to fluctuate around 5%, as for distances of up to 1,35m in most cases the error is kept under 10% and tends to fluctuate around 3%. The results show that the conversion had a closer behavior to thar of a lidar on the third scene, that presents closer obstacles, this behavior is expected as it is an effect of the correlation of disparity to depth.

As for the evaluation of the proposed methods they show similar results, and more data is required to define the better method to estimate the distance.

Moreover, the use of the developed node allows to simulate data published by a planar laser range-finder, and thus provides an alternative to make use of nodes that only support this type of message.

As future work, more data will be collected in order to select a method to estimate de distance for the conversion, the algorithm will be improved in order to reduce its computational cost and its performance will be tested when substituting a LiDAR in a mobile robot navigating in internal ambient.

Acknowledgements. The authors would like to thank the Federal Institute of Espírito Santo, Campus Serra (IFES-Serra) and the Industrial Automation Group (GAIn) for the technical support and the Fundação de Amparo à Pesquisa e Inovação do Espírito Santo (FAPES) for the financial support.

Authorship statement. The authors hereby confirm that they are the sole liable persons responsible for the authorship of this work, and that all material that has been herein included as part of the present paper is either the property (and authorship) of the authors, or has the permission of the owners to be included here.

References

[1] Tesla. "Autopilot". 2021. https://www.tesla.com/autopilot (accessed Aug. 13, 2021).
[2] Waymo. "Waymo Driver". 2021. https://waymo.com/waymo-driver/ (accessed Aug. 13, 2021).
[3] D. Hutabarat et al. "Lidar-based obstacle avoidance for the autonomous mobile robot". 12th International Conference on Information & Communication Technology and System (ICTS). IEEE, pp. 197-202 2019.
[4] C. Pang et al. "Adaptive obstacle detection for mobile robots in urban environments using downward-looking 2D LiDAR". Sensors, v. 18, n. 6, pp. 1749, 2018.
[5] S. Gehrig et al. "Stereo vision during adverse weather—Using priors to increase robustness in real-time stereo vision". Image and Vision Computing, v. 68, p. 28-39, 2017.
[6] G. Gil et al. "Motorcycles that see: Multifocal stereo vision sensor for advanced safety systems in tilting vehicles". Sensors, v. 18, n. 1, p. 295, 2018.
[7] Y. WANG et al. "Pseudo-LiDAR from Visual Depth Estimation: Bridging the Gap in 3D Object Detection for Autonomous Driving". Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition., 2019. 8445-8453.
[8] V. De Silva, R. Jamie, and K. Ahmet. "Robust fusion of LiDAR and wide-angle camera data for autonomous mobile robots." Sensors, v. 18, n. 8, pp. 2730, 2018.
[9] L. JOSEPH. "ROS Robotics Projects". Birmingham. Packt Publishing Ltd., 2017.
[10] StereoPi. "ROS: A simple depth map using Raspberry Pi". 2020. https://stereopi.com/blog/ros-simple-depth-map-using-raspberry-pi (accessed Aug. 13, 2021).
[11] StereoPi. "OpenCV: Comparing the speed of C++ and python code on the Raspberry Pi for stereo vision". 2020. https://stereopi.com/blog/opencv-comparing-speed-c-and-python-code-raspberry-pi-stereo-vision (accessed Aug. 13, 2021).
[12] G. Bradski and A. Kaehler. "Learning OpenCV". 1st. ed. Sebastopol: O'Reilly, 2008.
[13] OpenCV. "OpenCV modules". 2021. https://docs.opencv.org/4.5.2/ (accessed Aug. 13, 2021).
[14] StereoPi Wiki. "StereoPi Specifications". 2020. https://wiki.stereopi.com/index.php?title=StereoPi_Specifications#Features_.26_Specifications (accessed Aug. 13, 2021).
[15] C. Fairchild and T. L. Harman. "ROS Robotics By Example". 2ª. ed. Birmingham. Packt Publishing Ltd., 2017.