Increase the Processing Speed on Slam by Raspberry PI

Nguyen Le Dung, PhD
Department of Automation and Control,
Faculty of Electrical and Electronics Engineering
Ho Chi Minh City University of Technology
VNU-HCM
Viet Nam

Phan Huynh Lam
Maxwell Solution Technology Limited Company,
Viet Nam

ABSTRACT

SLAM algorithm is very important in the process of building maps for robots, the speed of building maps affects much from algorithms, hardware speed, image resolution. The integration on raspberry pi increases the flexibility of building a robot model, but the speed of raspberry pi is not high, so we propose a method of speeding up by building a vector model for the D435i camera and accelerating. process on usb intel modivus.

General Terms

SLAM algorithm, ORB-Slam2 algorithm

Keywords

SLAM, Modivus, D435i, LMeds

1. INTRODUCTION

The current development trend of technology is related to automation systems, intelligent technology systems. It is not only for developers but also technology-related companies or people who are tech-savvy. The development of artificial intelligence, machine learning, deep learning is a topic that they aim to meet the increasing development needs of society, with systems that must ensure the highest productivity and ensure that Best experience for users, reducing human strength during labor and production. Through this many products have appeared, many algorithms, many systems have been built.

Automated systems, self-propelled robots (self-driving cars, drones, etc.) gradually replace people in many different areas of life. These systems require precision sensors, powerful processing systems and have a low price, so in our country this technology has not been developed strongly, not yet operated. to expose all of their effectiveness. In parallel with the development, we have to mention the 3D stereo construction system, recreating a 3D space from the data obtained from specialized cameras, which are being studied by many researchers and businesses. mind because of its high practical applicability. And this is also an open system that can be applied in many different fields such as managing a certain area, or integrating into unmanned aerial systems and probes in some places. Human safety is not guaranteed.

After a period of research on products, building algorithms, recreating 3D space environment, with the strengths and weaknesses of existing algorithms, realize that the ability to develop corresponding models using SLAM algorithms to build this 3D space such as ORB SLAM [1], PTAM [2], can evolve by optimizing the way it handles with lots of potential, applicability in the many areas it offers.

2. OVERVIEW

SLAM algorithm based on 3D floating vision platform using Intel D435i camera, raspberry pi 3 hardware and Inel modivus accelerator.

D435i is intel's floating visual camera, the camera has built-in 3D image building functions. Therefore, the process of creating images is significantly reduced with the combination of two separate cameras, not to mention the D435i also has a pattern projector system to increase the process of creating 3D images, increasing the accuracy of depth. The D435i has a built-in IMU sensor that helps each resulting image to be oriented, reducing the process of building digital maps in the SLAM algorithm.

Figure 1: D435i Intel with IMU inside

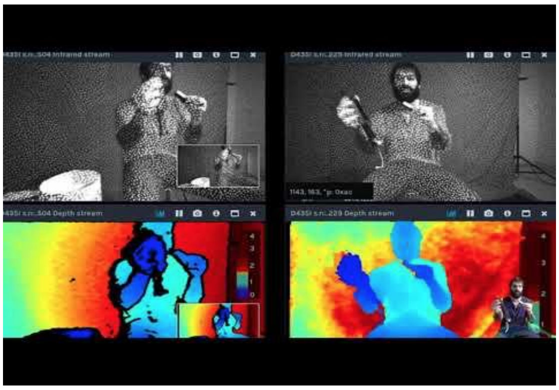

D435i s.n.:504 Infrared stream
D435i s.n.:229 Infrared stream
1143, 163, *p: 0xac
D435i s.n.:504 Depth stream
D435i s.n.:229 Depth stream
4
3
2
1
0

Figure 2: D435i 3D camera and captured images.

Intel modivus is a device that accelerates the processing process.When we implement image processing algorithms, AI, the performance acceleration is very important, requiring the machine to have CPU and GPU speed fast enough to The process takes place in real time.

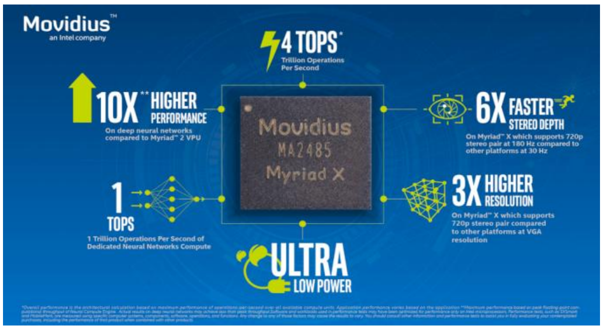

Movidius
an Intel company
4 TOPS
Trillion Operations Per Second
10X HIGHER PERFORMANCE
On deep neural networks compared to Myriad 2 VPU
Movidius MA2485 Myriad X
6X FASTER STEREO DEPTH
1 TOPS
1 Trillion Operations Per Second of Dedicated Neural Networks Compute
3X HIGHER RESOLUTION
ULTRA LOW POWER

Figure 3: Movidus chipset

3. ALGORITHM AND PROCESS ACCELERATION

SLAM algorithm is a map building algorithm. Usually built with 5 modules:

- The basic module [3]:

1. Initialization

2. Tracking (tracking)

3. Mapping (mapping-mapping)

- Additional modules: aiming to stabilize and increase the accuracy of SLAM [3]:

• Relogcalization (relocation)

• Global map optimization (global map optimization)

* Initializaion: from the initial information, a part of the environment is recreated as a starting map. Camera replacement and 3D reconstruction.

* Tracking: is the process of continuously identifying new images on the map, so that they are correlated with the old locations and photos. That is, the correlation between the received 3D image and the overall 2D map of the map. In order to do this, it is necessary to link or trace the characteristics of each image.

* Mapping: calculate the 3D structure of the unknown environment

Relogcalization:

- When needed: when camera movement is too fast or noise occurs leading to Tracking failure

- Work: recalculate camera posture with map compliance.

Global map optimization (global map optimization):

- Because maps often contain cumulative estimation errors depending on the distance traveled by the camera, the map will be misleading

- How it works: the map is refined by considering the consistency of all map information. The first position information is saved, the accumulated error information from the first position to the present is calculated. Based on the information from the loop, we refer to the reference to bind the positions to a reasonable, then adjust the map.

In a nutshell: As listed above, the framework of SLAM algorithms consists of five modules: initialization, tracking, mapping, repositioning and global map optimization. Because each SLAM algorithm uses different methods for each module, the features of the SLAM algorithm depend heavily on the methods used. Therefore, it is important to understand each module of the SLAM algorithm to know its performance, advantages and limitations.

The algorithm is thus to optimize SLAM algorithm we optimize each of its processes. An important factor is to get input information for processing, this input is the 3D point cloud. To shorten this process, we recommend using the D435i camera - a device that enhances 3D point cloud performance significantly compared to conventional cameras. Implementation diagram of D435i is shown as below:

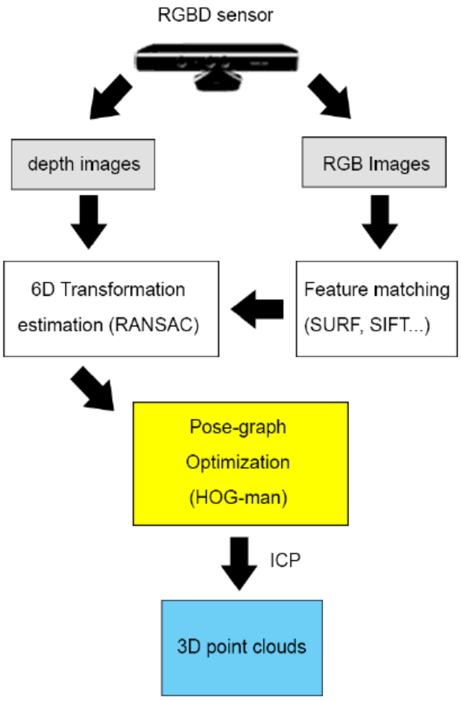

RGBD sensor
depth images
RGB Images
6D Transformation estimation (RANSAC)
Feature matching (SURF, SIFT...)
Pose-graph Optimization (HOG-man)
ICP
3D point clouds

Figure 4: Implementation diagram of D435i

We have 3D point cloud from D435i hardware and IMU value thereby reducing processing time for initialization and tracking (Initialization and Tracking). Thus, the process of creating the map only requires Mapping and Searching acceleration.

Mapping two or more 3D point clouds requires hardware with sufficient RAM to deploy and CPU speed fast enough to perform real time. We can reduce it with the following solution:

Deploy vectorization on 3D image cloud → 2 ^ n vector

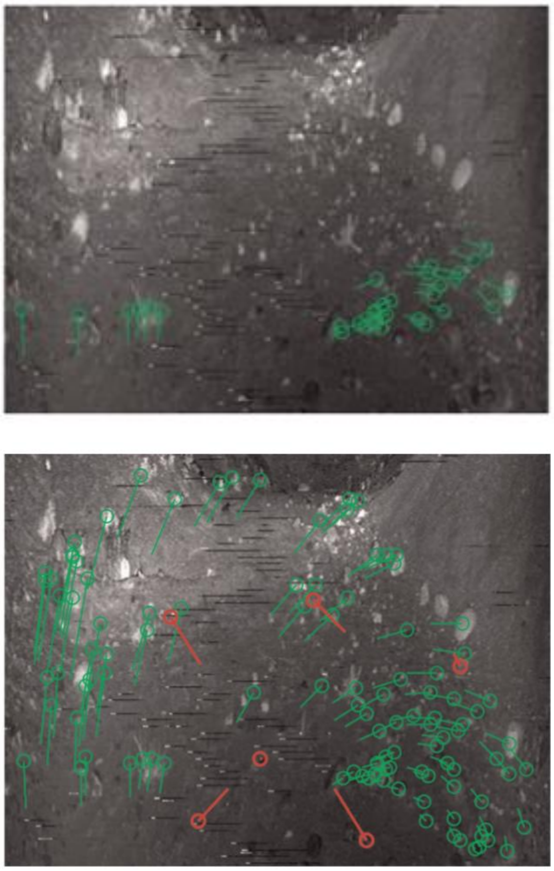


Figure 5: 3D image cloud

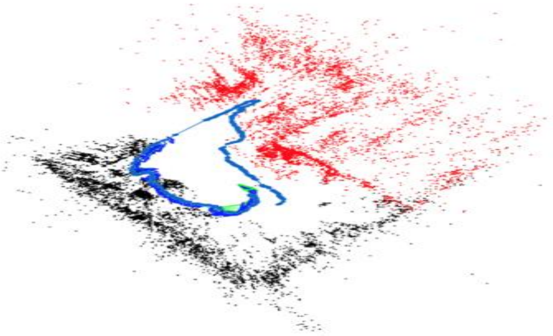


Figure 6: Short-working group find a distance min (LMeds instead of Ransac)

Putting the algorithm through Intel Movidius acceleration processing

The inclusion of intel movidius, requires conversion of the training library as follows:

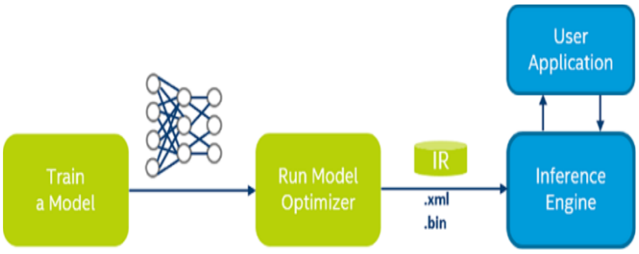
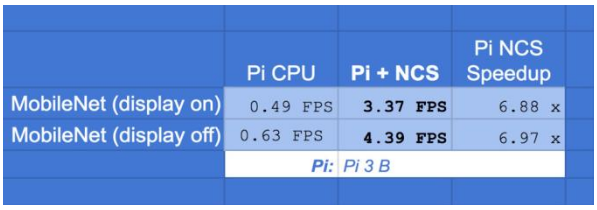


Figure 7: Inclusion of Intel Movidius.

Similar to the Searching process, we can treat as object findding problem in image processing. The use of intel movidus helps to accelerate significantly.



	Pi CPU	Pi + NCS	Pi NCS Speedup
MobileNet (display on)	0.49 FPS	3.37 FPS	6.88 x
MobileNet (display off)	0.63 FPS	4.39 FPS	6.97 x
	Pi:	Pi 3 B	

Figure 8: Result

4. CONCLUSION

The use of D435i significantly reduces the process of creating 3D images, the algorithm of combining 3D image areas with vector reduces processing time, the processing is accelerated on intel's modivus kit, which speeds up 3.7 times compared to normal. The acceleration makes it possible to use SLAM for mobile robots with raspberry pi in real time, so we can put into practice many other integrated applications.

The group's next development direction will upgrade CNN's network for SLAM in the following model:

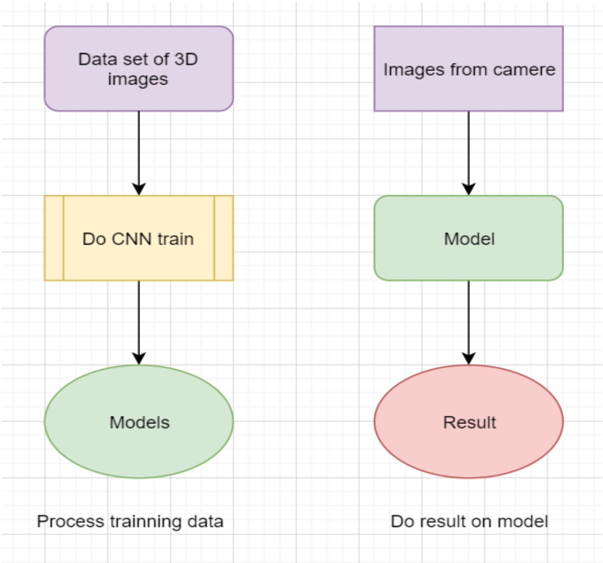


Figure 9: Flowchart of model CNN for SLAM

2D and 3D image data are classified and grouped into arrays, which helps to speed up Tracking-Object process. Compared to the conventional method, the classification is significantly reduced.

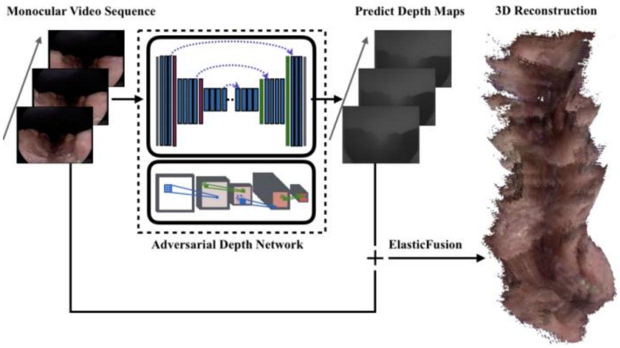


Fig 10. Diagram of model CNN

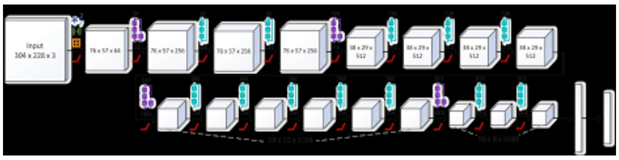


Fig 11. Flow chart of processing data in CNN model

5. ACKNOWLEDGEMENT

This research is funded by Ho Chi Minh City University of Technology – VNU-HCM under grant number T-ĐĐT-2018-76.

6. REFERENCES

[1] Mur-Artal R, Montiel JMM, Tardós JD (2015) ORB-SLAM: a versatile and accurate monocular SLAM system.

[2] Klein G, Murray DW (2007) Parallel tracking and mapping for small AR workspaces In: Proceedngs of International Symposium on Mixed and Augmented Reality, 225–234.

[3] Takafumi Taketomi, Hideaki Uchiyama & Sei Ikeda. Visual SLAM algorithms: a survey from 2010 to 2016

[4] C. Mei, G. Sibley, M. Cummins, P. Newman, and I. Reid, “RSLAM: A System for Large-Scale Mapping in Constant-Time Using Stereo,” IJCV, vol. 94, no. 2, pp. 198–214, June 2010.

[5] S. Williams, V. Indelman, M. Kaess, R. Roberts, J. J. Leonard, and F. Dellaert, “Concurrent Filtering and Smoothing : A Parallel Architecture for Real-Time Navigation and Full Smoothing,” IJRR, vol. 33, no. 12, pp. 1–47, 2014.

[6] A. Ahmad, G. D. Tipaldi, P. Lima, and W. Burgard, “Cooperative Robot Localization and Target Tracking based on Least Squares Minimization,” in ICRA, 2013. [13] R. Reid and T. Braunl, “Large-scale Multi-robot Mapping in MAGIC ¨ 2010,” RAM, 2011.

[7] T. A. Vidal-Calleja, C. Berger, J. Sola, and S. Lacroix, “Large scale ` multiple robot visual mapping with heterogeneous landmarks in semistructured terrain,” RAS, vol. 59, no. 9, pp. 654–674, Sept. 2011.

[8] R. C. Leishman, T. W. McLain, and R. W. Beard, “Relative Navigation Approach for Vision-Based Aerial GPS-Denied Navigation,” ICUAS, 2013.

[9] B. Kim, M. Kaess, L. Fletcher, J. Leonard, A. Bachrach, N. Roy, and S. Teller, “Multiple Relative Pose Graphs for Robust Cooperative Mapping,” in ICRA, 2010.

[10] A. Cunningham, V. Indelman, and F. Dellaert, “DDF-SAM 2.0: Consistent Distributed Smoothing and Mapping,” in ICRA, 2013.

[11] H. Hirschmuller, P. Innocent, and J. Garibaldi, “Fast, unconstrained camera motion estimation from stereo without tracking and robust statistics,” ICARCV, vol. 2, pp. 1099–1104, 2002.

[12] E. Olson, “AprilTag: A robust and flexible visual fiducial system,” in ICRA, 2011.

[13] K. Schmid, F. Ruess, M. Suppa, and D. Burschka, “State Estimation for highly dynamic flying Systems using Key Frame Odometry with varying Time Delays,” in IROS, 2012.

[14] C. Brand, M. J. Schuster, H. Hirschmuller, and M. Suppa, “Submap ¨ Matching for Stereo-Vision Based Indoor/Outdoor SLAM,” in IROS, Hamburg, Germany, 2015.

[15] G. Grisetti, R. Kummerle, C. Stachniss, and W. Burgard, “A Tutorial ¨ on Graph-Based SLAM,” ITSM, vol. 2, no. 4, pp. 31–43, 2010.

[16] G. H. Lee, F. Fraundorfer, and M. Pollefeys, “Robust Pose-Graph Loop-Closures with Expectation-Maximization,” in IROS, 2013. [24] F. Tombari, S. Salti, and L. Di Stefano, “A combined texture-shape descriptor for enhanced 3d feature matching,” in ICIP, 2011.