

# Early Model of Vision-Based Obstacle Mapping Utilizing Grid-Edge-Depth Map



Budi Rahmani, Agus Harjoko, Tri Kuntoro Priyambodo, Hugo Aprilianto

**Abstract:** This paper described a new method of obstacle mapping in an indoor environment utilizing a Grid-edge-depth map. The Grid-edge-depth map contained the information of distance and relative position of the object in the front of the robot. This mapping method utilized this information to mark off the visible obstacle/s in a particular virtual map. The 2D map created as a representative of the environment using a 300 by 500 pixels image. Every pixel represents a one by one cm of the environment and the obstacle's size. The obstacle's size was 30 by 30 pixels when it mapped by the system. It was a fixed size in the mapping process since the system cannot calculate the dimension of the detected obstacle. If the obstacle detected, the system checked its distance in GED-map. Then the system calculated the obstacle's position against the goal, and finally map it in the 2D map. In this case, the proposed method in building a 2D map of the obstacle in the indoor environment combined with the rules to decide the direction of the mobile robot. The rules used to avoid the collision to the obstacle. The evaluation of the method showed that the system could map the detected obstacles, the initial position, and the goal's relative distance and position. The robot also reaches the goal position while avoiding the collision to the obstacle.

**Keywords:** GED-map, Map building, Mobile robot, Stereovision, distance

## I. INTRODUCTION

The first model designed in the previous study is the Grid-Edge-Depth Map. The world map used is attached to the depth map. The depth map obtained from computing the disparity map with the focal length values of the stereo camera and also the distance between the stereo cameras used. While the disparity map is the result of computing two images (right and left) which were acquired by a stereo camera using the SAD (sum of absolute difference) algorithm [1], [2]. The value substitutes the depth map image in the next process to the edge image. A new edge feature image obtained and contained the value of the distance or depth of

the object at each pixel. The image called edge-depth map. Then to determine the position of objects both horizontally and vertically, the edge-depth map is divided into five parts horizontally and also five parts vertically, and called grids. Each grid calculated its minimum value, to bring up a single value attached to each grid. The whole part of the image is called a Grid-Edge-Depth map (GED-map). The Grid-Edge-Depth map shown in Fig. 1

The GED-map informs the minimum distance that has been calculated by the system for each grid. The unit for each minimum distance information is centimetres. Adding information to the program created is intended to facilitate the process of observation when conducting experiments.

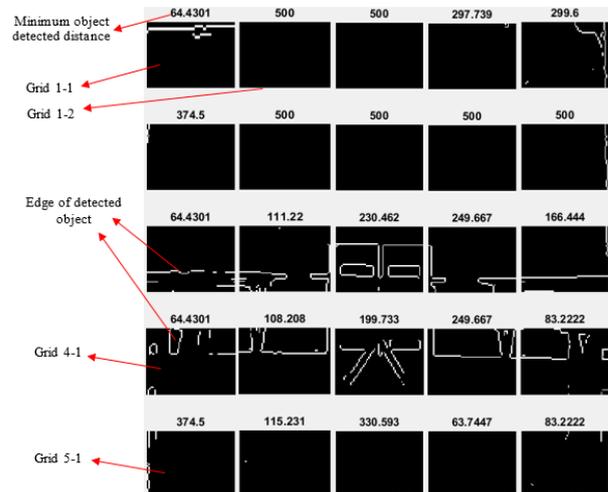


Fig. 1. Grid-edge-depth map

The previous research has designed control rules for a wheeled robot in avoiding the obstacle utilizing the GED-map. This research has been continuing the previous result, especially in the obstacle's map building process. The map is in two-dimensional form. Once the map built by the system, the robot could use it to navigate to the initial position [3][4]. Based on the previous research used, the two-dimensional mapping process carried out was able to map the obstacle and also the robotic navigation point. It is just that the resulting map still contains obstacles that are mapped several times by the system, so the numbers are not the same as the reality in the environment. The mapping problem makes it difficult for the robot when reading back the digital map to return to its initial position. Therefore it is necessary to study the appropriate method or environment mapping algorithm to be used by the robot to return to its original position, by utilizing information on the distance and relative position of the obstacle on the GED-map [5] [6][7][8].

Revised Manuscript Received on December 30, 2019.

\* Correspondence Author

**Budi Rahmani\***, Department of Informatics, STMIK Banjarbaru, Kalimantan Selatan, Indonesia. Email: budirahmani@gmail.com

**Agus Harjoko**, Department of Computer Science and Electronics, Faculty of Mathematics and Natural Science, Universitas Gadjah Mada, Yogyakarta, Indonesia. Email: aharjoko@ugm.ac.id

**Tri Kuntoro Priyambodo**, Department of Computer Science and Electronics, Faculty of Mathematics and Natural Science, Universitas Gadjah Mada, Yogyakarta, Indonesia. Email: mastri@ugm.ac.id

**Hugo Aprilianto**, Department of Informatics, STMIK Banjarbaru, Kalimantan Selatan, Indonesia. Email: hugo.aprilianto@gmail.com

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>)

The use of sensors in the robotic navigation system always adjusted to the operational environment.

A robot can navigate both inside and outside a room [9]. In Outdoors navigation, the global positioning system is an alternative for its navigation base while other sensors used for robot navigation purposes include ultrasonic sensors, infrared sensors, laser range finders, cameras and other sensors. Some robots are only equipped with one type of sensor, for example, ultrasonic. Nevertheless, some are equipped with several sensors at once, for example, cameras and laser range finders [10].

Camera sensors are sensors that can provide real information about the environment compared to other sensors. Therefore the use of cameras as the primary sensor for autonomous robot navigation continues to grow until now. Table 1.1 shows the advantages of the camera compared to other sensors used for robot navigation purposes. In terms of cost, for example, the camera sensor is relatively inexpensive. Then the camera sensor also has good accuracy, variety of uses, and relatively unlimited range.

Table 1. Comparison of the use of sensors in robots [11]

Sensor	cost	Accurac y	Versatili ty	Range
Sonar	Mediu m	Poor	Few	Limited
Laser	High	Good	More	Limited
Vision	Low	Good	Many	Unlimite d

Robots equipped with sensors can then be programmed to have capabilities such as mapping, random exploration, and autonomous navigation [12] [13]. Autonomous navigation divided into three categories: map-based, map-building, and mapless. Map-based navigation systems tend for the environment to remain unchanged for quite an extended period. Therefore in the navigation process, the robot can utilize a static map that contains in detail the position of the obstacle and also information on its distance from a particular position, for example from the initial position the robot move [12][14][15].

Another autonomous robot navigation category is map-building. In this category, the robot builds the environment map while the navigation process [16]. The map may contain information about the position of the obstacle, or landmarks contained in the environment. In this category, the robot may utilize or reads the map to navigate without utilizing its sensors. The process of navigating and building a map conducted almost simultaneously. It is mean when detecting the existence of an obstacle, the system also deciding the direction of movement, and mapping the obstacle [17], [18], [19].

II. RESEARCH METHOD

This research used a wheeled robot. The robot consists of four Omni wheels and four DC motors. They are controlled by an Arduino mega based microcontroller that receives commands from the computer via a USB port. The robot uses a holonomic-drive system. The system allows the robot to move in all directions according to the command. All the commands coded into an upscale alphabetical code, that is A, B, C, D, E and F., In this case, the robot programmed to go

forward (A), backwards (B), move to the right (C), move to the left (D), move 45 degrees forward to the right (E), and move 45 degrees forward to the left (F).

The goal of this mapping and or navigating process is a yellow lamp, placed on a table. The same another table also used as the obstacle when the robot was moving during its navigation process. This study used an assembled stereo camera. The calibration process used a toolbox by Jean-Yves Bouguet [14,15]. Both cameras connected to the computer via a USB port [20][21][22]. **Error! Reference source not found.** and **Error! Reference source not found.** show our experimental robot this study.

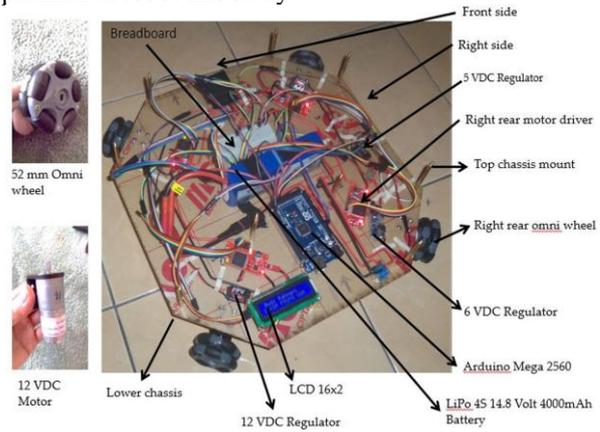


Fig. 2. Assembled wheeled robot (slave controller) [2]

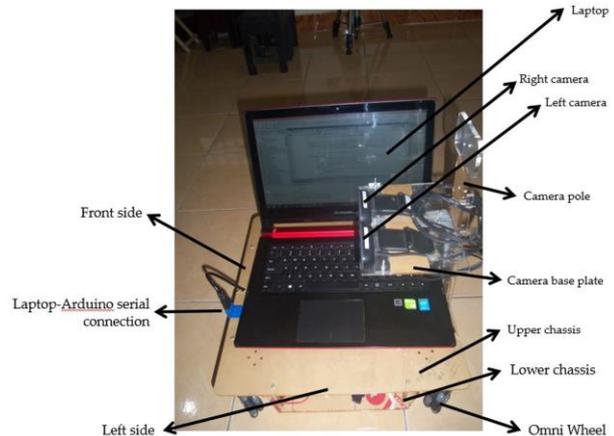


Fig. 3. A Complete robot with a stereo camera [2]

This research proposed an algorithm to build a 2D map using a 300x500 pixels image. The first process of the proposed algorithm is that the system creates a 300x500 pixels white image. Then the system marks the start position of the robot using a 37x37 pixels grey image square (pixel's intensity value is 230). The next process initializes the size of the obstacle row/ors (25 pixels) and the minimum distance of the robot to the obstacle/dRO (65 pixels). Next process is building a GED-map, and it described in [1]. The system then detects the presence of a yellow lamp as the goal of navigation. If the system detects the yellow lamp, it gives a bounding box and specifies the midpoint (xYellow and yYellow values). Then the system reads the distance of yellow lamp on the depth map based on its coordinates (xYellow and yYellow).

Furthermore, the system marks the yellow lamp in the 2D map (pixel's intensity value is 100).

The next process is to calculate the value of the obstacle distance to the lamp ( $dOL$ ) using the following formula:

$$dOL = yellowLampDistance - dOL - orS_e$$

Then the system checks whether the  $dOL$  is less than one or not. If the  $dOL$  is less than one, then the system set  $dOL$  the value to one. Furthermore, if the yellow lamp distance is more than 65 cm, and there are no obstacles, the system instructs the robot to move forward to the goal.

If the distance of the yellow lamp is more than 65 cm and there is an obstacle in front of the robot with a less than or equal to 65 cm distance, the system instructs to avoid the obstacle [2]. Then the system marks the found obstacle in the 2D map while determining these values:

$$bA \text{ (top row)} = dOL$$

$$bB \text{ (bottom row)} = bA - orS = bA - 25$$

$$cA \text{ (Initial column)} = 121 + ((60-orS) / 2) \text{ and}$$

$$cB \text{ (End column)} = cA + orS$$

The steps of the algorithm are:

**Initialisation:**

whiteImage = 255\*ones (500, 300, 'uint8');

whiteImage (433:470,132:169) = 230;

obstacleRowSize (orS) = 25;

minimumDistanceRobotObstacle (dRO) = 65;

Build a GED-map

Detect the presence of the yellow lamp

If the yellow lamp is detected

Give a bounding box of yellow lamp

Specify xYellow and yYellow values (the midpoint)

Read the distance of yellow lamp on the depth map based on xYellow and yYellow

Mark the Yellow lamp in the whiteImage

xYellow = xYellow - 10;

If xYellow < 1

xYellow = 1;

End If

xYellow2 = xYellow+10;

whiteImage (20:30,xYellow:xYellow2) = 100;

Calculate the obstacle distance to the lamp.

$dOL = yellowLampDistance - dOL - orS$

If  $dOL < 1$

then  $dOL = 1$ ;

End If

If  $dOL > 65$  and No Obstacle

Move forward;

End If

If  $dOL > 65$  and there is an Obstacle with  $\leq 65$  cm

Avoid the collision to the obstacle;

Mark the found obstacle using the value of initialRow, bottomRow, initialColumn and endColumn

$bA = uint16(dOL-23)$ ;

$bB = uint16(bA+37)$  ;

$cA = uint16(xYellow-(37/2))$ ;

$cB = uint16(xYellow+(37/2))$ ;

End If

End If

the environment, including the wheeled robot, the yellow lamp as a goal of navigation, and the obstacles. In the experiment of the map building process, the decision to avoid the obstacle based on rules in [2]. **Error! Reference source not found.** shows the 2D map (left side) and the real-time objects that the robot can see through the stereo camera. Once the robot is turned on, and the system starts to detect the presence of the goal, the system creates a 300x500 pixels image at the same time. Then the system mapped the robot's position against the detected goal position and mapped the goal in the 2D map. If the system detected the goal (in this case is a yellow lamp placed on the 30 cm height of a table), the robot would start to move. The destination is the yellow lamp. While the robot moves to the goal, it also detects the presence of the obstacle. At the same time, the system maps it in the 2D map. The system also tries to avoid the collision to the obstacle. The process of detecting and mapping the obstacle in the 2D map described in **Error! Reference source not found.** The position of the mapped obstacles in the 2D map calculated by the system. The calculation based on the distance of the robot to the obstacle. The calculation also based-on the distance of the robot to the goal. The distance of the obstacle to the robot calculated and called the Grid-Edge-Depth map [1], [2]. The information from GED-map always compared to the distance and position of the goal to the robot. It was the main idea of this paper. **Error! Reference source not found.** then shows the scene when the robot succeeds to avoid obstacles to its left position. **Error! Reference source not found.** shows the scene when the robot got through the obstacles and still detected the goal at the same time.

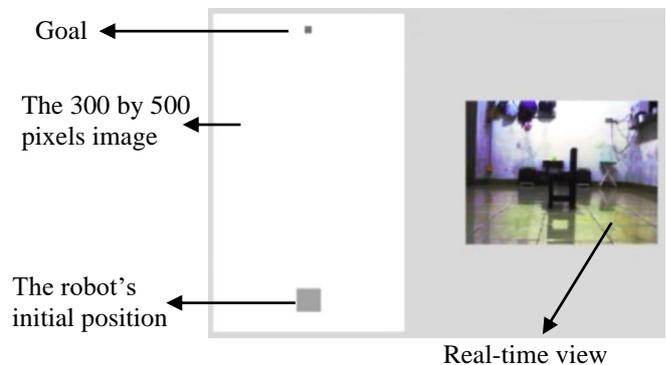


Fig. 4. An initialization of map building's image and the goal detection in the first proses

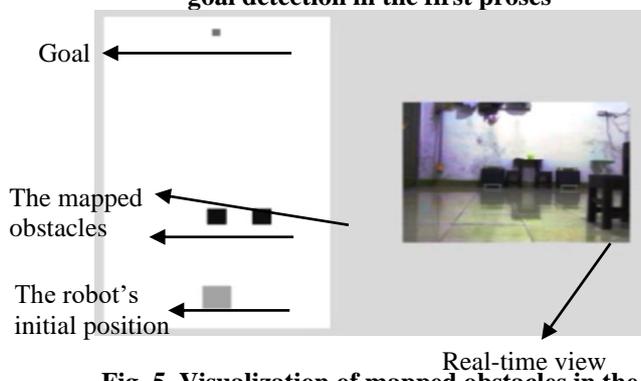


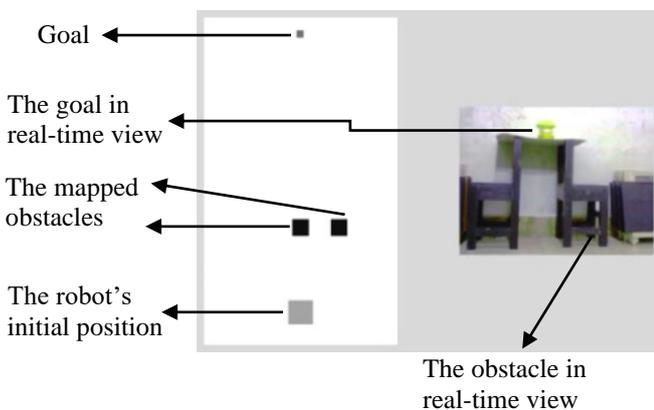
Fig. 5. Visualization of mapped obstacles in the navigation's process

**III. RESULTS AND ANALYSIS**

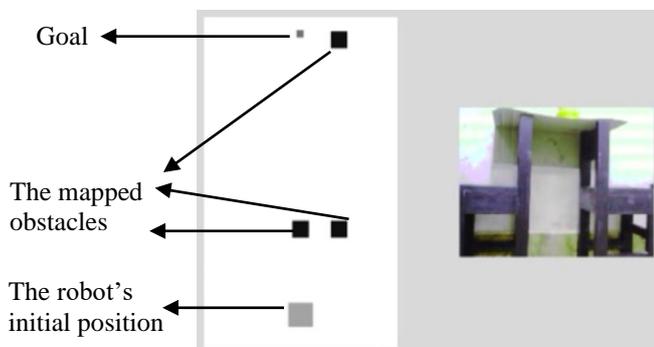
The obstacle map built using a 300 by 500 pixels image. Every pixel represented the one-centimetre size of objects in

Since the robot does not reach the goal, as shown in **Fig. 4** and **Fig. 5**, the navigation and mapping process is not finished yet. The robot continues to navigate to the goal position while avoiding the collision at the same time against the obstacle.

Fig. 6 showed the final scene when the robot reached its goal or final position. The final 2D map also shows in **Fig. 7**. The final 2D map shows that there are three detected obstacles. Real-time view from the camera in **Fig. 5** shows that the final obstacle seen on the right side of the goal. Then the system also mapped the obstacle in the 2D map. However, the system did not map the table where the yellow lamp placed on it. The system did not detect the table or the chairs as an obstacle since it is a concern to the yellow lamp detection as the robot's goal or destination. The system also did not map the detected table into the 2D map. The system calculates the distance to the robot (in GED-map).



**Fig. 6. Visualization of mapped obstacles to the goal position**



**Fig. 7. A 300 by 500 pixels image as the final map when the robot reaches the goal**

The experiment conducted and found that the system sometimes mapped the same position of detected obstacle twice. Ideally, the system compares the detected and mapped of the obstacle when it changes the robot's position because of avoiding the collision to the obstacle's process. Alternatively, the system can delete the new mapped of a detected obstacle when it changes the robot's position when succeed avoiding the collision against the obstacle.

### IV. CONCLUSION

GED-map based obstacle mapping in an indoor environment described in this study. A 300 by 500 pixels image used as an initial blank map. This research proposed a new method to placed a visualization of a known size of the

obstacle in the blank image as a map. Every pixel of the map represents one centimetre of the obstacle and the goal. In this case, the goal is a yellow lamp placed on the table. Using the rules, the system detecting the goal, detecting the obstacle, and deciding the decision of the direction of the robot. When the system found the obstacle, the system placed a 30 by the 30-pixel black image on the map. The position of the detected obstacle at the same time is compared to the goal (yellow lamp) position and distance. So the system can estimate the position of a placed obstacle in the map. The experiment showed that the proposed method is effective. Especially in mapping the obstacles in the 2D map, based on its relative position to the goal as a comparison of distance. Some of the experiment results showed in **Fig. 4**, **Fig. 5**, **Fig. 6**, and **Fig. 7**. The problem came when the system placed more than once of the same obstacle in the map. This problem becomes our next research to fix the problem.

### ACKNOWLEDGEMENTS

This work supported by a research grant from the Ministry of Research, Technology and Higher Education of the Republic of Indonesia (2019).

### REFERENCES

1. B. Rahmani, A. Harjoko, and T. K. Priyambodo, "Grid-edge-depth map building employing sad with sobel edge detector," *Int. J. Smart Sens. Intell. Syst.*, vol. 10, no. 3, pp. 551–566, 2017.
2. B. Rahmani, A. Harjoko, and T. K. Priyambodo, "THE ACCURACY IMPROVEMENT OF OBJECT'S DISTANCE MEASUREMENT BASED ON GRID-EDGE-DEPTH MAP IN THE DETERMINATION OF WHEELED ROBOT'S DECISION OF DIRECTION," Universitas Gadjah Mada, 2019.
3. B. Rahmani *et al.*, "Review of Vision-Based Robot Navigation Method," *IAES Int. J. Robot. Autom.*, vol. 4, no. 4, pp. 254–261, 2015.
4. T. Pire, T. Fischer, J. Civera, P. De Cristoforis, and J. J. Berles, "Stereo parallel tracking and mapping for robot localization," *IEEE Int. Conf. Intell. Robot. Syst.*, vol. 2015-Decem, pp. 1373–1378, 2015.
5. N. Kuntz, "Depth Map Tutorial," no. 2006, pp. 1–4, 2016.
6. I. A. Shcherbatov, N. T. Dung, V. P. Glazkov, and O. Protalinskiy, "Control movement of mobile robots inside building based on pattern recognition algorithm," in *2017 International Siberian Conference on Control and Communications (SIBCON)*, 2017.
7. J. Y. Park, S. S. Kim, C. S. Won, and S. W. Jung, "Accurate vertical road profile estimation using v-disparity map and dynamic programming," in *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, 2018, vol. 2018-March, no. 1, pp. 1–6.
8. G. Zhou, L. Fang, K. Tang, H. Zhang, K. Wang, and K. Yang, "Guidance: A visual sensing platform for robotic applications," *IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work.*, vol. 2015-October, pp. 9–14, 2015.
9. J. K. Yoo and J. H. Kim, "Gaze Control-Based Navigation Architecture with a Situation-Specific Preference Approach for Humanoid Robots," *IEEE/ASME Trans. Mechatronics*, vol. 20, no. 5, pp. 2425–2436, 2015.
10. P. Alves, H. Costelha, and C. Neves, "Localization and navigation of a mobile robot in an office-like environment," in *2013 13th International Conference on Autonomous Robot Systems*, 2013, pp. 1–6.
11. D. S. Kumar, "Vision-based robot navigation using an online visual experience," International Institute of Information Technology, Hyderabad, 2007.
12. E. K. Kim, H. Cho, E. Jang, M. K. Park, and S. Kim, "Map Building of Indoor Environment using Laser Range Finder and Geometrics," in *IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*, 2015, pp. 1–6.
13. K. Schmid, T. Tomic, F. Ruess, H. Hirschmuller, and M. Suppa, "Stereo Vision based indoor / outdoor Navigation for Flying Robots," *IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, pp. 3955–3962, 2013.

14. R. Mittal, N. Mishra, and V. Pathak, "Exploration of designed robot in non linear movement using EKF SLAM and its stability on loose concrete surface," *Int. J. Eng. Adv. Technol.*, vol. 9, no. 1, pp. 875–878, 2019.
15. K. Prasanna Lakshmi, "Motion of multiple robot in a curved boundary & obstacles," *Int. J. Eng. Adv. Technol.*, vol. 9, no. 1, pp. 6011–6014, 2019.
16. T. T. Auke-dirk, "Vision for Mobile Robot Navigation : A Survey Keyfeatures," *Pattern Anal. Mach. Intell.*, vol. 24, no. 2, pp. 237–267, 2002.
17. E. Nelson, M. Corah, and N. Michael, "Environment model adaptation for mobile robot exploration," *Auton. Robots*, vol. 42, no. 2, pp. 257–272, 2018.
18. H. Ghazouani, M. Tagina, R. Zapata, L. Manouba, and C. U. De La, "Robot Navigation Map Building Using Stereo Vision Based 3D Occupancy Grid," *J. Artif. Intell. Theory Appl.*, vol. 1, no. 3, pp. 63–72, 2010.
19. Z. Kowalczyk and T. Merta, "Three-dimensional Mapping for Data Collected using Variable Stereo Baseline," *IEEE*, pp. 1082–1087, 2016.
20. B. O. Kennedy, "Stereo Camera Calibration," University of Stellenbosch, 2002.
21. J.-Y. Bouquet, "Camera Calibration Toolbox for Matlab," 2015. [Online]. Available: [http://www.vision.caltech.edu/bouquetj/calib\\_doc/](http://www.vision.caltech.edu/bouquetj/calib_doc/).
22. M. Papoutsidakis, "Intelligent Design and Algorithms to Control a Stereoscopic Camera on a Robotic Workspace Z-error Z-Error," *Int. J. Comput. Appl.*, vol. 167, no. 12, pp. 32–35, 2017.

### AUTHORS PROFILE



**Budi Rahmani** received his bachelor's of Electrical Engineering, Master of Informatics, and Doctor of Computer Science from Yogyakarta State University and Dian Nuswantoro University, and Universitas Gadjah Mada in 2003, 2010 and 2019 respectively. He was interested in Computer Vision, Embedded System and

Robotics. Currently, his research focused is computer vision and control system for the robot. His other research interests include decision support system using an artificial neural network. His contact email: [budirahmani@gmail.com](mailto:budirahmani@gmail.com) <http://budirahmani.wordpress.com>



**Agus Harjoko** received his bachelor's degree in 1986 Universitas Gadjah Mada, Indonesia; His master degree in 1990 from University of New Brunswick, Canada; and his doctoral degree in 1996 from University of New Brunswick, Canada. Since 1987 he has been a lecturer and a researcher at the Department of Computer Science and

Electronics, Universitas Gadjah Mada. His research interests lie in the field of Digital Image Processing, Machine Vision, Sensor Network, Multimedia IR (Image, Audio, Video Processing). His contact email: [aharjoko@ugm.ac.id](mailto:aharjoko@ugm.ac.id) <http://acadstaff.ugm.ac.id/aharjoko>



**Tri Kuntoro Priyambodo** currently he is an Associate Professor at the Department of Computer Science and Electronics, Gadjah Mada University. He is a member of IEEE. He also holds a position as a Secretary of Satellite and Aerospace Electronics Research Group, Universitas

Gadjah Mada. During 2010-2013 he was INSPIRE National Project Leader. His responsibility was coordinating the development of the First Indonesia Inter-University Satellite (IiNUSAT-1), a project funded by Directorate General of Higher Education, Ministry of Education and Culture. He actively joins to Small Satellite Standardization group in Kitakyushu Institute of Technology. Also, active in APSCO's Small Student Satellite Project Meeting. Currently, he conducts research in the field of nanosatellite and Autonomous Unmanned Systems. His other research interests include Computer Network Security, eGovernment Systems. His contact email: [mastri@ugm.ac.id](mailto:mastri@ugm.ac.id) <http://acadstaff.ugm.ac.id/mastri>



**Hugo Aprilianto** received his bachelor's of Informatics, and Master of Informatics from Universitas Dr Soetomo Surabaya and Sekolah Tinggi Teknik Surabaya in 1998 and 2007 respectively. Currently he is a lecturer at STMIK Banjarbaru. He was impressive in Computer

Architecture, Embedded System, and Neural network. He can be contacted by email: [hugo.aprilianto@gmail.com](mailto:hugo.aprilianto@gmail.com)