# Automated mapping of ground planes for Visual SLAM applications[*]

Juan Felipe Rodriguez Vasquez, Mechanical engineer[1]
[1]Universidad EAFIT, Colombia, jrodri56@eafit.edu.co
*Mentor:* Davinson Castano Cano, PhD[2]
[2]Universidad EAFIT, Colombia, dcasta25@eafit.edu.co

***Abstract– With the increased use of machine learning methods in all areas of engineering, new methods for acquiring and training data need to be devised. In Visual SLAM applications using ground planes as image data, there is still a lack of research in this area. In this work, we analyze how certain parameters influence the production of ground plane panoramas intended for training of data-driven algorithms. The parameters tested are the lighting conditions, the terrain type and the image source type (video or picture). For this purpose, we generated four panoramic pictures of challenging terrains. The resulting analysis serves as a guide to other researches that want to produce a dataset of images with the purpose of training deep learning models for use in industrial environments.***

***Keywords—Visual SLAM, data generation, ground textures, ground panorama, localization.***

## I. Introduction

Localization is one of the fundamental fields of research in autonomous robotic systems, as a lot of the functionality of such a system rely on the robot knowing its current position accurately. Simultaneous localization and mapping (SLAM) is the problem of estimating the robot position and orientation while mapping the surrounding environment, for which a lot of progress has been made in sensors, algorithms, and approaches.

One of these approaches is the Visual SLAM one, in which a camera (or multiple ones) is used as the main sensor used for both mapping and localization. Depending on their intended environment, Visual SLAM methods can be categorized as indoor or outdoor methods. In outdoor environments, the camera is usually pointed to planes perpendicular to the floor, that is, the front, back or sides of the robot. However, in open indoor environments such as warehouses or production lines, the lack of proximity to features makes looking to the ground or the ceiling a better approach (Ground SLAM and CV-SLAM respectively), as the features are usually closer and are therefore more reliable.

Another added benefit of pointing the camera upwards or downwards is that in a dynamic environment, such as a modular factory line, the features of the ground or ceiling are less variant in time. This makes the methods more robust and reliable in time at a much less cost when compared with other types of sensors, such as LIDAR or RGB-D cameras. Better precision can be further obtained by using sensor fusion with the on-board odometry or a kinematic model of the robot.

Multiple algorithms or methods have been proposed using these techniques [1], [2], with ongoing research on the topic [3]. More recently, with the surge of data-driven algorithms and machine learning techniques, a variety of new methods have been developed. Among others [4], [5] and [6] show very promising results in this field. In [4], they present a solution using convolutional neural networks trained to predict depth maps and fusing them with depth measurements from direct monocular SLAM. In [5], the authors optimize the convolutional procedure by introducing parallelism to the neural network, achieving an improvement in accuracy with a lower number of parameters. Finally, in [6], they use an online training procedure that gives the neural network the ability to adapt itself to new, unknown environments.

These methods, in spite of the advantages they present, require big amounts of data to work properly, as their performance is linked directly with the quantity and quality of the collected data. The task of collecting the data is then a crucial step of developing one such algorithm, and it is not a trivial one when environmental effects like lighting conditions, camera lens corrections, motion blur or terrain flatness are taken into account.

In this paper, we analyze the procedure of producing ground plane image data (or map) in the form of stitched panoramas of ground textures. We test the influence of different parameters in the resulting image; the lighting conditions, the texture of the floor and the image generation procedure. These panoramas can later be used as training data for the aforementioned algorithms by, for example, cropping them and generating "semi-synthetic" data in the form of image sequences that simulate real world motion and working conditions.

The rest of this paper is organized as follows: in Section II the related work is presented. Then, in Section III we present the methodology of acquiring the data and the setup of the results. Afterwards, in Section IV we present the results obtained. Section V presents the discussion of the results to finally present the main conclusions of the work in Section VI.

[*] This work was supported by EAFIT University and the Vicerrectoría de Descubrimiento y Creación. Both authors are with the with the Operations and Logistics Group from the Production Engineering Department, School of Engineering.

**Digital Object Identifier:** (to be inserted by LACCEI).
**ISSN, ISBN:** (to be inserted by LACCEI).

## II. Related Work

### A. Panorama Stitching

Regarding panorama stitching, research has made great advances with still-going efforts to produce faster methods that can give results from lower quality pictures, in more challenging environments. For example, [7] and [8] introduce methods for creating panoramas with videos as input, with [7] focusing on videos with low detail. In [9] and [10], the authors use SURF features to produce results using different procedures. Additionally, [11] presents a method tailored for planar panorama stitching together with exposure correction.

### B. Visual SLAM Datasets

In Visual SLAM, several datasets exist for benchmarking SLAM algorithms. Some popular ones are the KITTI Dataset [12], the TUM Dataset [13] and the EuRoC Dataset [14]. In KITTI, they obtain images from a set of cameras placed in the outside of a car that drives in multiple outdoor environments. TUM uses a Pioneer robot with the camera of a Kinect sensor mounted on top of it to produce pictures in an indoor environment. On the other hand, in EuRoC they obtained data from cameras present on a Micro Aerial Vehicle.

Nevertheless, to the best of our knowledge, no dataset for two dimensional data or procedure for generating one has been proposed yet.

## III. Methodology

We used an omnidirectional robot with a camera mounted on top of it as shown in Fig. 1. The robot, controlled by a computer, followed a predefined path capturing images of three ground planes, these had both different textures and different lighting conditions. In total, four panoramas were created using two procedures outlined below.

### A. Procedures

Two different procedures were followed in order to produce the panoramas:

1) The first one, **picture mode**, consists of alternating between moving the robot and capturing images as such: First, a picture was taken at the initial position, then the robot moved to the next position in the trajectory and waited for a short period of time to take the next picture. This was repeated until the trajectory was complete. The waiting time was introduced in order to reduce motion blur in the photos.
2) The second one, **video mode**, consists of the robot moving while recording a video at the same time. Pauses in the movement were also carried out in between *x* and *y* movement, but the video was not paused in these moments.

Each mode has a set of advantages and disadvantages. Picture mode takes considerably more time to produce an image in the same area as video mode, but has the benefit of low motion blur together with lower stitching time, as the stitching program has to process fewer images overall. Video mode has the added inconvenience of motion blur and higher stitching time, but it can take pictures faster and, because of the way the wheels work, can have more precise movement over long distances.

### B. Hardware

For the tests, we built a mobile robot that has three main components:

- An omnidirectional robot: **KR0003 4WD Mecanum Wheel Mobile Robotic Platform/Kit** [2] . The embedded Arduino Mega 2560 controls the motion and allows serial communication with the Computer.
- A webcam (Logitech C270). We take images at a resolution of 640x480.
- A Laptop computer (HP ProBook 440 G3) that processes the high-level algorithm for the mapping routine and that sends G-code commands to the microcontroller in the robot.

We have fixed the camera to the mobile platform using a laser-cut acrylic structure with the laptop sitting on top of the mobile platform.

The four mecanum wheels the robot has, give the ability to move in all directions in the plane. The robot has four ultrasonic sensors located at the sides as well, but we do not use them in the present study. The robot design has a pivot in two of the four wheels to maintain the grip of the wheels on the floor and to avoid slip.

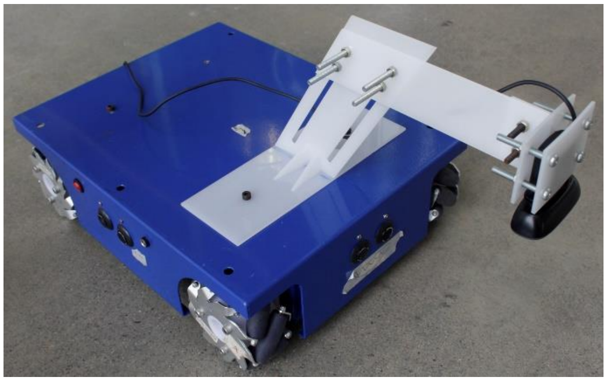

Fig. 1 Robot used, the computer is not present in this picture.

### C. Software

[2] http://www.kingkongrobot.com/index.php?m=content&c=index&a=show&catid=22&id=21

The robot was coded to receive G-Code as input, that is, when the robot is turned on, the initial position is set as the (0,0) coordinate. Then, when a G1 command is sent by the computer with the coordinates ($x$, $y$) the robot moves to that point in the plane. Even though simultaneous movement in the two coordinate axes is possible, it was not used because of the reasons outlined below.

To capture the pictures in picture mode, Python together with OpenCV was used, whereas in video mode the images were captured by the Gnome application Cheese. In both modes, communication between Python and Arduino was achieved using Pyserial.

The images were stitched using the software Microsoft ICE, a program specially designed for creating panoramic images from a set of pictures or a video. We created all panoramas assuming planar motion, except for the one in a stone floor, where assuming rotary motion with an orthographic projection led to better results.

### D. Trajectories

All trajectories were zig-zagging trajectories that were performed as shown in Fig. 2. All the trajectories were discretized by having a step size of four centimeters in both directions. This was done in order to have an overlapping region in two consecutive images.

In image mode, pauses were performed at every resulting point in the resulting path. As the camera mount tended to vibrate more with horizontal movement, the trajectories in video mode had some pauses at places where this movement took place. This was chosen this way in order to minimize the vibration of the camera.

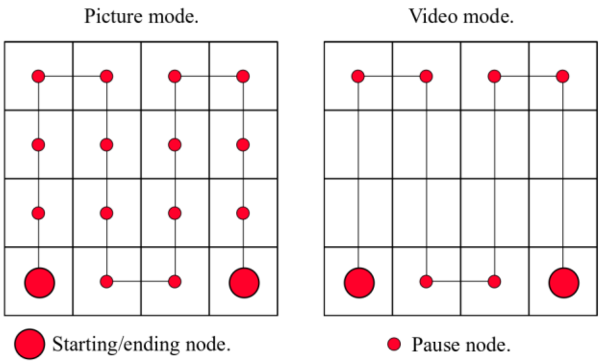

Fig. 2 Trajectories for each of the two procedures.

### E. Textures

Three textures were considered for this study. The first two are made from smooth concrete, the first one being in an indoor environment and the second one in an outdoors one. The third one is made from rough stone slabs that have various colors. The stone slab floor presents the additional challenge of being less planar than the smooth concrete ones.

The indoor concrete environment was illuminated by halogen lights, whereas the outdoor stone slab floor was naturally illuminated. In the stone floor, the pictures were taken under a shadow that allowed for a gradient of light intensity across the image (depicted in Fig. 3), making it an even more challenging task.

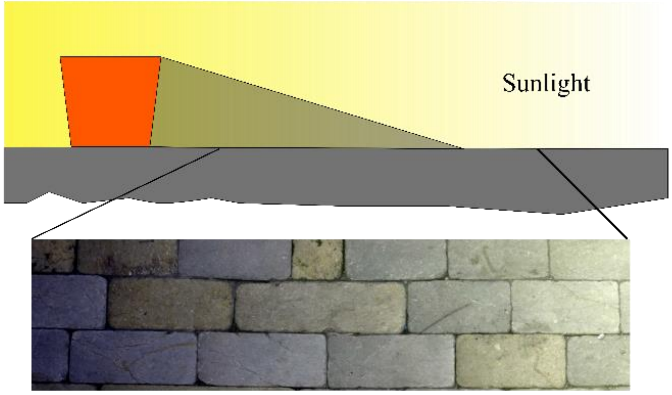

Fig. 3 Lighting conditions for the stone floor.

### F. Panoramas

The different panoramas produced are showcased in Table I. In it, the parameters of each image are also registered. These are the real world dimensions covered by the image, the resolution, the texture on which it was generated, the lighting conditions and the procedure used. The area covered in each of the terrains was chosen according to the available space at the moment of obtaining the pictures/videos.

Data were collected with each of the two procedures for every terrain. It was not possible however to reconstruct a panorama for each set of images obtained, the reasons why this might have been the case are discussed further below.

TABLE I
PANORAMAS

| Name | Dimensions | Texture | Procedure |
|---|---|---|---|
| SI | 24x24 cm | Indoor C.* | Images |
| SO-Im | 64x48 cm | Outdoor C. | Images |
| SO-Vid | 64x48 cm | Outdoor C. | Video |
| SS | 120x60 cm | Stone slabs | Video |

*Here C. stands for concrete.

## IV. Results

Results are divided into each of the different terrain textures. Only for the smooth concrete texture located outdoors, two successful panoramas were obtained. In the indoor concrete floor, the video data failed to successfully recreate the panorama as the resulting image was stitched incorrectly, as features such as cracks were not properly joined. In the outdoor stone slabs, the program was not able to produce a panorama whatsoever from the input pictures.

*A. Smooth indoor concrete.*

The resulting panorama (**SI**) for the smooth concrete floor in the indoor environment is shown in Fig. 4. In it, artificial features are introduced by the reflection of the halogen lamps on the floor.

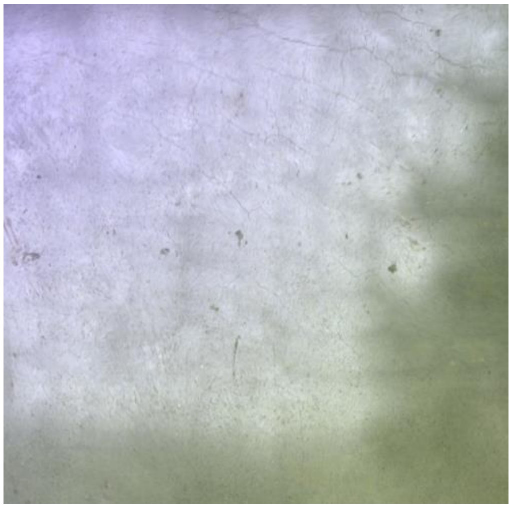

Fig. 4 Panorama SI.

Depending on the location of the robot at a certain point in the trajectory, these reflections might occupy a different part of the image, with varying results on the resulting picture.

Fig. 5 shows the field of view of the robot in one of the used pictures with a reflection present in it. As can be seen, the reflection is not symmetric, which means the reflection is dependent on the orientation of the robot.

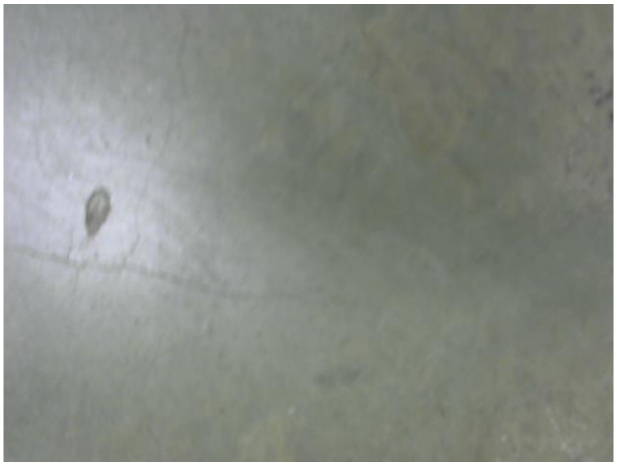

Fig. 5 One of the pictures of panorama SI with a reflection present.

*B. Smooth outdoor concrete.*

Panorama **SO-Im** is presented in Fig. 6. The absence of reflections from the source of light makes the resulting panorama, one without artificial features or artifacts.

It is important to notice that the quality of the pictures taken for this panorama is the same as that in Fig. 5, making the reflections on the floor the only cause of the lower quality panorama obtained in the first texture.

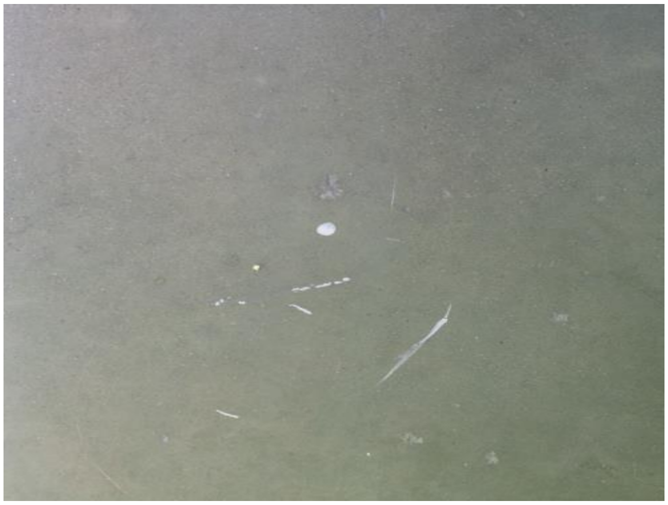

Fig. 6 Panorama SO-Im.

The panorama in video mode for this type of texture, **SO-Vid**, is showcased in Fig. 7. The areas were the pictures were taken for each panorama differ slightly in location.

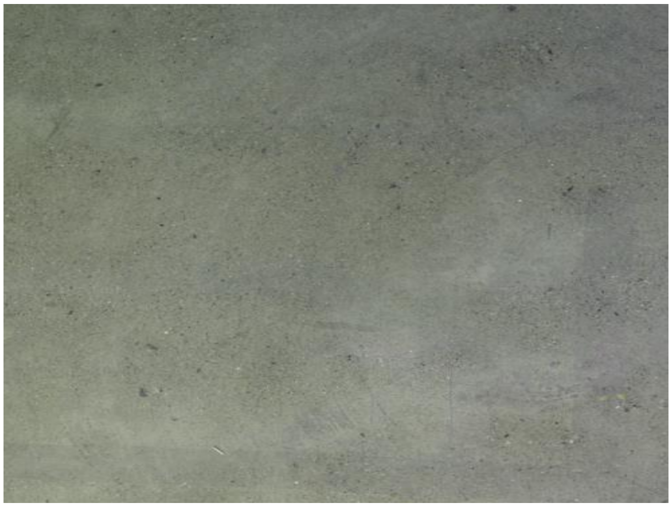

Fig. 7 Panorama SO-Vid.

*C. Stone slabs.*

For the stone slabs the resulting panorama **SS** is shown in Fig. 8. In it, the collinearity of the slabs is clearly preserved, as well as the perpendicularity of the lines in between slabs.

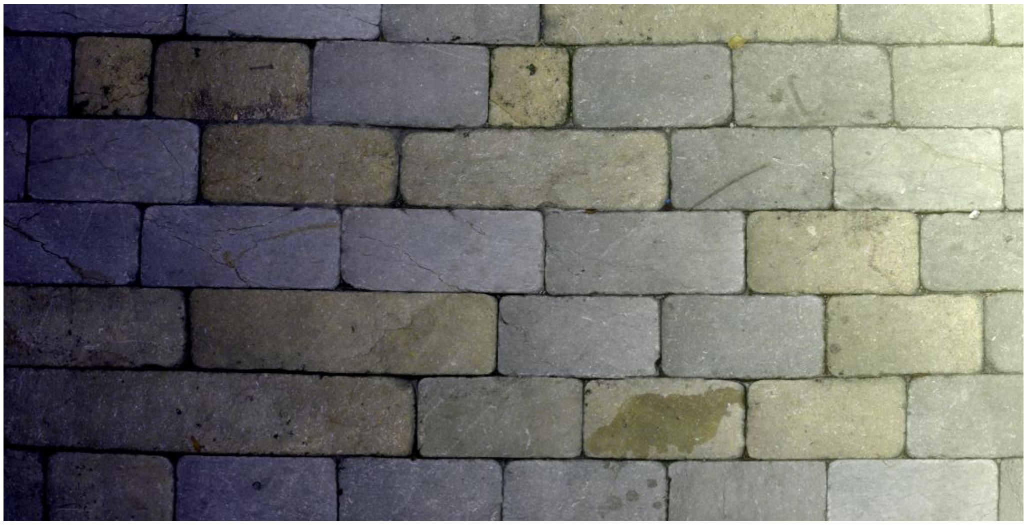

Fig. 8 SS Panorama.

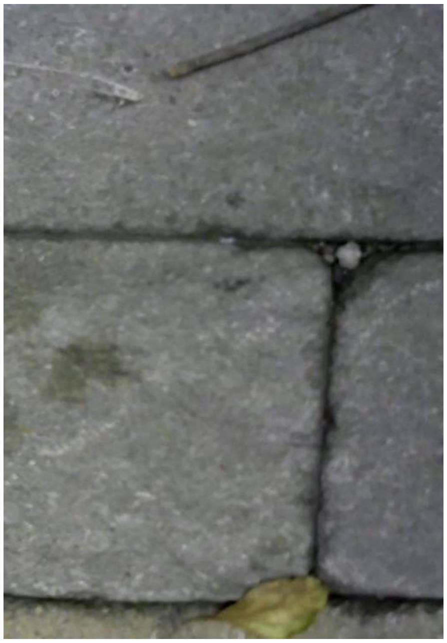

Fig. 9 Frame of the video used to produce panorama SS.

Parallel lines get warped by a small distance in the horizontal direction.

One frame of the video used to produce the Stone Slab panorama is shown in Fig. 9. Here, it is possible to see the low quality of the input picture as well as some of the motion blur caused by the vibration of the camera. This frame was selected at random and corresponds to a location in the upper right part of the panorama in Fig. 8.

## V. Discussion

From the panorama achieved in the first environment (**SI**), we can see the relevance of the floor reflectivity, together with the lighting disposition, in the goal of mapping the ground of an environment. The lack of proper control of these reflections can lead to artifacts in the results that are not present in the terrain itself, which could in turn generate features that an algorithm cannot rely on when tested in the field.

However, this issue can be alleviated by creating an enclosure for the camera, in a way that guarantees the reduction of the reflections on the floor. Nevertheless, it would be useful to perform further inspection of the way these reflections are generated in order to mimic them synthetically. When training an algorithm for a certain target environment, data augmentation with simulated lighting conditions would benefit the performance.

Furthermore, if we look at the second set of images (panoramas SO Im and Vid), it is clear how the quality of the resulting panorama can drastically increase for the same ground texture when lighting conditions are accounted for. However, these might not always represent a real-life scenario.

Regarding the image obtaining procedure, the fact that for the second texture both modes (video and picture) were successful indicates that reflections do not allow for proper data capturing when in video mode. Further experimentation is needed in order to support this claim.

For the third texture, we could only obtain a panorama for video mode. A varying source of light in time might be the reason why picture mode failed, as images in this mode take considerably more time to be generated.

In panorama SS, the software parameters for the generation played a key role in the quality of it. We made the rest of the panoramas assuming planar motion, but this led to wrong results in this texture. As shown in Fig. 10, choosing planar motion led to an unsatisfactory result. To solve this problem, we used the rotary motion option and then projected the resulting stitched panorama using orthographic projection.

From all the parameters tested, the lighting had the most impact in the obtained results, with different motion blur (higher for video mode and stone slab floor) and textures playing a minor role in the pictures.

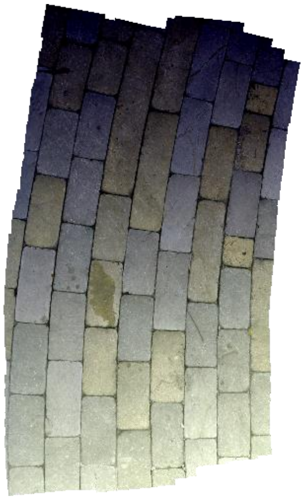

Fig. 10 Failed version of panorama SS. This version assumes planar motion.

Both capturing procedures generated satisfactory panoramas in more than one texture, making them a useful tool. As a recommended practice, video mode should be used whenever possible in order to reduce the capturing time, thus reducing the variation of lighting conditions in a given panorama.

Finally, in order to reduce motion blur when taking the pictures, the use of a stabilizing device such as a Gimbal should be considered together with a higher rigidity mount for the camera.

## VI. Conclusions

We present an analysis of the different parameters that are involved in the mapping of a ground plane with the purpose of creating datasets for further use in the training of neural networks.

We test two ways of producing the panoramas, one with a set of pictures taken at different strategic locations and another one from a stream of images or video. We find that the most convenient one is the video, but cannot be used in all cases.

We also find that the presence of reflections in the ground texture makes the resulting panorama not suitable for training, as it produces artificial features in the ground which are not reliable for real environments.

We also plan on making the robot less prone to vibrations by improving the transmission system and using a Gimbal to stabilize the camera mount.

With these findings, we plan on constructing a dataset of multiple textures on bigger terrains. With this data, we will propose new Ground-based Visual SLAM algorithms with the use of convolutional neural networks (CNN).

We will also apply the approach followed here to determine which parameters influence the most in data collection for Ceiling Visual SLAM, with the goal of helping other researchers develop novel algorithms to achieve better localization.

## References

[1] H. Fang, M. Yang, R. Yang y C. Wang, «Ground-texture-based localization for intelligent vehicles,» *IEEE Transactions on Intelligent Transportation Systems,* vol. 10, pp. 463-468, 2009.

[2] O. Wulf, A. Nuchter, J. Hertzberg y B. Wagner, «Ground truth evaluation of large urban 6D SLAM,» from *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2007.

[3] X. Chen, A. S. Vempati y P. Beardsley, «StreetMap-Mapping and Localization on Ground Planes using a Downward Facing Camera,» from *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018.

[4] K. Tateno, F. Tombari, I. Laina y N. Navab, «CNN-SLAM: Real-time dense monocular SLAM with learned depth prediction,» from *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 2017.

[5] X. Wang, «Autonomous Mobile Robot Visual SLAM Based on Improved CNN Method,» from *IOP Conference Series: Materials Science and Engineering*, 2018.

[6] H. Luo, Y. Gao, Y. Wu, C. Liao, X. Yang y K. T. Cheng, «Real-Time Dense Monocular SLAM with Online Adapted Depth Prediction Network,» *IEEE Transactions on Multimedia,* 2019.

[7] B. S. Timofeev, N. A. Obukhova y A. A. Motyko, «The method of video panorama construction from low detail source videos,» from *2014 IEEE Fourth International Conference on Consumer Electronics Berlin (ICCE-Berlin)*, 2014.

[8] S. Dawn, A. Khera, N. Agarwal y A. Arora, «Panorama Generation from a Video,» de *2018 5th IEEE Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON)*, 2018.

[9] V. C. S. Chew y F.-L. Lian, «Panorama stitching using overlap area weighted image plane projection and dynamic programming for visual localization,» from *2012 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, 2012.

[10] L. Juan y G. Oubong, «SURF applied in panorama image stitching,» from *2010 2nd international conference on image processing theory, tools and applications*, 2010.

[11] S. Lee, S. J. Lee, J. Park y H. J. Kim, «Exposure correction and image blending for planar panorama stitching,» from *2016 16th International Conference on Control, Automation and Systems (ICCAS)*, 2016.

[12] A. Geiger, P. Lenz, C. Stiller y R. Urtasun, «Vision meets robotics: The KITTI dataset,» *The International Journal of Robotics Research,* vol. 32, pp. 1231-1237, 2013.

[13] J. Sturm, N. Engelhard, F. Endres, W. Burgard y D. Cremers, «A benchmark for the evaluation of RGB-D SLAM systems,» from *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012.

[14] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. W. Achtelik y R. Siegwart, «The EuRoC micro aerial vehicle datasets,» *The International Journal of Robotics Research,* vol. 35, pp. 1157-1163, 2016.