# Distributed Visual-Based Ground Truth System For Mobile Robotics

Sergey Sorokumov, Sergey Glazunov
Saint Petersburg Electrotechnical University "LETI"
Saint Petersburg, Russia
svsorokumov, sagluzunov@stud.etu.ru

Konstantin Chaika
Constructor University
Bremen, Germany
chaika.konstantin.v@gmail.com

***Abstract*—This article analyzes ground truth systems for determining the position of a mobile robot in space. The aim of this work is to detect the location of an object based on data from the camera. The article presents ground truth, a system based on computer vision algorithms that process the video stream from cameras located above the polygon where the robot moves. The developed system was tested and achieved an average error of 15mm at a processing speed of 15-17 frames per second. These results turned out to be no worse than analogues at a lower cost of the entire system.**

## I. Introduction

Autonomous transport control systems need constant development and improvement. An important and time-consuming task for humans is to determine the position of transport and other objects in space at any moment in time. In this regard, there is a need to automate the localization process. The object of the research of this article is ground truth systems. The aim of the work is to develop a low-cost distributed system of visual localization of mobile robots, which will be able to work in online mode with an accuracy of up to 2 cm. To achieve this goal it is planned to solve the following tasks: To conduct a review of existing analogues. To implement a module for the localization of mobile robots. To realize the module of visualization of the received coordinates. To test the implemented system.

## II. Overview of the subject area

The systems that determine the position of the tested object in space and visualize the obtained information were chosen as analogues.

### A. Analogs

*1) Analog 1. OptiTrack:* The system was created to efficiently create an image of the virtual world [1]. The system has an accuracy of up to 0.1 mm. The main disadvantage is its high cost. For example, the Camera Primex 41 costs $6,500. The system supports online processing. One of the features of this system is the number of cameras can vary depending on the desired result, and that all cameras are focused on one area. The maximum distance of 30m. This system is used for cinematography, motion analysis, VR, robotics and animation [2].

*2) Analog 2. 3D Random Occlusion and Multi-Layer Projection for Deep Multi-Camera Pedestrian Localization:* This system uses the CNN neural solution [3], which receives as input not an image from a single camera, but multiple images from different cameras located at different coordinates, but directed at the same area. The system is used for multiview pedestrian detection. The CNN task based on a set of images from different cameras is to analyze the height of objects and compare one object in one image with another object in another image. The article does not specify the characteristics of the equipment or the area covered by the system. [4].

*3) Analog 3. Duckietown-autolab-localization:* Marker detection takes place on stationary camera towers located throughout the test site. The graph is also optimized, thanks to the G2O library, and the odometry is determined by working with encoders located inside the object engine [5].

The system consists of several parts: A module responsible for detecting markers in the image, which is located at a certain fixed location. The purpose of this module is to detect markers that are in a predetermined location, and to determine the location of the robot marker if it appears in the camera's field of view. The coordinates are then sent to the module responsible for detecting markers in global coordinates. The module responsible for the graph, which is optimized using the G2O library. The data from the above module, as well as the data responsible for the robot's odometry, is used as input. The purpose of this module. The odometry comes from the encoders installed on the robots. The result of this system is the path traveled by the robot on the polygon with an accuracy of 0.1 cm. One disadvantage of this solution is that one camera covers an area of 0.5 m2. The price of each element, which detects markers and robots consists of the price of RPi4 and RPi Camera (1) +- 400$ [6].

### B. Criterions

*1) Criterion 1. Online processing capability:* Many tasks require an online update of the object's trajectory/tracking. Let us define what online means in this case. A system is called online if the update delay is less than 2 second.

*2) Criterion 2. Accuracy.:* One of the main parameters of any ground-truth system is the accuracy of determining the coordinates of the object in space. For different tasks the accuracy requirements may vary. For example, in the tasks

TABLE I. COMPARISON TABLE OF ANALOGUES

| Criterion | OptiTrack | Duckietown | Our |
|---|---|---|---|
| Online processing | + | - | + |
| Accuracy, mm | **0.1** | 3 | 15 |
| Cost per square meter, $ | **700** | 215 | 88 |

"+" - Meets the requirements of the criterion. "-" - Does not meet the requirements of the criterion.

of human motion tracking the accuracy should be maximum, since even a small error can lead to large conflicts. While for some tasks, in which the coordinates of the robot in the room are determined, an accuracy of a couple of centimeters is sufficient.

*3) Criterion 3. Cost per square meter.:* One of the main criteria for choosing any equipment is its cost.

### C. Conclusions based on the results of the comparison

The Optitrack system has the highest accuracy among the reviewed analogs. The ability to use this system in online mode allows it to be used in many scenarios, but its disadvantage is the high cost compared to other analogues. Our solution allows you to get accuracy of 15 mm, with minimal monetary costs. For tasks where there is no need for 0.1mm accuracy, but the ability to work online is critical, our solution is suitable.

## III. CHOOSING A SOLUTION METHOD

The tool being developed should be presented as a console application.

As a result of the review of analogues, it was found that existing analogues have a number of disadvantages. Because of this, none of them will be able to be used for a distributed visual localization system. Based on the review, the requirements that the tool under development should have were formulated:

1) The solution should detect markers on the polygon and on the robot body to localize the position.
2) The solution should calculate the robot's position during the experiment from the camera's video stream.
3) The solution should be able to work with both one camera and several, without losing the accuracy of calculations.

## IV. DESCRIPTION OF THE SYSTEM

Our polygon was divided into 6 zones, above each of which there is a camera, an example is shown in Fig. 1. Each of the zones was marked with Apriltag markers of the tag36h11 family [7]. The markers were squared along the outer borders of the markers. The coordinates of the lower-left corner in the polygon coordinate system were determined for each marker. The coordinate (0, 0) of the polygon is considered to be the lower left edge.

Fig. 1. Polygon collage made up of images from cameras

The developed system consists of modules:

1) Removing distortion from the image.
2) Changing the perspective of the image.
3) Detection of markers in the field of view of the camera.
4) Transformation of coordinates from the camera coordinate system to the polygon system.
5) Visualization.

### A. Camera calibration

To remove distortion and change perspective, you need to know the internal parameters of the camera. These parameters will be unique for each camera, even if the camera models are identical. The camera parameters consist of two matrices:

1) The camera matrix is a 3x3 matrix that stores the focus projections on the x and y axes and the intersection point of the optical axis with the image plane [8].
2) The distortion matrix is a 5x1 vector that stores the image distortion coefficients [9].

Calibration takes place using a chessboard and the OpenCV library.

### B. Changing perspective

All cameras of the system are located at different angles relative to the ground. With this location, the marker coordinates will be determined incorrectly. To correctly determine the coordinates of the marker, it is necessary that the view from all cameras is strictly at an angle of 90 degrees relative to the ground. To solve this problem, a perspective transformation was used for each camera of the system, that is, the projection of the image onto a new plane. To do this, apriltag markers

are used, which are squared in the area of each camera. To calculate the transformation matrices, the pixel coordinates of the marker data from the camera image are used [10]. Using the transformation matrix data, you can transform the coordinates of objects for each camera.

### C. Robot detection

To detect the robot, an AprilTag marker was attached to its side. The marker is searched on the raw image from the camera, after which matrix operations are applied to the found marker coordinates to eliminate distortions and change the perspective. This approach was chosen because of the speed of work, it gives an increase of 10-12 frames per second, compared with the primary image processing (distortion elimination, perspective change).

TABLE II. SYSTEM ERROR FROM IMAGE RESOLUTION

| resolution/error | x, m | y, m |
|---|---|---|
| (3480x2160) | 0.0043 | 0.0137 |
| (3200x1800) | 0.0046 | 0.0150 |
| (2560x1440) | 0.0050 | 0.0152 |
| (1920x1080) | 0.0047 | 0.0149 |
| (1366x768) | 0.0049 | 0.0157 |
| (1280x720) | 0.0047 | 0.0155 |
| (1024x576) | 0.0045 | 0.0158 |

### D. Coordinate transformation and visualization

To calculate the real coordinates of the robot, the transition from the camera coordinate system to the polygon coordinate system was used. Communication between coordinate systems occurs by marked markers in the area of each camera. Each marker uses the lower left corner, the coordinates in the polygon system were calculated manually, the coordinates in the camera system were calculated using the marker detector. Processing modules calculate the real coordinates of the robot in parallel and independently of each other, after which a message with coordinates is sent to the visualization module. The obtained coordinates are visualized as points on top of a pre-prepared map. An example is shown in Fig. 2.

### E. Tests

To select the optimal system settings, such as the resolution of the video stream, the size of the apriltag marker, the following tests were carried out:

1) The error of calculating the coordinates of the robot was measured depending on the resolution of the video stream Table II.

2) The probability of detecting a marker of a certain size depends on the resolution of the video stream Table III.

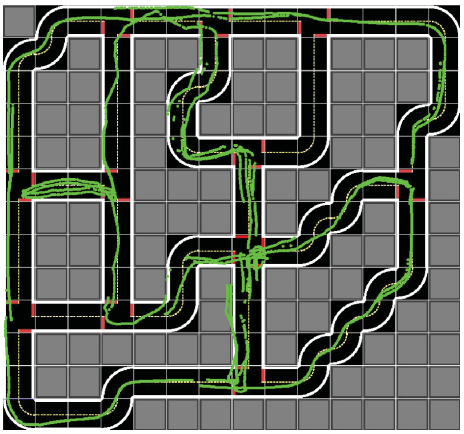

Fig. 2. Example of the system operation

Based on the data obtained, the optimal parameters of the system will be the lowest image resolution and the highest marker resolution. If the image resolution is low, processing by the system will take minimal time relative to other resolutions. But such parameters lead to technical problems due to the fact that the size of the marker will exceed 3 times the size of the robot.

TABLE III. THE PROBABILITY OF DETECTING A MARKER DEPENDS ON THE SIZE OF THE MARKER AND THE RESOLUTION OF THE IMAGE

| resolution/size | 16cm | 13cm | 9cm | 6.5cm | 5.5cm |
|---|---|---|---|---|---|
| (3480x2160) | 1 | 1 | 1 | 1 | 0.85 |
| (3200x1800) | 1 | 1 | 1 | 1 | 0.85 |
| (2560x1440) | 1 | 1 | 1 | 1 | 0.85 |
| (1920x1080) | 1 | 1 | 1 | 0.81 | 0.74 |
| (1366x768) | 1 | 0.928 | 0.8 | 0.68 | 0.53 |
| (1280x720) | 1 | 0.856 | 0.768 | 0.616 | 0.45 |
| (1024x576) | 1 | 0.8 | 0.632 | 0.42 | 0.36 |

## V. CONCLUSION

In this paper a study of existing approaches to ground truth of mobile robots was conducted. As a result of the study, it was found that none of the systems visualizes the position of the object under test during the experiment.

A distributed ground truth system of mobile robots was implemented and tested. The average error value is 15 mm. This system, unlike analogues, visualizes the position of the object under test throughout the experiment. The processing speed is 15-17 frames per second.

This system can be used in schools or universities to conduct classes or research on the topic of driving vehicles due to the low cost of equipment and acceptable error compared to analogues. The system can also be used by companies where accuracy up to a millimeter is not important, for example, tracking the movement of objects in a warehouse.

In future work on the system, it is planned to add metrics: the time spent inside the lane, the position of the robot relative to the lane, the estimation of the trajectory of the robot's rotation at the intersection relative to the optimal trajectory, the average speed of the mobile robot, with which it will be possible to assess the accuracy of autonomous control systems.

## REFERENCES


[1] J. S. Furtado, H. H. Liu, G. Lai, H. Lacheray, and J. Desouza-Coelho, "Comparative analysis of optitrack motion capture systems," in *Advances in Motion Sensing and Control for Robotic Applications: Selected Papers from the Symposium on Mechatronics, Robotics, and Control (SMRC'18)-CSME International Congress 2018, May 27-30, 2018 Toronto, Canada*. Springer, 2019, pp. 15–31.

[2] E. Candela, L. Parada, L. Marques, T.-A. Georgescu, Y. Demiris, and P. Angeloudis, "Transferring multi-agent reinforcement learning policies for autonomous driving using sim-to-real," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 8814–8820.

[3] L. O. Chua and T. Roska, "The cnn paradigm," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 40, no. 3, pp. 147–156, 1993.

[4] S. D. et al., "Marker localization with a multi-camera system," *IEEE*, pp. 135–139, 2013.

[5] G. Grisetti, R. Kümmerle, H. Strasdat, and K. Konolige, "g2o: A general framework for (hyper) graph optimization," in *Proceedings of the IEEE international conference on robotics and automation (ICRA), Shanghai, China*, 2011, pp. 9–13.

[6] G. B. Jacopo Tani, Andrea F. Daniele, "Duckietown localization," 2020.

[7] M. Kalaitzakis, B. Cain, S. Carroll, A. Ambrosi, C. Whitehead, and N. Vitzilaios, "Fiducial markers for pose estimation: Overview, applications and experimental comparison of the artag, apriltag, aruco and stag markers," *Journal of Intelligent & Robotic Systems*, vol. 101, pp. 1–26, 2021.

[8] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[9] Y. Wang, Y. Li, and J. Zheng, "A camera calibration technique based on opencv," pp. 403–406, 2010.

[10] G. Bradski and A. Kaehler, *Learning OpenCV: Computer vision with the OpenCV library*. " O'Reilly Media, Inc.", 2008.