Evaluation and Performance Comparison of Slam Algorithms

Doruk Arslan
Ankara, Turkey
doruk.arslan.dev@gmail.com
December 2022

***Abstract*—The purpose of the research topic found in this paper is to compare the performance of various slam algorithms operating in different environments. In this process, test videos taken to test different environmental conditions will be combined with sensor data and various data sets will be created. Slam algorithms related to the created data sets will be run and then the results of the algorithms in the tested environments will be compared. During the comparisons, the execution speeds and the accuracy of the algorithms will be considered, the results will be visualized and finally, the behavior of the algorithms will be analyzed depending on the tested conditions.**

I. Introduction

SLAM (Simultaneous localization and mapping) algorithms, which will be mentioned throughout the research, have not only remained a technical research topic but have taken their place as a game changer in today's world, especially in the field of robotics, thanks to the technological needs emerging every day and the rapid developments in Computer Science. The primary goal of the aforementioned algorithms is to track the location inside any area depending on the movement direction of the device used and then map the analyzed area in the most optimal way. As a result of this process, users, or devices working with autonomous systems such as UAVs and robots can map any desired region in a very short time, and consequently, understand and analyze the environment they exist.

As with most algorithms, there are multiple methods and approaches that exist in the use and implementation of SLAM algorithms, which are used extensively in augmented reality applications. In this context, the most common approaches are LIDAR and VSLAM, considering current algorithms and their uses. In VSLAM (Visual Slam) applications, while performing the analysis using images obtained from various devices or sensors with cameras, the Lidar Slam (Light detection and ranging) approach uses a distance or laser sensor for this process. As can be predicted, since the mapping process to be performed in Visual Slam applications uses data from a single camera as a source, it will be insufficient to determine the depth and various values of the environment and will require external sensors (e.g., IMU) capable of measuring physical values for more accurate results. Similarly, in the Lidar Slam approach, the data coming from the laser or distance sensor as a source will be interpreted, the coordinates will be assigned to the points, and then a cumulative point cloud will be obtained from them to complete the mapping process. However, since these created points contain only the assigned coordinate information and have more sterile data compared to the images that can be used, they will not be able to accurately determine the density of some objects that may exist, and in some cases, this approach will not be able to produce the desired result. Therefore, it would be misleading to describe any of the different approaches in SLAM algorithms as the most effective or most successful, each approach may have advantages and disadvantages over other approaches in different environments and conditions.

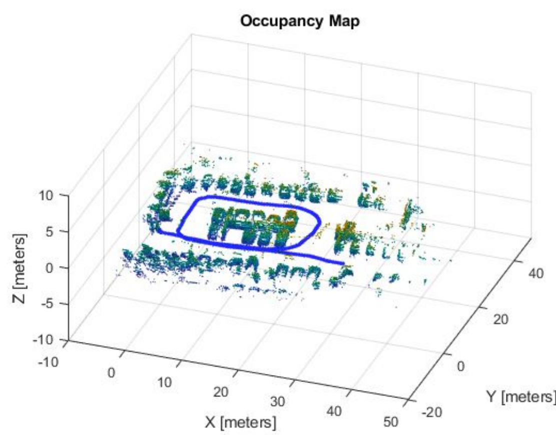

Fig. 1. Sample slam application with 3D Lidar usage [1]

Moreover, the performance of SLAM algorithms and the reliability of their output will be linked to the environment in which the process takes place. Weather conditions at the time of the test, light intensity in the environment, or different objects in the environment will affect the accuracy of the results to be produced. Similarly, as mentioned, it is of great importance that the data come from the source appropriately so that the algorithms can analyze and map the environment most effectively. While scanning the area to be mapped, factors such as the device sensor being cloudy, or moving the sensor too quickly will affect the consistency of the outputs and lead to undesirable results. For this reason, when comparing SLAM algorithms, it is important to consider the conditions of the situation to be tested and accordingly choose the right dataset for the test to yield relevant results. Therefore, existing datasets with various difficulty levels will be used in the algorithm comparison to be made throughout the paper, as well as newly prepared datasets that were created from scratch to test some specific situations and evaluate the performance of algorithms under these conditions. Finally, ORBSLAM and DSO algorithms will be used to perform the various number of tests. In this context, algorithms will be run with the help of all datasets to be used (or produced) and their performance under specific situations will be observed. Then the obtained results will be compared, analyzed, and interpreted considering the test conditions.

II. Related Work

A. Algorithms

1) : ORB-SLAM

ORB-SLAM is one of the most frequently used algorithms in the SLAM research literature, which can operate in different environments. One of the most distinctive features of the algorithm is that it works simultaneously on three separate tasks: tracking, local mapping, and loop closing. While maintaining the tracking part, the algorithm compares each frame obtained with the local map and tries to determine the camera location in real-time in the most effective way. Choosing the right keyframe is one of the most essential parts of the algorithm's work. During this process, the algorithm evaluates whether the difference is sufficient by comparing it with the previous keyframe before accepting a newly encountered Key Frame and inserting it into the Local Map, and continuing the process depending on this response. Then, it creates a local map to continue with the local mapping task and optimizes the local map to find the most possible location of the camera. Iterative Closest Point (ICP) and various algorithms are used in the optimization part. Finally, with the help of pose-graph optimization, possible errors or drifts in the map are corrected and loop closure is applied.

Fig. 2. Sample output of ORB-Slam Algorithm using KITTI dataset [2]

2) : Direct Sparse Odometry

Another algorithm that will be used throughout the tests is the Direct Sparse Odometry (DSO) algorithm, which is similarly treated according to the selection and evaluation of the correct keyframe. The working principle of the algorithm is based on the continuous optimization of the photometric error in a different window, which is formed by considering a photometrically arranged model to create the desired image. In this way, the proposed model produces a complete photometric calibration by taking into account the lens vignetting, exposure time, and non-linear response functions. While comparing performance with the ORB SLAM algorithm throughout the tests, the LDSO algorithm, a more advanced and newer feature version of the DSO algorithm, will be used to obtain more fair and accurate results. The algorithm has become capable of loop closure and pose-graph optimization with new features added. LDSO retains the robust properties of the DSO algorithm on the image pixels, while additionally having features like more distinct analysis of edge points and ensuring repeatability in those points.

B. Datasets

As emphasized in the introduction, the correct testing of the algorithms and the effective comparison of the results are directly proportional to the consistency of the datasets to be selected. It is important to test algorithms under different conditions and to observe and interpret their behavior correctly so that the research subject can give accurate results. In other words, since there is no different dataset for each case to be tested, using only existing datasets will be insufficient for the testing process. Therefore, in addition to the existing datasets, new datasets with the conditions to be tested will be created from scratch and used in the performance comparison of algorithms. Although the algorithms that will take place in the test process can work with more than one type of dataset, EuRoC MAV[3] dataset format will be used for the test process. The EuRoC micro aerial vehicle datasets (EuRoC MAV) are one of the most used datasets during SLAM research and its applications. The reason for this is that the data collected with micro aerial vehicles are created with information from both the camera and the inertial sensor and contain high-quality ground truth. Therefore, when creating a new dataset, the EuRoC MAV dataset format will be considered and the image from the camera and the information from the sensors on the device (e.g, IMU, ground truth) will be combined at the time of testing.

III. Proposed Approach

During the research, the performances of the related algorithms in different environmental conditions will be analyzed and compared. Therefore, various steps must be applied throughout the process in order to create the desired environments and perform the tests most accurately.

A. Dataset Creation

For each different environment, datasets must be created from scratch and brought to the appropriate format that the algorithm can use. As mentioned, the datasets to be created will be created in EuRoC MAV[4] format. Therefore, recording the test environment with the camera and creating a dataset only from a source in this format (i.e. mp4, etc.) is not sufficient. In order to create a dataset in EuRoC MAV format, both images that can be recorded with the camera and instant data to be collected with the help of inertial sensors are also needed. To achieve this condition, the image obtained from the camera sensor must be processed correctly and combined with the data obtained from other sensors.

B. Processing the Recorded Video

As explained in the Algorithms section, both algorithms work depending on the selection and interpretation of the correct keyframe. Therefore, it is imperative that the video obtained with the camera sensor be correctly separated into frames. In this way, the video recording, which is a whole piece, can be made into separate frames and each frame can be interpreted one by one. An important factor to consider is the correct naming of each split frame. Since each frame is a separate part of the whole video, it is important to label the frames according to the period in which they belong so that the camera's location is determined correctly at that moment. For this reason, with the help of the algorithm to be used, while the video is divided into separate frames, at the same time, each frame will be labeled as the timestamp they belong to. In addition, when the sample EuRoC MAV datasets presented by the Swiss Federal Institute of Technology Zurich (ETH) are examined, it is observed that the data used is in a specific format. First of all, the resolution of each frame in the dataset used is 752x480. Since pixel density and image quality may deteriorate, detection of corners, recognition, and interpretation of objects may give erroneous results in recordings taken below this resolution. Therefore, since the resolution of the video to be obtained with the camera sensor may be different from the desired resolution, each separated frame will be resized and brought to the desired resolution with the help of an algorithm to be used. In addition, when each of the sample datasets is examined, it is seen that the presented frames are in grayscale format. For the SLAM algorithms to be used to work more effectively and for the objects to be recognized more easily, the obtained frames will also be brought to grayscale format. As the last step, as mentioned, the data coming from the inertial sensor during the test should be associated with the information of the frames in the dataset. Since the device on which the camera sensor is located cannot use both the camera and the inertial sensor at the same time, two different sensors will be used on two different devices during the test and the obtained data will be combined. To achieve this, after each separated frame is resized and converted to grayscale format, the IMU and Ground Truth information obtained by the inertial sensor will be processed. The timestamp information of each frame will be evaluated and combined with the correct inertial data. In this way, the labeling of the frames will be completed and the dataset to be used will be completely converted to the EuRoC MAV format.

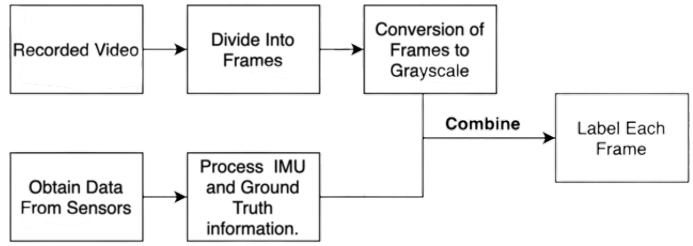

Fig. 3. Dataset creation process.

C. Performance testing of algorithms

After the appropriate datasets are created for the conditions to be tested, the relevant algorithms will be tested with datasets with different difficulty levels. During the tests, considering the output produced by each algorithm for different datasets, it will be graphed and statistically analyzed with the help of the tools we will use. During this analysis, the final resulting Ground Truths will be compared with the expected Ground Truths and their behavior will be evaluated. Similarly, since different datasets to be used will have different difficulty levels depending on the conditions in the environment, unexpected results may be seen during the analysis. Hence, if surprising statistics are encountered while evaluating the results produced by any algorithm, the reason for this behavior will be tried to be explained depending on the conditions of the related dataset. Finally, in the tests to be carried out, not only will the statistically accurate results be evaluated, but also the execution time will be interpreted by analyzing the performance of the algorithm based on its execution time.

IV. Test Datasets

In this section, the objectives and content of the Dataset used during the comparison of the algorithms will be discussed. With each different Dataset created, it is aimed to observe how the algorithms behave in different environments, under different conditions, or at different moments of action. In this way, each algorithm will be run with the same Dataset and ultimately their accuracy and performance will be evaluated by considering their behavior. After the comparison is completed, the results will be evaluated and a final conclusion will be reached.

A. Movement in the Home Environment Dataset

In this Dataset process, the recording was taken at night in a home environment with sufficient artificial light sources. During the recording, one walked into the home environment and a transition was made between the two rooms. In addition, since the hall environment where the test was started contains various items that can be easily differentiated from each other, it is aimed to analyze the edges correctly by the algorithms. This test can be considered simple, as the Dataset does not include challenging examinations.

Fig. 4. Sample image from Dataset A

B. Outdoor Walking Dataset

The relevant test was performed with a simple walking recording in the open field. Since the Dataset contains many different objects that change throughout the motion, it is aimed to test the accuracy of the algorithms. Similarly, at the end of the linear motion, the camera angle was changed rapidly and it was aimed to evaluate how the algorithms behave when faced with sharp camera movements at this stage. Although the direction in the movement process is mostly linear, the direction changed slightly at the moment of sudden camera movement. Therefore, the difficulty level of this Dataset can be considered challenging for algorithms.

Fig. 5. Sample image from Dataset B

C. Outdoor Driving Dataset

Similarly, the related test was also carried out in the open field. The most important difference with the previous test was that the recording during this test was done by driving, not walking. Therefore, the distance difference between the sensor data corresponding to the timestamps in the Dataset is greatly increased. During the test process, it was aimed to evaluate the performance of the algorithm at high speeds. Although the process was carried out at a speed of about 60 km / h, sharp camera movements were avoided. Since a long distance was covered during the test process, many objects that could be detected by algorithms were encountered along the way.

Fig. 6. Sample image from Dataset C

V. Experiment results

Under the heading of Experimental Results, performance comparisons of the two algorithms, whose behavior is observed separately for each dataset, will be made and their accuracy will be evaluated. For this, the output map presented by each algorithm at the end of the test will be considered and the accuracy of the results will be evaluated by comparing it with the IMU visualization obtained by the phone sensor data. Ultimately, a consistent conclusion will be reached by evaluating the results.

A. *Movement in the Home Environment Test (Dataset A)*

In tests on A Dataset, none of the algorithms experienced an error in mapping the environment. Since there are objects of different colors and sizes that can be easily distinguished from each other in the hall environment where the test was started, the encountered objects were analyzed correctly by both algorithms and an accurate and ultimately consistent mapping was achieved. However, since the light source in the balcony environment reached at the end of the test was weaker than the light source in the rest of the test, it was observed that the ORB-SLAM3 algorithm was able to track the environment more successfully than the LDSO algorithm in the low-light environment. In addition, the LDSO algorithm's interpretation and execution speed of the presented dataset was slower than the ORB-SLAM3 algorithm during the test process.

The final mapping obtained from both algorithms at the end of the test and the visualization of the actual movement within the Dataset can be observed from the figures below.

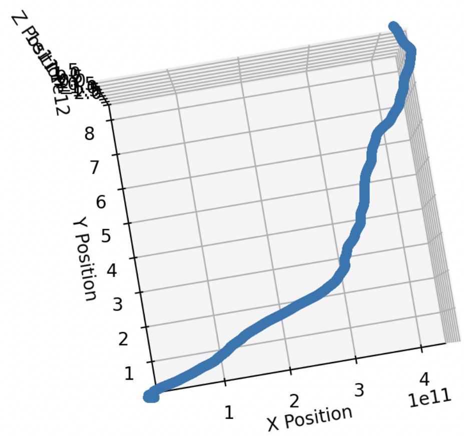

Fig. 7. 3D space view of the movement in the dataset

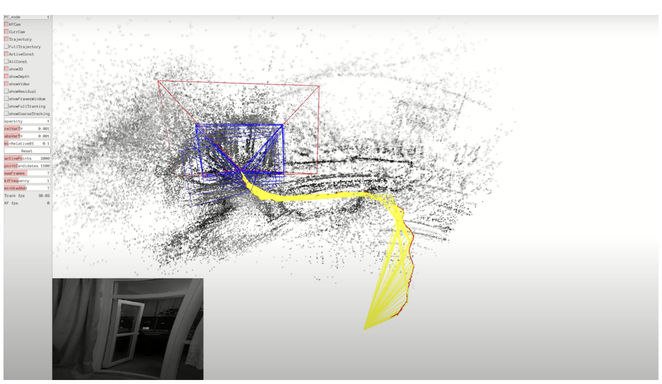

Fig. 8. The result of LDSO for the relevant dataset

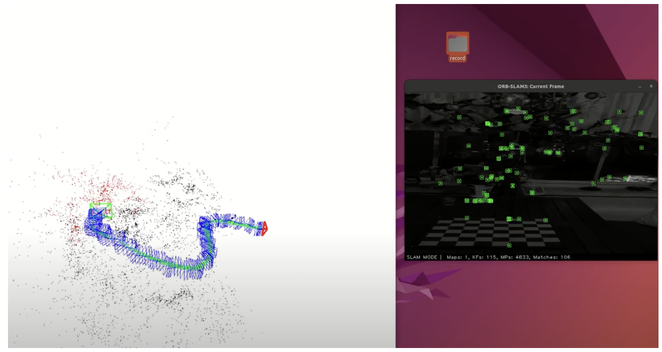

Fig. 9. The result of ORB-SLAM3 for the relevant dataset

B. *Outdoor Walking Test (Dataset B)*

The tests performed on Dataset B are the first tests performed in the open field. Since the conditions outside cannot be controlled manually as in the home environment, the effects of the conditions on the behavior of the algorithms were observed (i.e. amount of light, variety of objects). Moreover, in a part of the test, sharp camera turns were made after the linear motion from the beginning and how the algorithms performed in this case was evaluated.

The linear motion at the beginning of the test was successfully mapped by being interpreted consistently by both algorithms. Objects (e.g. Different types of Cars, Trash, etc.) located next to the camera angle during linear motion are detected much more clearly by the LDSO algorithm. Apart from that, the ORB-SLAM3 algorithm lost mapping tracking after sharp camera rotation after linear movement. Although the testing process continued, it could not re-merge the old track records and stopped the mapping. On the other hand, the LDSO algorithm did not lose track after sharp camera rotation but misinterpreted the mapping. The reason for this is that the color range and density in the frames interpreted by the LDSO algorithm change abruptly after sudden rotations, so it misinterprets the environment it is in and thinks that the dataset has suddenly switched to a different environment in that time period.

Considering the results, it is clear that the LDSO algorithm achieved better results in all respects than ORB-SLAM3 in this testing process, despite a less consistent mapping and slow execution time.

The final mapping obtained from both algorithms at the end of the test and the visualization of the actual movement within the Dataset can be observed from the figures below.

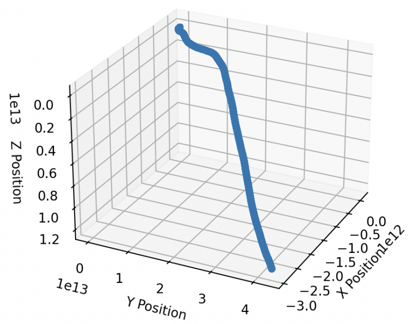

Fig. 10. 3D space view of the movement in the dataset

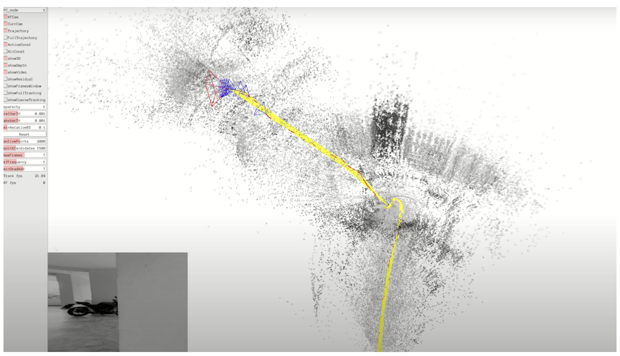

Fig. 11. The result of LDSO for the relevant dataset

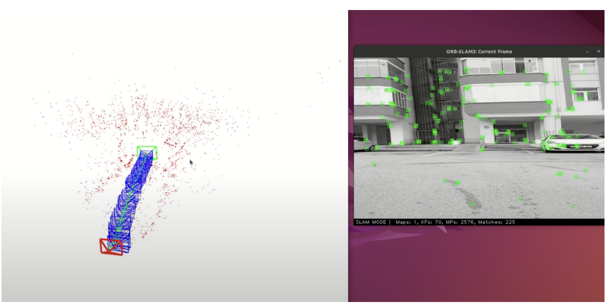

Fig. 12. The result of ORB-SLAM3 for the relevant dataset before failing

C. Outdoor Driving Test (Dataset C)

Although Dataset C included high-speed movements, both algorithms did not encounter major errors as in the test with Dataset B. At the beginning of the test, although the LDSO algorithm evaluated the objects and motion in the frame close to perfect, it could not adequately analyze the environment in the other half of the test. After the wide-angle turn with the car during the test, the algorithm began to misinterpret the environment. The biggest reason for this is that even if the camera rotation is not very sudden, the width of the rotation is misinterpreted by the LDSO and therefore the direction tracked in the mapping is inconsistent. With the last two tests performed, it has been found that the LDSO algorithm interprets the rotations made during recording differently from the actual sensor data (actual movement).

On the other hand, the ORB-SLAM3 algorithm performed better on this test compared to the previous test. Throughout the test, it maintained the same accuracy from start to finish, despite not being able to evaluate subjects in the frame as well as the LDSO algorithm initially did.

Moreover, although ORB-SLAM3 did not get the mapping completely right, it did not make as many errors as the LDSO algorithm did after the rotation. While it did get more accurate results, it did not get the right on-axis depth perception in the mapping process as the LDSO did.

As a result of this test, although both applications did not produce exactly correct results in the relevant test, there are features that both of them made more successful than the other during the testing process. Although the LDSO algorithm made the tracking of the mapping wrong after the wide-angle rotation, it successfully detected the objects and edges at the beginning of the test and correctly created the depth perception on the axes in the map it created. At the same time, although the ORB-Slam3 algorithm did not make the mapping completely accurate, it achieved a much more successful mapping result compared to the final output of the LDSO algorithm and maintained the accuracy of recognizing the environment throughout the test.

The final mapping obtained from both algorithms at the end of the test and the visualization of the actual movement within the Dataset can be observed from the figures below.

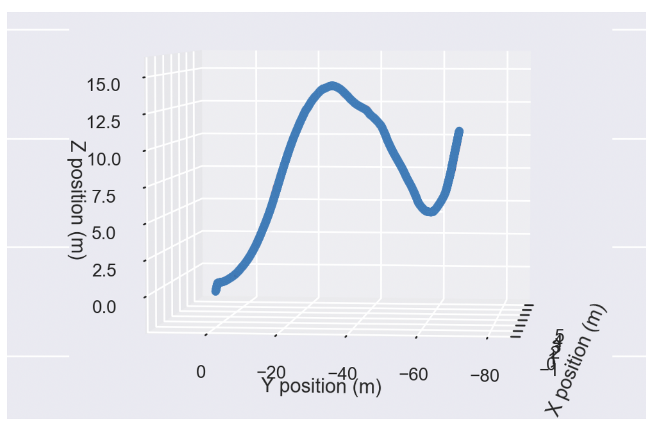

Fig. 13. 3D space view of the movement in the dataset

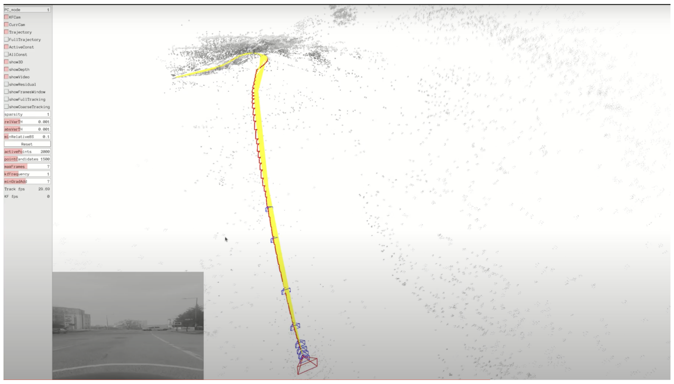

Fig. 14. The result of LDSO for the relevant dataset

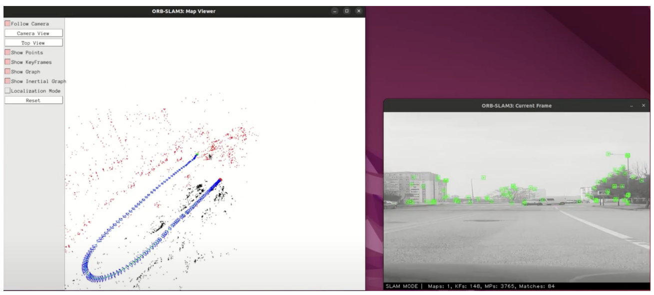

Fig. 15. The result of ORB-SLAM3 for the relevant dataset

VI. Conclusion

Considering all the tests performed, it can be seen that the results produced by LDSO are more consistent than ORB-SLAM3, even if they contain erroneous interpretations or analyses. However, when the test results are evaluated, the environmental conditions in which the test took place and the challenging exams have had a great impact on the behavior of the algorithms. Considering the first test (with Dataset A), it is seen that the LDSO algorithm analyzes the environment much better and makes a near-perfect mapping, but this performance decreases further when it enters an environment where the light source is less. At the end of the test, the behavior of the algorithms against sharp camera rotation was observed with a simple movement (it can be thought of as a small addition to the end of the test). Sudden camera rotations did not have a major impact on the analysis of the LDSO algorithm, while ORB-SLAM3 lost track. With the related test, the effect of the intensity of controlled artificial light in the environment and the sharp camera turns in narrow environments on the performance of the algorithms can be interpreted.

When the second test is evaluated, it is seen that the final map provided by the LDSO algorithm is considerably more consistent. In this external test, the LDSO algorithm correctly analyzed the object it encountered in each frame and interpreted linear motion more successfully than ORB-Slam3. Due to the suddenly changing color intensity after the sharp camera rotation, he interpreted the movement as if he had passed into another environment and could not make the map in the rest of the movement perfectly. However, it is observed that the result produced by the LDSO algorithm is more functional and consistent since ORB-SLAM3 loses tracking directly after a sharp turn. With this test, it was found that ORB-SLAM3 could not continue execution successfully when faced with sharp device movements and failed in this case and that the LDSO algorithm could not correctly analyze the movement and current frames at the time of sharp return. Another important finding is that despite various objects and external conditions in the environment, the LDSO algorithm successfully identified the objects and edges in the environment.

The moment when the ORB-SLAM3 algorithm loses the tracking of the environment during the second test after the sharp camera rotation and the output displayed from the terminal can be observed in the next figure.

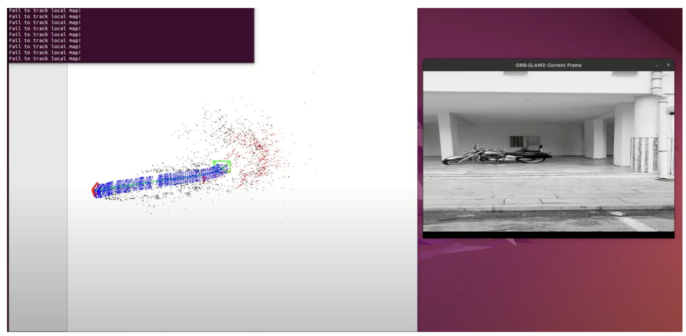

Fig. 16. The moment that ORB-SLAM3 lost track

When the third test was evaluated, considering that the test was recorded while driving and therefore occurred at high speed, both algorithms managed to map and analyze the motion to an acceptable degree. Even though it is seen that the LDSO algorithm is more successful in recognizing and evaluating the environment during the test process, it is seen that after the wide-angle turn with the car (no sharp camera rotation), it misinterprets the movement and reaches a more inconsistent mapping in the direction compared to ORB-SLAM3. Since the test process was carried out with the car, downhill and uphill movements with different altitude values were made throughout the recording. With this information in mind, the final maps provided by the algorithms visualized the depth between the axes of ORB-SLAM3 more consistently.

To sum up, when the final words are said for this research paper when the environmental conditions in which both algorithms are tested are considered, it is seen that they have features that are superior to each other. Along with the tests, it was observed how the algorithms behave in various challenging situations and which algorithm produces more consistent results for the tested feature. Moreover, for every test performed, regardless of the tested features, it is obvious that the ORB-SLAM3 algorithm has a very fast execution speed compared to LDSO. Although it is seen that the LDSO algorithm achieves more consistent results for the three separate tests carried out during the evaluation process, it is obvious that there are still many features that are open to improvement. Similarly, although it was found that ORB-SLAM3 achieved less consistent results with a small difference compared to the LDSO algorithm in the tests, the difference in performance and consistency between the two algorithms is related to the tested feature. Therefore, although the tests presented throughout the paper have found that the LDSO algorithm achieves more consistent results, it would be wrong to say that one of the related algorithms is superior to the other, considering the three separate tests performed during the research and only a few tested features. The answer to the relevant comparison question will vary according to the tested feature and the environmental conditions found.

REFERENCES

[1] J. Engel, V. Koltun, and D. Cremers, "Direct Sparse Odometry," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 40, no. 3, pp. 611-625, Mar. 2018, doi: 10.1109/tami.2017.2658577.

[2] Jakob Engel, Vladlen Koltun, Daniel Cremers: "Direct Sparse Odometry", 2016; arXiv:1607.02565.

[3] AndroSensor.(2013).AndroSensor (Version 1.9.4.4) [Mobile App]. Google Play Store. https://play.google.com/store/apps/details?id=com.fivasim.androsensorhl=trgl=US

[4] Swiss Federal Institute of Technology Zurich, IJRR 2016 MAV Visual Inertial Datasets(2016) https://projects.asl.ethz.ch/datasets/doku.php?id=kmavvisualinertialdatasets

[5] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. M. Montiel and J. D. Tardós, "ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual–Inertial, and Multimap SLAM," in IEEE Transactions on Robotics, vol. 37, no. 6, pp. 1874-1890, Dec. 2021, doi: 10.1109/TRO.2021.3075644.

[6] X. Gao, R. Wang, N. Demmel and D. Cremers, "LDSO: Direct Sparse Odometry with Loop Closure," 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 2018, pp. 2198-2204, doi: 10.1109/IROS.2018.8593376.

[7] Kanishk Vishwakarma, A review on ORB-SLAM-2 paper https://medium.com/@sallyrobotics.blog/a-review-on-orb-slam-2-paper-3554d4fcaa7c

[8] E. Mingachev et al., "Comparison of ROS-Based Monocular Visual SLAM Methods: DSO, LDSO, ORB-SLAM2 and DynaSLAM," Lecture Notes in Computer Science, pp. 222-233, 2020

[9] M. Burri et al., "The EuRoC micro aerial vehicle datasets," The International Journal of Robotics Research, vol. 35, no. 10, pp. 1157-1163, Jan. 2016, doi: 10.1177/0278364915620033.

[10] Raymond, M. (2019, December 5). Exploring data acquisition and trajectory tracking with Android devices and python. Medium.Dec 2, 2019. https://medium.com/analytics-vidhya/exploring-data-acquisition-and-trajectory-tracking-with-android-devices-and-python-9fdef38f25ee