

Mobile robot localization using visual odometry in indoor environments with TurtleBot4

Gurpreet Singh¹, Deepam Goyal¹, Vijay Kumar²

¹Chitkara University Institute of Engineering and Technology, Chitkara University, Punjab, India

²Symbiosis Skills and Professional University, Maharashtra, India

Article Info

Article history:

Received Jan 22, 2024

Revised Jun 28, 2024

Accepted Jul 26, 2024

Keywords:

Laser scanners

Mobile robot localization

Red green blue – depth cameras

TurtleBot

Visual odometry

ABSTRACT

Accurate localization is crucial for mobile robots to navigate autonomously in indoor environments. This article presents a novel visual odometry (VO) approach for localizing a TurtleBot4 mobile robot in indoor settings using only an onboard red green blue – depth (RGB-D) camera. Motivated by the challenges posed by slippery floors and the limitations of traditional wheel odometry, an attempt has been made to develop a reliable, accurate, and low-cost localization solution. The present method extracts oriented FAST and rotated BRIEF (ORB) features for feature extraction and matching using brute-force matching with Hamming distance. The essential matrix is then computed using the 5-point algorithm and decomposed to recover the relative rotation and translation between poses. The absolute pose is obtained by chaining the incremental motions estimated from VO. Through experimentation and comparison with wheel odometry, the findings demonstrate the effectiveness of our VO system, achieving a positional accuracy with minimal error of 4-5%. The article also compares VO with wheel odometry and shows the advantages of using a visual approach, especially in environments with slippery floors where wheel slippage causes large odometry errors. Overall, this work presents an effective VO system for reliable, accurate, and low-cost localization of TurtleBot4 in indoor environments without relying on external infrastructure.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Vijay Kumar

Symbiosis Skills and Professional University

Pune, Maharashtra, India

Email: vijay.jadon@gmail.com

1. INTRODUCTION

Over the last few decades, the field of mobile robotics and autonomous systems has garnered considerable global interest, leading to significant advancements and breakthroughs. Presently, mobile robots demonstrate the capability to independently execute intricate tasks, a departure from the past where human input and interaction were imperative [1]. The applications of mobile robotics span diverse fields, including military, medical, space, entertainment, and domestic appliances. In these applications, mobile robots are anticipated to carry out complex tasks, necessitating navigation in intricate and dynamic indoor and outdoor environments without human intervention [2]. Among various aspects, navigation emerges as a key issue closely tied to the concept of autonomy in mobile robots. Specifically, the capacity for safe navigation within its environment and effective path planning are critical tasks that characterize a mobile robot as an autonomous entity [3]. A key capability required by mobile robots to operate autonomously is self-localization, i.e., the ability to determine their pose (position and orientation) within the environment. Accurate and reliable localization is crucial for navigation, path planning, and other higher-level decision-making.

Various sensors and techniques exist for localizing mobile robots, such as global positioning system (GPS), WiFi, ultra-wide band radios, laser scanners, and cameras. However, many indoor environments lack GPS signals, and installing external infrastructure like landmarks or beacons for localization is cost-prohibitive and labor-intensive. Camera sensors provide a low-cost and infrastructure-free alternative for localization in indoor environments by utilizing visual information [4]. Visual odometry (VO) is one such technique where a robot estimates its motion by examining the changes in camera images over time [5]. It is analogous to wheel odometry, which integrates wheel rotation measurements to estimate the change in pose. However, wheel slippage has no impact on VO because it only uses images from a camera. With recent advances in computer vision and parallel processing hardware, VO has emerged as a promising approach for accurate and robust localization of mobile robots in indoor environments.

In this work, a VO framework for localizing is presented on a widely used mobile robot, TurtleBot4, in indoor environments. TurtleBot4 is equipped with a red green blue – depth (RGB-D) camera, which provides both color and depth information. This VO approach utilizes the RGB images and operates by tracking features across consecutive frames to estimate the incremental motion of the robot. The absolute pose is determined by chaining together these relative motions. The main contributions to this work are: i) an evaluation of VO for TurtleBot4 localization in lab environments using only its on-board RGB-D camera; ii) a comparison between VO and wheel odometry in conditions with wheel slippage; and iii) an open-source implementation of the complete VO framework.

The rest of the article is organized as follows: section 2 provides an overview of related work in VO. Section 3 describes the methodology and the experimental setup. Section 4 analyzes the results and compares VO with wheel odometry. Section 5 concludes the article.

2. RELATED WORK

VO is a key technique for egomotion estimation and localization for autonomous robots and vehicles using cameras. It has been an active research area in mobile robotics for over two decades, with a rich literature exploring various methods and system designs. This section provides an overview of related work in VO. Stein *et al.* [6] presented an approach for egomotion estimation using optical flow to match feature points between consecutive frames. Visual motion estimation from image sequences was formulated as an optimization problem using optical flow constraints. Nistér *et al.* [4] developed a real-time VO system that used Harris corners and normalized correlation for feature matching. They introduced an absolutization step to recover the initial position and scale, which are unobservable from VO alone. Whereas Konolige and Agrawal [5] described a stereo VO system for large outdoor environments using sparse bundle adjustment over keyframes. They also proposed efficient techniques for outlier rejection and reducing computational complexity to achieve real-time performance. Some other VO systems from this early period rely on optical flow to directly estimate egomotion from differences between subsequent images [7], [8]. Optical flow provides dense tracking but can be noisy and difficult to compute reliably.

Few researchers have explored the use of local invariant feature descriptors like scale-invariant feature transform (SIFT) [9] or speeded-up robust features (SURF) [10] for establishing sparse feature correspondences. These feature-based methods offered more robust matching than optical flow techniques. Examples include the work of Clemente *et al.* [11], who used a single handheld camera along with SIFT features and random sample consensus (RANSAC) for egomotion estimation. Pretto *et al.* [12] developed a VO system based on a sparse set of SURF features tracked using the Kanade-Lucas-Tomasi tracker. In recent years, there has been growing interest in using machine learning techniques to improve VO and simultaneous localization and mapping (SLAM) systems. Supervised learning methods leverage large datasets with ground truth to train convolutional neural networks (CNNs) for tasks like feature learning, pose estimation, and loop closure. Wang *et al.* [13] optimized a CNN model to generate keypoint locations, descriptors, and scores that rival traditional handcrafted features like SIFT. DeTone *et al.* [14] used self-supervised learning from videos to train a deep VO model that outperformed classical methods. Some end-to-end learning-based VO frameworks have also emerged. For example, vector of locally aggregated descriptors (VLAD)-VO formulates VO as sequence-to-sequence learning using recurrent CNNs trained on street view datasets with ground truth poses. Costante *et al.* [15] explored self-supervised pose regression and outlier rejection with deep networks showing competitive accuracy. However, such end-to-end techniques require large amounts of training data. They are also computationally intensive compared to classic methods.

Another direction is the use of stereo cameras or RGB-D sensors to provide depth information along with images for improving odometry. Examples include RGB-D SLAM systems like dense VO [16] and elastic fusion [17] that leverage depth data for tracking and mapping. Stereo systems like stereo parallel tracking and mapping (S-PTAM) [18] demonstrate high accuracy by combining stereo matching with visual SLAM. Depth information provides direct scale estimation and improves pose tracking and mapping quality. Some noteworthy VO systems built over the years include parallel tracking and mapping (PTAM) by

Klein and Murray [19], which introduced the separation of tracking and mapping threads. Strasdat *et al.* [20] developed double window optimization for efficient bundle adjustment in monocular SLAM. Fovis by Huang *et al.* [21] uses fisheye cameras for improved field-of-view coverage and robustness. Semi-direct monocular visual odometry (SVO) [22] demonstrates accurate, high-speed odometry by directly using image intensities for motion estimation. Large-scale direct monocular (LSD)-SLAM [23] performs direct alignment of images using photometric error. Oriented FAST and rotated BRIEF (ORB)-SLAM [24] presents a versatile visual SLAM system supporting monocular, stereo, and RGB-D configurations using ORB features with graph-based optimization.

More recently, visual-inertial techniques that fuse cameras and inertial measurement units (IMUs) have become popular. For example, visual-inertial system (VINS)-Mono [6] demonstrates accurate and robust odometry estimation on drones and other platforms using a monocular camera and IMU. Visual-inertial systems take advantage of complementary sensing modalities for improved performance across different environments. While VO has been extensively researched, as highlighted above, relatively little work has evaluated VO specifically targeted for low-cost, widely used robotic platforms like TurtleBot. Most prior VO systems were designed for drones, smartphones, and ground vehicles. The limited onboard computation makes directly deploying popular VO approaches challenging on resource-constrained robots. This motivates work to develop and validate a lightweight yet accurate VO system designed for the TurtleBot robot using its RGB-D sensor.

The proposed approach draws on established techniques like ORB features [25], essential matrix decomposition [26], and pose graph optimization [27]. ORB provides efficient feature extraction and matching suitable for limited computational budgets. The essential matrix allows recovery of the incremental motion between frames. Pose graph optimization improves the global consistency of the VO trajectory. However, these components were adapted and optimized for the TurtleBot platform with a focus on accurate, real-time odometry estimation using only its RGB-D camera. Additionally, some studies also evaluated the performance of a multilayer clustering network for similar problems [28], [29]. In summary, while VO is a mature research field, relatively little work exists on evaluating VO performance specifically targeted for popular consumer robots like TurtleBot. Most VO systems are designed for drones, phones, or cars with more powerful sensors and computational resources. This article aims to fill this gap by developing and validating an efficient VO system for accurate, real-time localization of TurtleBot in indoor environments using its onboard RGB-D camera.

3. METHOD

This section presents the methodology of the VO framework for TurtleBot4. The input to the algorithm is a stream of left RGB images from the RGB-D camera. The output is an estimation of the incremental motion of the robot from frame to frame. By chaining these relative motions, the full trajectory of the robot is obtained in the environment. The VO framework consists of the following key stages: i) feature detection and description; ii) feature matching; iii) motion estimation; and iv) pose graph optimization.

Figure 1 shows an overview of the VO system. Features are detected in consecutive frames and matched using brute-force matching with Hamming distance. The essential matrix is computed using the 5-point algorithm and decomposed to recover the incremental pose. Pose graph optimization improves the odometry trajectory.

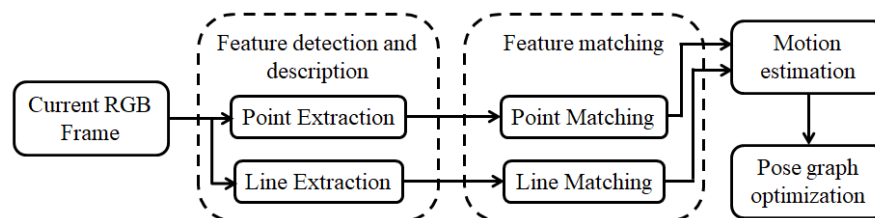


Figure 1. VO framework

3.1. Detection and description

The first step in the VO framework is detecting distinctive image features in each camera frame that can be tracked across frames. Good features for VO should be repeatable across viewpoints, allow precise localization, and be efficient to match. The ORB feature detector and descriptor [24] is used in the present approach due to its balance between accuracy, distinctiveness, and computational efficiency. ORB builds on

FAST keypoint detection [30] and the BRIEF descriptor [31], with additions to improve rotation invariance. FAST detects key points by looking for points that stand out from their surroundings within the image based on a corner response function. ORB improves on this by using a pyramid and FAST at multiple scales to extract around 2000 keypoints per image in a scale-invariant manner. Each keypoint is assigned an orientation based on image gradients to achieve rotation invariance. BRIEF generates a binary descriptor vector by comparing the intensities of points on a smoothed image patch around the keypoint location. ORB's rBRIEF descriptor rotates this patch by the keypoint orientation to obtain rotation-invariant descriptors. The resulting ORB keypoints and descriptors provide a lightweight yet high-quality visual feature representation. Matching descriptors using Hamming distance is also extremely efficient, enabling real-time tracking and motion estimation. This makes ORB highly suitable for computationally constrained platforms like TurtleBot.

ORB is specifically chosen over other features like SIFT and SURF due to its computational efficiency and accuracy trade-off. SIFT descriptors [18] are high-quality but slow to extract and match. SURF [10] accelerates this using box filters but is not as fast as ORB. Learning-based features [13] can outperform handcrafted ones given sufficient training data but require powerful GPUs for inference, which are not available on this robot. Overall, ORB offers the right balance of speed, distinctiveness, and invariance needed for real-time VO on TurtleBot using only its CPU. The ORB feature representation forms the basis for establishing reliable correspondences between frames to estimate incremental motion.

3.2. Feature matching

Distinctive 2D feature points (keypoints) are detected in each image to capture the environment's structure. The ORB feature detector is used to identify key points, which are scale, rotation, and illumination invariant. ORB provides a good balance between feature quality and extraction speed [32]. Around 2000 keypoints are detected per frame. Each keypoint is described using a 256-bit binary string generated from the ORB descriptor. This allows for efficient feature matching using Hamming distance.

3.3. Motion estimation

Once feature correspondences are established between two frames, the next step is to estimate the relative 6-DOF motion between the camera poses. Recovering this incremental camera motion provides the basic odometry information for localizing the robot as it moves frame-by-frame. The essential matrix formulation is utilized for motion estimation. The essential matrix E encapsulates the relative rotation and translation between two camera viewpoints with a small baseline separation. It has the following key properties: i) E is a 3×3 matrix with rank 2 satisfying the constraint: $E = [T_x] R$, where $[T_x]$ is the skew-symmetric cross product matrix of the translation t ; and ii) for noise-free correspondences $a \leftrightarrow a'$, we have the epipolar constraint: $a^T E a' = 0$.

The 5-point algorithm created by Nistér *et al.* [4] can estimate the essential matrix E can be estimated from a set of point correspondences across frames. This technique efficiently computes the exact E using only five keypoint matches. The minimal sample size makes it robust to outliers under RANSAC. The 5-point algorithm is combined with RANSAC to robustly estimate the essential matrix between consecutive frames. The best E is then decomposed using singular value decomposition (SVD) to recover the relative rotation R and translation T up to an unknown scale [26]. The basic odometry estimate between different camera poses is this incremental rotation and translation from the essential matrix. Chaining these frame-to-frame motion estimates yields the full camera trajectory and VO output. The essential matrix approach is efficient, flexible, and works well with monocular cameras. It does not require fully calibrated rigs or additional depth sensors. The 5-point algorithm produces accurate short-baseline odometry suitable for incremental VO estimation from a single moving camera.

3.4. Pose graph optimization

Chaining the incremental motion estimates from VO over time yields an estimate of the full robot's trajectory. However, small errors in incremental motions accumulate into drift over longer sequences. To improve global consistency and reduce drift, pose graph optimization is employed. The idea is to optimize the full robot trajectory by minimizing the reprojection error between matching keypoints observed from multiple poses [12]. This takes into account all constraints between matched features visible across different parts of the trajectory. Specifically, a pose graph is built where nodes are camera poses and edges represent relative pose constraints from VO between frames. The reprojection error across all matches provides a non-linear least squares objective to refine the pose graph. This global optimization distributes odometry errors throughout the graph to obtain a more consistent trajectory. The Levenberg-Marquardt algorithm is used to efficiently solve this non-linear optimization. The optimized pose graph finally provides drift-reduced VO output, combining all incremental motions and global constraints. This improves localization accuracy over long sequences by correcting drift and inconsistencies in the VO trajectory.

The pose optimization stage is key to achieving reliable performance for VO on TurtleBot4 over extended trajectories. While essential matrix decomposition provides accurate frame-to-frame motion estimates, global pose graph optimization leverages all constraints to reduce the accumulation of errors and drift as the robot traverses large distances. This enables accurate VO-based localization over long durations in indoor environments.

3.5. Experimental setup

TurtleBot4, used in the present study, is a low-cost, open-source wheeled robot equipped with a Raspberry Pi 4 and an Intel RealSense D435i RGB-D camera integrated into it. The RGB camera provides 1080 p color images, while the depth camera outputs 640x480 depth images at 30 FPS. Figure 2 shows the experimental setup, including the mobile robot platform as shown in Figure 2(a) and intel RealSense D435i RGB-D camera in Figure 2(b).



Figure 2. Experimental setup (a) mobile robot platform and (b) intel realSense D435i RGB-D camera

Two datasets designated as sequence 1 and sequence 2 were collected with TurtleBot4 driving autonomously in the Robotics and Mechatronics Research Lab (RMR Lab), Chitkara University, Punjab, and the corridor adjoining the RMR Lab, as shown in Figure 3. For ground truth poses, manual markings were marked on the floor tiles and measured. Sequence 1 consists of 1100 frames captured over a 10x12 m area of the lab with chairs and other furniture shown in Figure 3(a). Sequence 2 is 1400 frames long and recorded over a 20-metre traverse in a corridor with plain walls, doors, and windows, as shown in Figure 3(b). Both datasets include color images from the left RGB camera.

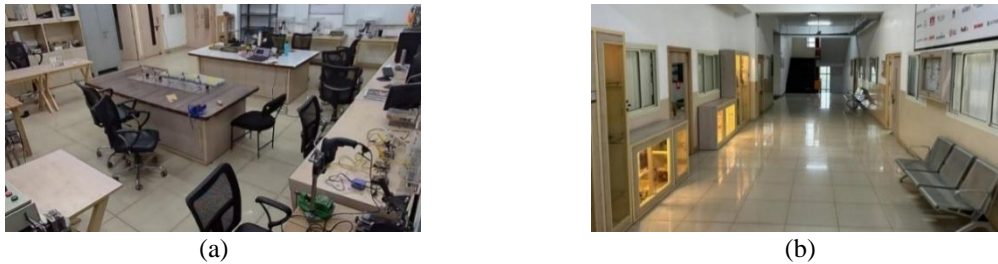


Figure 3. Working environment (a) RMR Lab and (b) corridor adjoining RMR lab

4. RESULTS AND DISCUSSION

This section presents the experimental results to evaluate the performance of VO on the two TurtleBot4 datasets. The VO trajectory is also compared with raw wheel odometry to analyze the benefits of VO. To quantitatively evaluate the VO trajectory, two standard metrics are used: i) translational root mean square error (RMSE) from ground truth; and ii) relative pose error (RPE).

RMSE measures the global consistency of the estimated trajectory, while RPE evaluates local frame-to-frame accuracy. For RPE, the error is computed between estimated incremental motions and ground truth over a window of 5 frames. Additionally, timings are provided for feature detection, matching, and motion estimation modules for analyzing the runtime performance of the VO system. A reliable and good-quality VO dataset is required to get the error metrics, which should be verified with the actual situation. This is critical since it not only examines the performance of the algorithm under consideration but also shows how altering specific parameters might result in various outcomes and error levels. The error may be computed by taking the root mean square of the variations between the expected and real coordinates and using the equation (1):

$$Error = \sqrt{\Delta x^2 + \Delta y^2} \quad (1)$$

Table 1 summarizes the trajectory accuracy of the VO framework in terms of RMSE and RPE metrics for the two sequences. For sequence 1, VO achieves an RMSE of 0.35 m over the entire run, which is only 4.4% of the total distance. This demonstrates accurate and consistent odometry estimation in the lab environment. RPE is 0.18 degree and 0.012 m, showing precise frame-to-frame motion estimates. Whereas on sequence 2, the RMSE of 0.52 m over a 20 m distance corresponds to 5% error. RPE also rises slightly to 0.22 degree and 0.018 m. This is likely because the corridor offers fewer distinguishing features, resulting in some drift accumulation. Overall, VO demonstrates competitive localization accuracy across diverse environments using only visual information.

To highlight the benefits of VO, it is compared against raw wheel odometry computed from the TurtleBot's wheel encoders. This represents odometry estimated by integrating incremental motions from wheel rotations. Table 2 shows wheel odometry performs significantly worse than VO, with 2-3x larger errors. On sequence 1, wheel odometry accumulates 1.5 m of error over the entire run, causing localization failure. Even on sequence 2, wheel odometry shows 1.8 m RMSE compared to 0.52 m for VO. This is because wheel odometry suffers from slippage on the office floors, resulting in incorrect motion estimates. VO does not rely on wheel measurements and is unaffected by wheel slippage. This highlights the robustness of VO for localization.

Table 1. Accuracy of VO on the two datasets

Metric	Sequence 1 (RMR Lab)	Sequence 2 (corridor)
RMSE (m)	0.35	0.52
RPE (deg/m)	0.18/0.012	0.22/0.018

Table 2. Comparison of wheel vs VO

Odometry	Sequence 1 (RMR Lab)	Sequence 2 (corridor)
Wheel	1.5m RMSE	1.8m RMSE
Visual	0.35m RMSE	0.52m RMSE

Videos were captured using the on-board RGB-D camera as the mobile robot moved independently throughout the lab and corridor as shown in Figure 4. The video contained an appropriate quantity of features, as well as favourable weather and lighting circumstances. Figure 4(a) shows the mobile robot moving ahead and taking a slight right turn as the rack on the left side begins to disappear from the camera and the chair enters the frame after the video has been sliced into frames. Similarly, videos were captured and sliced into the frame for the corridor as well. It can be seen from Figure 4(b) that the mobile robot is moving forward in an almost straight line as the door on the right side starts disappearing from the picture and the railing of the stair is in the center of the frames of the particular segment of the captured video.

Figure 5 shows the camera trajectory and ground truth obtained using the experimental results. Figure 5(a) illustrates the observed movement of the mobile robot in the RMR Lab, aligning with the actual camera trajectory. The graph indicates deviations of approximately 0.3 to 0.5 meters in both the x and y directions at various locations throughout the entire travel. In contrast, Figure 5(b) demonstrates the correlation between the monitored movement of the mobile robot and the real camera trajectory in the corridor. The deviation becomes evident as the robot gradually diverges from the camera trajectory, resulting in a disparity of 0.3 meters in the x-axis and approximately 0.5 meters in the y-axis.



Figure 4. Video frame sequence sample of (a) RMR Lab and (b) corridor

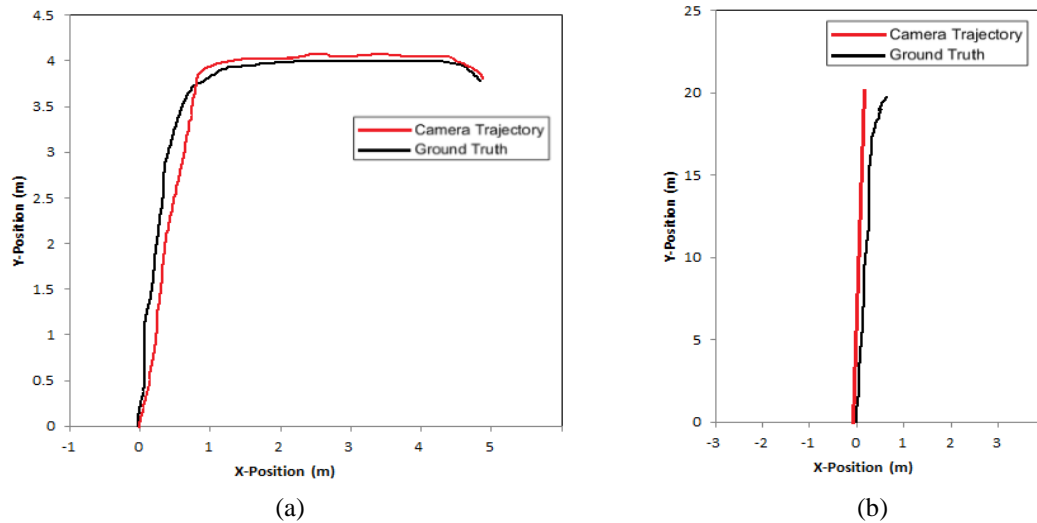


Figure 5. Camera trajectory and ground truth in (a) lab and (b) corridor

Figure 6 shows the feature detection (line) of the RMR Lab environment Figure 6(a) and corridor Figure 6(b) for the single frame of the whole run. Using the estimated motion data supplied by on-board wheel odometry, the features picked in the previous picture are projected into the second image. Following that, a correlation-based search exactly re-establishes the 2D locations in the second picture. The comparison of two successive frames after implementing the VO technique is shown in Figures 7(a) and 7(b) for the lab environment and corridor, respectively.

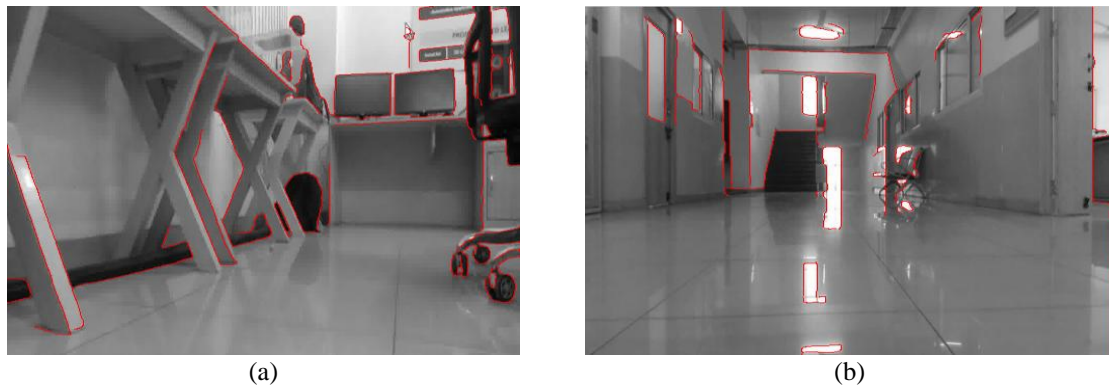


Figure 6. Feature detection (line) in (a) RMR lab and (b) corridor



Figure 7. Comparison of two consecutive frames after implementing the VO algorithm (a) RMR lab and (b) corridor

5. CONCLUSION

This article introduces a VO system designed for localizing the TurtleBot4 mobile robot in indoor environments, utilizing solely its onboard RGB-D camera. The proposed approach involves the detection and matching of ORB features across frames to estimate incremental motion, subsequently chained to recover the full trajectory. The findings obtained from real-world datasets illustrate that the VO system achieves accurate and reliable localization, demonstrating competitiveness with state-of-the-art VO systems. Notably, the visual approach outperforms standard wheel odometry, especially in conditions involving wheel slippage. The presented system offers a cost-effective, infrastructure-free solution for precise indoor localization of TurtleBot4 using visual information. The open-source implementation enables researchers to develop advanced navigation and mapping solutions, building upon a robust VO-based pose estimation. This work contributes to realizing the immense potential of autonomous mobile robots functioning safely in indoor environments.

ACKNOWLEDGEMENTS

The authors express their gratitude to the Robotics and Mechatronics Lab at Chitkara University for providing the necessary experimental setup and essential infrastructure. This research was self-funded and conducted without external funding or grants.





REFERENCES

- [1] K. Wang, S. Ma, J. Chen, F. Ren, and J. Lu, "Approaches, challenges, and applications for deep VO: Toward complicated and emerging areas," *IEEE Transactions on Cognitive and Developmental Systems*, vol. 14, no. 1, pp. 35–49, Mar. 2022, doi: 10.1109/TCDS.2020.3038898.
- [2] A. Zhanabatyrova, C. S. Leite, and Y. Xiao, "Structure from motion-based mapping for autonomous driving: practice and experience," *ACM Transactions on Internet of Things*, vol. 5, no. 1, pp. 1–25, Jan. 2024, doi: 10.1145/3631533.
- [3] X. Lin *et al.*, "A robust keyframe-based visual SLAM for RGB-D cameras in challenging scenarios," *IEEE Access*, vol. 11, pp. 97239–97249, 2023, doi: 10.1109/ACCESS.2023.3312062.
- [4] D. Nistér, O. Naroditsky, and J. Bergen, "Visual odometry," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2004, vol. 1, doi: 10.1109/cvpr.2004.1315094.
- [5] K. Konolige and M. Agrawal, "FrameSLAM: From bundle adjustment to real-time visual mapping," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 1066–1077, 2008, doi: 10.1109/TRO.2008.2004832.
- [6] G. P. Stein, O. Mano, and A. Shashua, "Robust method for computing vehicle ego-motion," in *IEEE Intelligent Vehicles Symposium, Proceedings*, 2000, pp. 362–368, doi: 10.1109/ivs.2000.898370.
- [7] Z. Zhao *et al.*, "A novel method of fabricating an antibacterial aluminum-matrix composite coating doped graphene/silver-nanoparticles," *Materials Letters*, vol. 245, pp. 211–214, Jun. 2019, doi: 10.1016/j.matlet.2019.02.121.
- [8] E. S. Sabry, S. Elagooz, F. E. A. El-Samie, N. A. El-Bahnasawy, and G. M. El-Banby, "SIFT and ORB performance assessment for object identification in different test cases," *Journal of Optics*, vol. 53, no. 3, pp. 1695–1708, Aug. 2024, doi: 10.1007/s12596-023-01170-5.
- [9] T. Brox and J. Malik, "Large displacement optical flow: Descriptor matching in variational motion estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 3, pp. 500–513, 2011, doi: 10.1109/TPAMI.2010.143.
- [10] S. Gauglitz, T. Höllerer, and M. Turk, "Evaluation of interest point detectors and feature descriptors for visual tracking," *International Journal of Computer Vision*, vol. 94, no. 3, pp. 335–360, Sep. 2011, doi: 10.1007/s11263-011-0431-5.
- [11] L. A. Clemente, A. J. Davison, I. D. Reid, J. Neira, and J. D. Tardós, "Mapping large loops with a single hand-held camera," in *Robotics*, The MIT Press, 2018, pp. 297–304, doi: 10.7551/mitpress/7830.003.0038.
- [12] A. Pretto, E. Menegatti, M. Bennewitz, W. Burgard, and E. Pagello, "A VO framework robust to motion blur," in *Proceedings - IEEE International Conference on Robotics and Automation*, 2009, pp. 2250–2257, doi: 10.1109/ROBOT.2009.5152447.
- [13] S. Wang, R. Clark, H. Wen, and N. Trigoni, "DeepVO: Towards end-to-end VO with deep recurrent convolutional neural networks," in *IEEE International Conference on Robotics and Automation*, Jul. 2017, pp. 2043–2050, doi: 10.1109/ICRA.2017.7989236.
- [14] D. DeTone, T. Malisiewicz, and A. Rabinovich, "Deep image homography estimation," *arXiv-Computer Science*, pp. 1–6, Jun. 2016.
- [15] G. Costante, M. Mancini, P. Valigi, and T. A. Ciarfuglia, "Exploring representation learning with CNNs for frame-to-frame ego-motion estimation," *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 18–25, Jan. 2016, doi: 10.1109/LRA.2015.2505717.
- [16] H. Pu, J. Luo, G. Wang, T. Huang, H. Liu, and J. Luo, "Visual SLAM integration with semantic segmentation and deep learning: a review," *IEEE Sensors Journal*, vol. 23, no. 19, pp. 22119–22138, Oct. 2023, doi: 10.1109/JSEN.2023.3306371.
- [17] A. Ma, P. Li, C. Zhang, Z. Wang, and Z. Wang, "MN-SLAM: Multi-networks visual SLAM for dynamic and complicated environments," in *2022 11th International Conference on Information Communication and Applications, ICICA 2022*, 2022, pp. 73–77, doi: 10.1109/ICICA56942.2022.00021.
- [18] T. Pire, T. Fischer, J. Civera, P. D. Cristoforis, and J. J. Berles, "Stereo parallel tracking and mapping for robot localization," in *IEEE International Conference on Intelligent Robots and Systems*, Dec. 2015, pp. 1373–1378, doi: 10.1109/IROS.2015.7353546.
- [19] G. Klein and D. Murray, "Parallel tracking and mapping for small AR workspaces," *2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality, ISMAR*, pp. 225–234, 2007, doi: 10.1109/ISMAR.2007.4538852.
- [20] H. Strasdat, J. M. M. Montiel, and A. J. Davison, "Real-time monocular SLAM: Why filter?," in *Proceedings - IEEE International Conference on Robotics and Automation*, 2010, pp. 2657–2664, doi: 10.1109/ROBOT.2010.5509636.
- [21] G. P. Huang, A. I. Mourikis, and S. I. Roumeliotis, "A first-estimates jacobian EKF for improving SLAM consistency," *Springer Tracts in Advanced Robotics*, vol. 54, pp. 373–382, 2009, doi: 10.1007/978-3-642-00196-3_43.
- [22] C. Forster, M. Pizzoli, and D. Scaramuzza, "SVO: Fast semi-direct monocular VO," in *IEEE International Conference on Robotics and Automation*, Sep. 2014, pp. 15–22, doi: 10.1109/ICRA.2014.6906584.
- [23] J. Engel, T. Schöps, and D. Cremers, "LSD-SLAM: Large-scale direct monocular SLAM," *European conference on computer vision*, pp. 1–25, 2014, doi: 10.1007/978-3-319-10605-2_54.
- [24] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "ORB-SLAM: A versatile and accurate monocular SLAM system," *IEEE*





- Transactions on Robotics*, vol. 31, no. 5, pp. 1147–1163, Oct. 2015, doi: 10.1109/TRO.2015.2463671.
- [25] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *Proceedings of the IEEE International Conference on Computer Vision*, 2011, pp. 2564–2571, doi: 10.1109/ICCV.2011.6126544.
- [26] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the IEEE International Conference on Computer Vision*, 1999, vol. 2, pp. 1150–1157, doi: 10.1109/iccv.1999.790410.
- [27] G. Grisetti, R. Kummerle, C. Stachniss, and W. Burgard, "A tutorial on graph-based SLAM," *IEEE Intelligent Transportation Systems Magazine*, vol. 2, no. 4, pp. 31–43, Dec. 2010, doi: 10.1109/MITS.2010.939925.
- [28] J. Bhola, M. Shabaz, G. Dhiman, S. Vimal, P. Subbulakshmi, and S. K. Soni, "Performance evaluation of multilayer clustering network using distributed energy efficient clustering with enhanced threshold protocol," *Wireless Personal Communications*, vol. 126, no. 3, pp. 2175–2189, Oct. 2022, doi: 10.1007/s11277-021-08780-x.
- [29] G. Kumar and R. Kumar, "A survey on planar ultra-wideband antennas with band notch characteristics: Principle, design, and applications," *AEU - International Journal of Electronics and Communications*, vol. 109, pp. 76–98, Sep. 2019, doi: 10.1016/j.aee.2019.07.004.
- [30] M. Ghahremani, Y. Liu, and B. Tiddeman, "FFD: Fast feature detector," *IEEE Transactions on Image Processing*, vol. 30, pp. 1153–1168, 2021, doi: 10.1109/TIP.2020.3042057.
- [31] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "BRIEF: Binary robust independent elementary features," in *Computer Vision—ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, Proceedings, Part IV 11*, 2010, pp. 1–14, doi: 10.1007/978-3-642-15561-1_56.
- [32] J. Li, T. Xu, and K. Zhang, "Real-time feature-based video stabilization on FPGA," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 4, pp. 907–919, Apr. 2017, doi: 10.1109/TCSVT.2016.2515238.

BIOGRAPHIES OF AUTHORS







Gurpreet Singh     is working as an Assistant Professor in the Department of Mechatronics Engineering at Chitkara University, Punjab, India, with over 15 years of teaching experience. Currently pursuing a Ph.D. in mobile robotics, he holds an M.Tech. (machine design). Throughout his tenure, he has undertaken various administrative roles, including serving as the Head of the Department of Mechanical Engineering and academic in-charge at Gian Jyoti Group of Institutions. He has authored more than 10 research articles and filed more than 25 patents. His research areas are mobile robotics, rapid prototyping, and machine design. He can be contacted at email: gurpreet.ace@gmail.com.



Deepam Goyal     received his B.E. (Hons.), M.E. with a gold medal, and Ph.D. degrees in mechanical engineering from Panjab University, Chandigarh, India. Currently, he serves as an Assistant Professor at Chitkara University, Punjab, India. During his doctoral program, he was an inspire fellow of the DST, India. He was recognized among the top 2% scientists by Stanford University in 2022 and 2023. He received the Gold MILCA Award from the Confederation of Indian Industries. His research interests include machine fault diagnostics, vibration, optimization, manufacturing technology, sensors, and machine learning. He has authored a book, 27 SCI journal articles, 55+ articles in other refereed international journals & conferences, 8 book chapters, and holds 6 patents and a copyright. He reviews for several prestigious journals of ASME, IEEE, Elsevier, Springer, Sage, Taylor & Fransis, and Emerald. He can be contacted at email: bkdeepamgoyal@outlook.com.



Vijay Kumar     is presently working as Dean Engineering, Symbiosis Skill and Professional University, Pune and has more than 30 years of teaching experience. He is graduated from MBMEC Jodhpur in 1990 and Gold Medalist in M.Tech. (machine design), and Ph.D. from IIT Roorkee. He is awarded Visiting Fellowship by DST, SRF by CSIR. He has guided 5 Ph.D. and more than 25 M.Tech. candidates. He authored four books and more than 50 publications. His areas of interest are mobile robotics, artificial intelligence, tribology, FEM, composite mechanics, and mechatronics. He has filed 17 patents including two related to the drones and unmanned aerial vehicles. He can be contacted at email: vijay.jadon@gmail.com.