Accurate and Augmented Navigation for quadcopter based on Multi-Sensor Fusion

Kiran Kumar Lekkala[1], Vinay Kumar Mittal[2]
kiran.l13@iiits.in[1], vkmittal@iiits.in[2]
Indian Institute of Information Technology Chittoor, SriCity, AP, India

***Abstract*—In this paper, we propose a navigation system consisting of a novel multi-sensor fusion method for calculating precise and accurate aerial coordinates and orientation, of a quadcopter in indoor and GPS-silent environments. A prototype system is developed that is composed of 2 modules: *Simultanious Localization and Mapping (SLAM)* system that uses *Oriented FAST and rotated BRIEF (ORB)* features, also known as *ORB-SLAM* and an *Extended Kalman Filter (EKF)*. A *Proportional Integral and Differential (PID)* controller is employed for achieving accurate aerial maneuvers. The novel fusion method used in the system enables the quadcopter to calculate the geographical pose of the quadcopter and simultaneously calculate the map of the surroundings with minimal error. Most importantly, we develop an accurate scale-estimation system which calculates the scale of the map generated by the monocular SLAM system. The results of the experiments conducted validate the effectiveness of the system in achieving an accurate estimation of aerial movements in an indoor environment with minimal error. The proposed system can be used even in large areas as it uses a *DBoW (Bag-of-Words) based Loop Closure* and an optimal *Bundle Adjustment* which is a non-linear technique for optimizing the pose and the 3D map-points. This navigation system can be helpful for quadcopters used in diverse applications such as monitoring, surveillance, transportation, construction, etc.**

***Index Terms*—multi-sensor fusion, scale-estimation, tracking and mapping, SLAM**

I. Introduction

In recent years, research on autonomous aerial vehicles has made significant progress that has facilitated the increased utilization of these flying robots for a vast range of applications, some of which are otherwise difficult for human beings. Few examples are transportation, mapping, surveillance, monitoring, person-following [1] relief operation and construction activities etc. [2]. In all these tasks, the key challenge lies in performing aggressive flight maneuvers [3], collaborative construction [4], coordination of quadcopters (Swarm robotics) [5], and throwing, flying and catching back these aerial vehicles/robots [6].

Accurate odometry is a key challenge in *navigation* of *Micro Aerial Vehicles (MAV)* in unknown, unstructured and cluttered environments [7] [8]. Few sensors that help the quadcopter in calculating precise pose-estiamtes are: *Motion Capturing (MoCap)* system, GPS, camera, depth sensor and *Time of Flight* camera etc. In the last two decades, a lot of approaches have been proposed for the problem of *Simultaneous Localization and Mapping (SLAM)* as well as a robust state estimation and control methods for autonomous

Fig. 1: Picture of the modified Parrot AR Drne

mobile and aerial robots. Few of the prominent studies involve use of LIDAR scanners [9], monocular cameras [10], stereo cameras and *'Red Green Blue and Depth (RGB-D)'* cameras [11] etc.

Eliminating *horizontal drift* and estimating *map scale* are two of the key challenges in estimating *Visual odometry* [12]. Initially, to estimate the scale of the map, artificial markers or beacons [13] were used across the trajectory as way-points. These enabled the MAV to exploit the metric measurements of the markers and estimate the scale in the map. Scaled maps can be used in many applications such as monitoring, surveillance, tracking etc. Several monocular SLAM methods have been proposed for quadcopters and few methods proposed on sensor fusion of vision and IMU for efficient scale estimation [14], [15]. The process essentially involves using both, the short-term accuracy of the IMU and the long-term precision of the vision sensor to estimate the motion of the robot [16]. Recently techniques like Image-based motion estimation by augmenting Inertial Navigation have also been proposed [17].

In this paper, a method for accurate localization and mapping is proposed that uses a lightweight monocular camera, mounted as a frontal camera on a Parrot AR Drone for pose-estimation, is used for *pose estimation*. Apart from the camera, the sensor data is recorded by using IMU and other devices onboard. Recently proposed approaches have been used, for *visual navigation* of Parrot AR Drone, using PTAM for *scale estimation*, along with Extended Kalman Filter (EKF) for *visual odometry* and eliminating falsely tracked frames [18] [19].

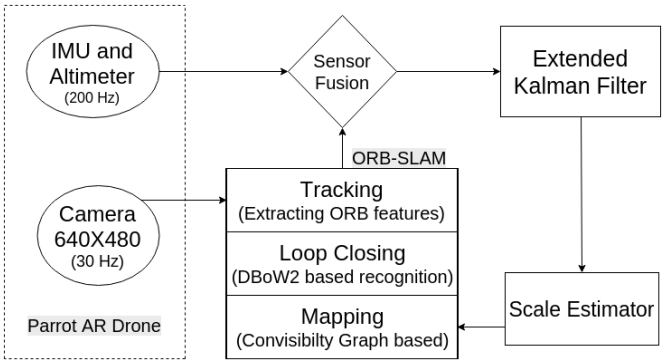

Fig. 2: Block diagram for the proposed-system

A *Robot Operating System (ROS)* package named *tum-ardrone* is also developed recently, which is practically demonstrated using the Parrot AR. Drone 2.0. A major contribution of this paper is a proposed novel multi-sensor based scale-estimation method. For estimating the scale of the environment rendered in the map, we use ORB-SLAM, IMU and a metric sensor (e.g., a pressure sensor to get the linear distance from the ground. ORB-SLAM assumes a monocular camera alone and no additional sensors like IMU, Barometer, Altimeter etc. The performance of the system can be enhanced if the additional sensor measurements are fused with the pose tracked by the ORB-SLAM, which can make the tracking more robust. For the quadcopter to achieve a robust flight, the translational measurements from the ORB-SLAM system are fused with the rotational measurements from the IMU, unlike the previous approach proposed by Engel et al. [20] which involves the quadcopter to rely only on the PTAM localization for state-estimation. This paper has been organized as follows: In Section 2, the prototype of the proposed quadcopter-based system is described in detail. In Section 3 the proposed method is discussed. In Section 4, the experiments carried and the performance evaluation results are discussed. Lastly, in Section 5 a summary is given, along with a scope of further improvements and our further work-in-progress, on this topic.

II. Details of Prototype System

In the proposed system prototype, the Parrot AR Drone 2.0 (Figure. 1) is used as the hardware platform, since it is widely available and is economical. Parrot AR Drone 2.0 is equipped with wide variety of improved sensors in compared to its predecessor Parrot AR Drone 1.0. To compute the orientation, a 9-axis IMU comprising of a gyroscope, accelerometer, and magnetometer is available. The front camera uses a 720 pixels sensor with 93° lens and recording speeds up to 30 fps. The bottom camera is a QVGA sensor with 64° lens and recording up to 60 fps. It essentially involves computing the coarse motion vectors, using optical flow. The planar velocities are then calculated by an ultrasound sensor, co-located with the bottom camera. Moreover, in AR Drone 2.0, Wi-Fi has been upgraded to follow the newer 802.11n standard.

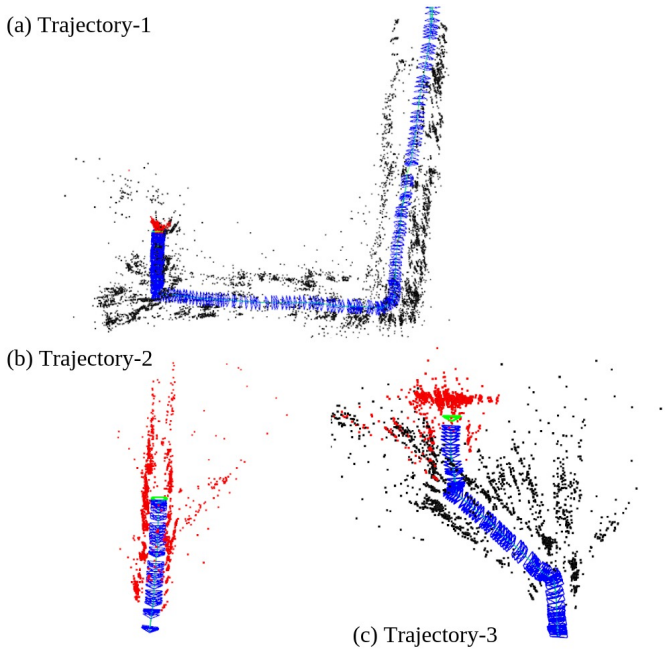

Fig. 3: Instances of trajectory execution when the quadcopter has EKF enabled ORB-SLAM tracking. Tracking of the trajectory is done piece-wise so as to depict the accuracy of the SLAM system.

TABLE I: *Performance of Bundle Adjustment in PTAM*. Note that (a) and (b) are the threads and operations in the system respecitively. (c) is the time taken by each operation.

(a) Thread(PTAM)	*(b) Operation*	*(c) Time (in sec)*
Tracking	Key frame preparation	2.2s
Tracking	Feature Projection	3.8s
Tracking	Patch search	10.2s
Tracking	Iterative pose update	4.8s
Tracking	Total	21s
Mapping	Local Bundle Adjustment	440s
Mapping	Global Bundle Adjustment	6900s
Mapping	Total	7340s

III. Proposed Method

The entire process for the procedure mentioned in this paper is shown in Figure-2:

1) Tracking and Mapping: For visual pose estimation, we employ a Monocular SLAM technique, in our case ORB-SLAM [21]. ORB-SLAM extracts features from the input images and then tracks the successive frames for pose estimation by minimizing the reprojection error using the point distances for different feature vectors. The feature vectors obtained are then updated on a shallow map which is initialized during the mapping process. *Tracking* and *mapping* are run as individual threads so as to simultaneously track and build the depth-map. One instance of the execution of the trajectory is shown in Figure 3.

After initialization, the global map is rotated such that the xy plane corresponds to the accelerometer data. Initially, the

TABLE II: *Performance of Bundle Adjustment in ORB-SLAM*. (a) and (b) are the threads and operations in the system respecitively. (c) is the time taken by each operation.

(a) Thread(ORB-SLAM)	*(b) Operation*	*(c) Time (in sec)*
Tracking	ORB extraction	11.42s
Tracking	Initial Pose Est.	3.45s
Tracking	Track Local Map	16.01s
Tracking	Total	31.60s
Local Mapping	KeyFrame Insertion	11.88s
Local Mapping	Map Point Culling	3.18s
Local Mapping	Map Point Creation	72.96s
Local Mapping	Local BA	360.41s
Local Mapping	KeyFrame Culling	15.79s
Local Mapping	Total	464.27s

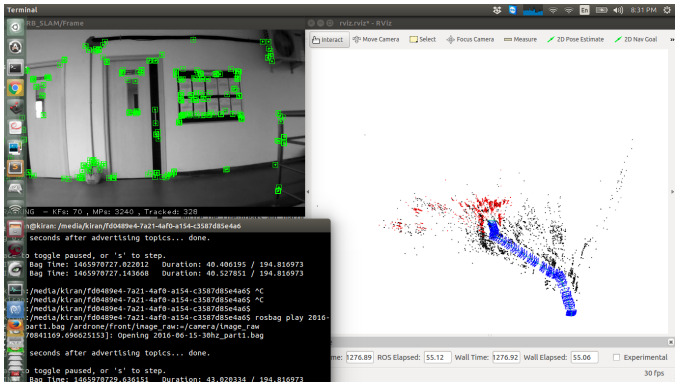

Fig. 4: Program used to evaluate the system.

average key point depth is taken as unity. But during tracking the scale ($\lambda \in R$) is estimated by a closed-form *Maximum Likelihood* approach. The proposed system mentioned for scaling is used, for essentially estimating the λ (map coefficient), while mapping from x to y. This problem is first statistically modeled by the data obtained from the sensor measurements, and then the scale (map coefficient) is estimated using maximum-likelihood (ML) approach.

2) Extended Kalman Filter: This is a non-linear version of a linear Kalman filter which is used to use all available data to estimate a full motion model of the quadcopter. This model can then be used to predict the state of the quadcopter using the IMU data, ORB-SLAM data, and the altimeter measurements.

A. Improved Scale Estimation

To estimate the scale of the quadcopter, we use a monocular SLAM system based on ORB-SLAM and a metric sensor like altimeter present in Parrot AR Drone. Since the altimeter is capable of collecting one-dimensional information namely depth and would get a very poor scale-estimate, the system would be a lot more accurate if we also could obtain the metric information of the planar position as well. For every interval, we obtain the distance calculate the pair of values v and m. In our case m corresponds to the altimeter readings(in meters) and v corresponds to the fused pose estimate from the IMU and the visual ORB-SLAM system. Both the variables are estimates of the motion of the quadcopter and are related as $v = \lambda m$ where λ is the scale of the map estimated by the system. Furthermore, We assume that all the sensor readings are noisy, and so, we model each of them probabilistically using Gaussian noise. Please note that the dimension of the identity matrix is three because we are able to get distance measurements from the altimeters in all the three dimensions.

$$v_i \sim N(\lambda\mu_i, \sigma_v^2 I_{3\times3}) \tag{1}$$

$$m_i \sim N(\lambda\mu_i, \sigma_m^2 I_{3\times3}) \tag{2}$$

Once the sensor data is collected, to find the scale, we minimize the log-likelihood which is shown as follows:

$$L(\mu 1...\mu n, \lambda) \propto \frac{1}{2}\sum_{i=1}^{n}\left(\frac{|v_i - \lambda\mu_i|}{\sigma_v^2} + \frac{|m_i - \mu_i|}{\sigma_m^2}\right) \tag{3}$$

After solving this minimization problem we get to know that the above equation has a global minimum at:

$$\mu_i^\star = \frac{\lambda^\star\sigma_m^2 + \sigma_v^2 m_i}{\lambda^{\star 2}\sigma_m^2 + \sigma_v^2} \tag{4}$$

Finally, using the obtained global mean, we can determine the scale at every instant which is as follows:

$$\lambda^\star = \frac{s_{vv} - s_{mm} + sign(s_{vm})\sqrt{(s_{vv} - s_{mm})^2 + 4s_{vm}^2}}{2\sigma_v^{-1}\sigma_m s_{vm}} \tag{5}$$

where $s_{vv} = \sigma_v^2\sum_{i=1}^{n} v_i^T v_i$, $s_{mm} = \sigma_v^2\sum_{i=1}^{n} m_i^T m_i$ and $s_{vm} = \sigma_m\sigma_v^2\sum_{i=1}^{n} v_i^T m_i$. Please note that n is the number samples used. Please note that, we are assuming $s_{xy} > 0$ which holds for a large data and $\lambda > 0$ To estimate the scale, we require the measurement variances, in our case σ_m (metric/altimeter variance) and σ_v (vision estimate variance) which can directly be calculated by the sensor data during the trajectory.

B. Improved Control Models and Dynamics

In the global state space, the motion model which is used in the EKF is as follows:

$$\mathbf{s_t} = \left(x_t, y_t, z_t, \dot{x}_t, \dot{y}_t, \dot{z}_t, \Phi_t, \Theta_t, \Psi_t, \dot{\Psi}_t\right) \tag{6}$$

In the above equation, the 10 variables which consist the state space are in metric unites. (x_t, y_t, z_t) are the linear positions in *m*. $(\dot{x}_t, \dot{y}_t, \dot{z}_t)$ correspond the velocity in *m/s* also in the world coordinates. The state space also contains $(\Phi_t, \Theta_t, \Psi_t)$ which represent the angle of the quadcopter, essentially, the roll, pitch and yaw angles measured in *deg*. $\dot{\Psi}_t$ is yaw angular velocity in *deg/sec*.

Each sensor has an observation function $h(x_t)$ which describes how the particular observation vector z_t is derived from the sensor readings. Instead of taking all the individual sensor readings into account, we club all of them in an Extended

Kalman Filter to calculate the odometry. The observation function and measurement vector for the onboard sensor system are given below:

$$h_{met}(\mathbf{s_t}) := \left(\dot{x}_t, \dot{y}_t, \dot{z}_t, \mathbf{\Phi_t}, \mathbf{\Theta_t}, \dot{\mathbf{\Psi}}_\mathbf{t}\right) \tag{7}$$

$$z_{met,t} := \begin{pmatrix} \hat{v}_x \\ \hat{v}_y \\ \frac{\hat{h}_t - \hat{h}_{t-1}}{\delta_{t-1}} \\ \hat{\Phi}_x \\ \hat{\Theta}_t \\ \frac{\hat{\Psi}_t - \hat{\Psi}_{t-1}}{\delta_{t-1}} \end{pmatrix} \tag{8}$$

where $\hat{v}_x$ and $\hat{v}_y$ represent the planar velocity, $\hat{h}_t$ represents the height measurement given by the altimeter and δ_{t-1} is the time interval. a strap-down algorithm is used to get the planar velocity of the quadcopter from the accelerometer measurements:

$$v_{IMU} = \int a_{IMU} \tag{9}$$

$$v_{IMU}[k+1] = v_{IMU}[k] + T.a_{IMU}[k] \tag{10}$$

The monocular SLAM system implemented, using ORB-SLAM sends 6DoF Pose to the EKF at 30Hz. The advantage of the proposed system is that the EKF is computed whenever the new set of data is obtained and not each and every time.

$$h_{vis}(\mathbf{s_t}) := \left(x_t, y_t, z_t, \mathbf{\Phi_t}, \mathbf{\Theta_t}, \mathbf{\Psi_t}\right)^T \tag{11}$$

$$z_{vis,t} := (E_{C,t}, E_{DC}) \tag{12}$$

In the above equation, $E_{C,t}$ is the 6DoF pose from the camera frame and E_{DC} is the pose from the world coordinate system.

$$\begin{pmatrix} x_{t+1} \\ y_{t+1} \\ z_{t+1} \\ \dot{x}_{t+1} \\ \dot{y}_{t+1} \\ \dot{z}_{t+1} \\ \Phi_{t+1} \\ \Theta_{t+1} \\ \Psi_{t+1} \\ \dot{\Psi}_{t+1} \end{pmatrix} = \begin{pmatrix} x_t \\ y_t \\ z_t \\ \dot{x}_t \\ \dot{y}_t \\ \dot{z}_t \\ \Phi_t \\ \Theta_t \\ \Psi_t \\ \dot{\Psi}_t \end{pmatrix} + \delta_t \begin{pmatrix} \dot{x}_t \\ \dot{y}_t \\ \dot{z}_t \\ \dot{\dot{x}}(s_t) \\ \dot{\dot{y}}(s_t) \\ \dot{\dot{z}}(s_t, u_t) \\ \dot{\Phi}(s_t, u_t) \\ \dot{\Theta}(s_t, u_t) \\ \dot{\Psi}_t \\ \dot{\dot{\Psi}}(s_t, u_t) \end{pmatrix} \tag{13}$$

In both the observation functions, namely for the metric and the visual sensor, we use $(\mathbf{\Phi_t}, \mathbf{\Theta_t}, \mathbf{\Psi_t})$ which is the roll-pitch-yaw obtained from the IMU. This data is obtained at a frequency of 200 Hz and is stored in the Extended Kalman Filter (EKF).

We used a prediction model based on the Extended Kalman Filter for predicting the state transition from one state to the next. The control command is $\mathbf{u_t} = (\phi_t, \theta_t, \dot{\bar{z}}, \psi_t)$ The designed model is as follows:

TABLE III: Control commands for Parrot AR. Drone

(a) Control Command	(b) Movement	(c) Range
-linear.x	move backward	-1.0 to +1.0
+linear.x	move forward	-1.0 to +1.0
-linear.y	move right	-1.0 to +1.0
+linear.y	move left	-1.0 to +1.0
-linear.z	move down	-1.0 to +1.0
linear.z	move up	-1.0 to +1.0
-angular.z	turn right	-1.0 to +1.0
+angular.z	turn left	-1.0 to +1.0

$$\dot{\mathbf{\Phi}}(\mathbf{s_t}, \mathbf{u_t}) = c_3 \bar{\mathbf{\Phi}}_t - c_4 \mathbf{\Phi_t} \tag{14}$$

$$\dot{\mathbf{\Theta}}(\mathbf{s_t}, \mathbf{u_t}) = c_3 \bar{\mathbf{\Theta}}_t - c_4 \mathbf{\Theta_t} \tag{15}$$

$$\dot{\dot{\mathbf{\Psi}}}(\mathbf{s_t}, \mathbf{u_t}) = c_5 \bar{\dot{\mathbf{\Psi}}}_t - c_6 \dot{\mathbf{\Psi}}_\mathbf{t} \tag{16}$$

$$\dot{\dot{z}}(\mathbf{s_t}, \mathbf{u_t}) = c_7 \bar{\dot{z}}_t - c_8 \dot{z}_t \tag{17}$$

In the above equations, c_1, c_2, c_3, c_4, c_5, c_6, c_7 and c_8 are the proportionality coefficients obtained from the data collected in a series of test flights. Also, note that the control commands for Parrot AR. Drone are mentioned in Table 3 which are further processed by the PID controller [22].

C. System Augmentation

We utilize a novel ORB-SLAM based augmentation method of efficient Bundle adjustment as proposed by [21]. Using this technique, which, essentially contains the following steps, the quadcopter can maneuver over an increased range with a scale-aware accurate map. Map points are created by triangulating ORB features from connected keyframes K_c in the convisibility graph. For map points to be restrained in the map during tracking they should pass a restrictive test so that they are not wrongly tracked. They will be removed if and only if a specific map point is observed from less than 3 keyframes. This way, the resulting global map would contain minimal number of outliers.

Unlike PTAM, where Bundle Adjustment (BA) complexity enormously grows with the number of keyframes, in this method, the redundant keyframes in K_c are deleted or excluded, if 90% of the map points belonging to those keyframes are seen in at least three other keyframes belonging to almost the same scale. This condition makes sure that the keyframes, from which a specific map point was measured most accurately, are retained.

Using ORB-SLAM, we use a novel method to detect loop closures as proposed by [8]. Using this method, the quadcopter can maneuver over an increased range with an accurate map and pose estimates essentially by a fast relocalisation and loop closing [23]. This method is described as follows:

1) Loop Candidates detection: To detect all the possible loop closure candidates, a convisibility graph with threshold $\theta = 90^\circ$ is used to compute the similarity transform with a particular BoG vector of K_i

TABLE IV: Performance of different SLAM systems on the TUM RGB-D dataset. Note that TL denotes Tracking lost and all the values given in the table are in seconds.

(a) Dataset	*(b) PTAM*	*(c) LSD-SLAM*	*(d) ORB-SLAM*
fr1-xyz	1.15s	9.00s	0.90s
fr2-xyz	0.20s	2.15s	0.30s
fr1-floor	TL	38.07s	2.99s
fr1-desk	TL	1.69s	10.65s
fr2-360kidnap	3.81s	TL	2.63s
fr3-longoffice	TL	38.53s	3.45s
fr3-nstr	2.74s	7.54	1.39s
fr3-str	1.04s	TL	1.58s
fr2-desk	TL	31.73s	0.63s
fr3-sit	0.83s	7.73s	0.79s
fr3-half	TL	5.87s	1.34s

2) Computation of the Similarity Transform: Using $Sim(3)$ in Monocular SLAM system which essentially involves 7 DoF state vector i.e. translation, rotation, and scale, We close a loop by computing the Similarity Transform from the current Key-frame K_i and the loop Keyframe K_l.

3) Loop Fusion: Once the similarity transform is computed, the primary step in correcting the loop is to fuse duplicated map-points and insert new edges in the convisibility graph that will attach the loop-closure. The inliers between the corresponding frames are computed and then fused.

4) Essential Graph Optimization: Once the loop-closure is done, the pose graph must be appropriately optimized by distributing the loop-closing error along the entire graph with suitable constraints to correct the scale drift.

IV. Experiments and Results

A series of experiments are carried out, to observe the differences from the predicted results. To test the system, we used the EuRoC MAV Dataset [24] as a benchmark because of its reliability and accurate ground truth measurements.

A. Positioning Accuracy of the Trajectory

We used the V2 and the MH sequences in the EuRoC MAV Dataset [24] for calculating the accuracy of the system in calculating the accuracy of the system with the help of the ground truth provided in the dataset. The results can be seen in Figure 5.

B. Stable Visual Tracking

Unlike the Visual System (PTAM) used in the previous approach by Engel et al, ORB-SLAM is much better in terms of accuracy and robustness. PTAM which was used in the earlier paper does search by patch correlation to track each frame whereas the ORB-SLAM extracts ORB Features which are far more accurate and robust. The time taken by PTAM and LSD-SLAM is compared with the time taken by the ORB-SLAM in Table 4. If we take many state-of-the-art datasets like TUM RGB-D Benchmark, KITTI dataset, College dataset etc. into account PTAM is very much prone to the loss of visual tracking, even to minor occlusions and jerky motions.

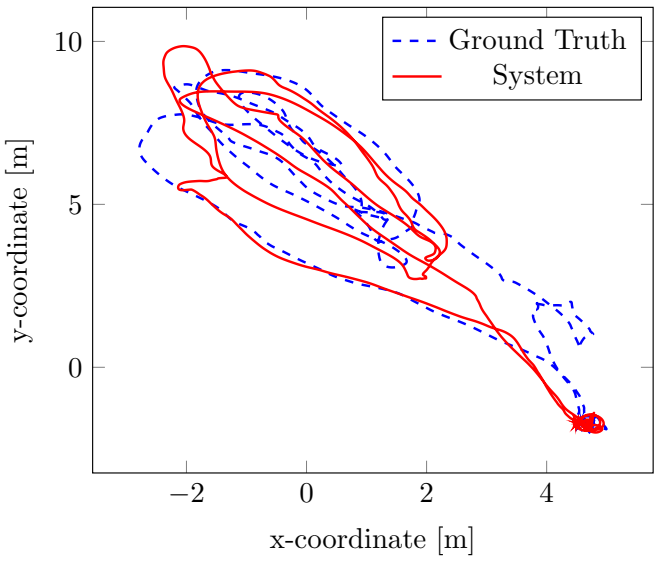

Fig. 5: Analysis of the system performance on the EuRoC dataset. The above figure depicts the ground truth and the trajectory traced out by proposed system.

To further compare our approach, we have also chosen LSD-SLAM which is a semi-dense direct SLAM system, unlike PTAM or ORB-SLAM which are feature based. In terms of accuracy, ORB-SLAM also surpasses LSD-SLAM, as shown in Table 3, which compares the performance of different SLAM systems when run on TUM RGB-D [25] benchmark. Please note that TL stands for Tracking Lost.

C. Operation in increased range

One of the biggest problems with PTAM is that as the number of keyframes increase, the global bundle adjustment tremendously increases over several orders of magnitude. The entire Map building thread slows down and is no longer real-time. This makes PTAM unfit to be used in the bigger SLAM systems as for large trajectories, the navigational system must have a reliable SLAM backend [26]. In our case, ORB-SLAM fits in properly with an optimal bundle adjustment. To verify, we have chosen a long trajectory of about 300m and compared the previous PTAM approach and the ORB- SLAM approach. The time taken for local and global bundle adjustment is given in Table 1 and 2. But in the present approach, owing to optimal approach of Bundle Adjustment, Keyframes and Map points are culled allowing for inclusion and tracking of large-scale trajectories.

V. Summary and Conclusion

In this paper, we presented an improved localization and mapping system which can be implemented on quadcopters operating in indoor surroundings. In the proposed system, we don't completely depend on the visual SLAM system (ORB-SLAM) or the EKF system (IMU) for localization and mapping but utilize the fused data obtained from ORB-SLAM, EKF, and planar range data from an Exteroceptive sensor like an altimeter. To determine the scale of the map using the fused

pose estimate, we use a metric sensor (altimeter) to estimate the scale using a novel Maximum Likelihood approach. Also, by integrating the EKF module with the SLAM system and removing falsely tracked frames by ORB-SLAM, we have increased the efficiency of the SLAM system. Finally, we have also shown how, using the proposed system, the quadcopter can Localize accurately even in large-scale environments. One of the most important contributions of our work is to extend the adaptability of PTAM to environments that are intractable for that system. Coming to the augmentation part, our system detects loop-closures by a novel Bag of Words model and optimizes the pose-graph accordingly.

The proposed system can be extended by including better SLAM systems both by enhancing geometry and by adding Deep learning models [27] or appearance based SLAM techniques [28]. One of the biggest disadvantages of monocular SLAM systems is that they are not accurate and loose tracking if there are any occlusions and strong rotations. There are a lot of other disadvantages of these SLAM systems like tracking without an absolute scale, recognizing false positives etc. Also, for monocular SLAM systems to work there must be an adequate translation with rotation which is not always the case in robotic movements. This results in inefficient translational measurements as the tracking gets lost frequently whenever the drone performs pure rotations. The position estimate, scale estimation and the tracking of the quadcopter could be enhanced by increasing the precision of an exteroceptive sensor. Presently, we use a sonar which gives the depth information linearly.

REFERENCES

[1] Kiran Kumar Lekkala and Vinay Kumar Mittal. Simultaneous aerial vehicle localization and human tracking. Singapore, November 2016.

[2] L Doitsidis, S Weiss, A Renzaglia, E Kosmatopoulos, R Siegwart, D Scaramuzza, and M Achtelik. Optimal surveillance coverage for teams of micro aerial vehicles in gps-denied environments using onboard vision. *Autonomous Robots*, 2012. Online First.

[3] Daniel Mellinger and Vijay Kumar. Minimum snap trajectory generation and control for quadrotors. In *IEEE International Conference on Robotics and Automation, ICRA 2011, Shanghai, China, 9-13 May 2011*, pages 2520–2525, 2011.

[4] Quentin Lindsey, Daniel Mellinger, and Vijay Kumar. Construction of cubic structures with quadrotor teams. In *Proceedings of Robotics: Science and Systems*, Los Angeles, CA, USA, June 2011.

[5] Aleksandr Kushleyev, Vijay Kumar, and Daniel Mellinger. Towards A swarm of agile micro quadrotors. In *Robotics: Science and Systems VIII, University of Sydney, Sydney, NSW, Australia, July 9-13, 2012*, 2012.

[6] Robin Ritz, Mark W. Müller, Markus Hehn, and Raffaello D'Andrea. Cooperative quadrocopter ball throwing and catching. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2012, Vilamoura, Algarve, Portugal, October 7-12, 2012*, pages 4972–4978, 2012.

[7] M Bloesch, S Weiss, D Scaramuzza, and R Siegwart. Vision based mav navigation in unknown and unstructured environments. In *Proc. of The IEEE International Conference on Robotics and Automation (ICRA)*, May 2010.

[8] C. Kerl. Odometry from rgb-d cameras for autonomous quadrocopters. Master's thesis, Technical University Munich, Germany, Nov. 2012.

[9] Slawomir Grzonka, Giorgio Grisetti, and Wolfram Burgard. Towards a navigation system for autonomous indoor flying. In *2009 IEEE International Conference on Robotics and Automation, ICRA 2009, Kobe, Japan, May 12-17, 2009*, pages 2878–2883, 2009.

[10] Markus Achtelik, Simon Lynen, Stephan Weiss, Laurent Kneip, Margarita Chli, and Roland Siegwart. Visual-inertial SLAM for a small helicopter in large outdoor environments. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2012, Vilamoura, Algarve, Portugal, October 7-12, 2012*, pages 2651–2652, 2012.

[11] Erik Bylow, Jürgen Sturm, Christian Kerl, Fredrik Kahl, and Daniel Cremers. Real-time camera tracking and 3d reconstruction using signed distance functions. In *Robotics: Science and Systems IX, Technische Universität Berlin, Berlin, Germany, June 24 - June 28, 2013*, 2013.

[12] Hauke Strasdat, J. M. M. Montiel, and Andrew J. Davison. Scale drift-aware large scale monocular SLAM. In *Robotics: Science and Systems VI, Universidad de Zaragoza, Zaragoza, Spain, June 27-30, 2010*, 2010.

[13] John J. Leonard and Hugh F. Durrant-Whyte. Mobile robot localization by tracking geometric beacons. *IEEE Trans. Robotics and Automation*, 7(3):376–382, 1991.

[14] Philipp Tiefenbacher, Timo Schulze, and Gerhard Rigoll. Off-the-shelf sensor integration for mono-slam on smart devices. In *2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops, Boston, MA, USA, June 7-12, 2015*, pages 15–20, 2015.

[15] G Nuetzi, S Weiss, D Scaramuzza, and R Siegwart. Fusion of imu and vision for absolute scale estimation in monocular slam. *Journal of Intelligent and Robotic Systems*, 61:287299, 2011.

[16] L Doitsidis, A Renzaglia, S Weiss, E Kosmatopoulos, D Scaramuzza, and R Siegwart. 3d surveillance coverage using maps extracted by a monocular slam algorithm. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2011.

[17] Stergios I. Roumeliotis, Andrew Edie Johnson, and James F. Montgomery. Augmenting inertial navigation with image-based motion estimation. In *Proceedings of the 2002 IEEE International Conference on Robotics and Automation, ICRA 2002, May 11-15, 2002, Washington, DC, USA*, pages 4326–4333, 2002.

[18] J. Engel, J. Sturm, and D. Cremers. Accurate figure flying with a quadrocopter using onboard visual and inertial sensing. In *Proc. of the Workshop on Visual Control of Mobile Robots (ViCoMoR) at the IEEE/RJS International Conference on Intelligent Robot Systems (IROS)*, Oct. 2012.

[19] J. Engel, J. Sturm, and D. Cremers. Camera-based navigation of a low-cost quadrocopter. In *Proc. of the International Conference on Intelligent Robot Systems (IROS)*, Oct. 2012.

[20] J. Engel, J. Sturm, and D. Cremers. Scale-aware navigation of a low-cost quadrocopter with a monocular camera. *Robotics and Autonomous Systems (RAS)*, 62(11):1646–1656, 2014.

[21] Montiel J. M. M. Mur-Artal, Raúl and Juan D. Tardós. ORB-SLAM: a versatile and accurate monocular SLAM system. *IEEE Transactions on Robotics*, 31(5):1147–1163, 2015.

[22] Kiran Kumar Lekkala and Vinay Kumar Mittal. Artificial intelligence for precision movement robot. In *Signal Processing and Integrated Networks (SPIN)*, New Delhi, India, February 2015.

[23] Raul Mur-Artal and Juan D. Tardós. Fast relocalisation and loop closing in keyframe-based SLAM. In *2014 IEEE International Conference on Robotics and Automation, ICRA 2014, Hong Kong, China, May 31 - June 7, 2014*, pages 846–853, 2014.

[24] Michael Burri, Janosch Nikolic, Pascal Gohl, Thomas Schneider, Joern Rehder, Sammy Omari, Markus W Achtelik, and Roland Siegwart. The euroc micro aerial vehicle datasets. *The International Journal of Robotics Research*, 2016.

[25] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers. A benchmark for the evaluation of RGB-D SLAM systems. In *Proc. of the International Conference on Intelligent Robot Systems (IROS)*, Oct. 2012.

[26] Gibson Hu, Kasra Khosoussi, and Shoudong Huang. Towards a reliable SLAM back-end. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, November 3-7, 2013*, pages 37–43, 2013.

[27] Dong Ki Kim and Tsuhan Chen. Deep neural network for real-time autonomous indoor navigation. *CoRR*, abs/1511.04668, 2015.

[28] Mark Joseph Cummins and Paul M. Newman. Accelerated appearance-only SLAM. In *2008 IEEE International Conference on Robotics and Automation, ICRA 2008, May 19-23, 2008, Pasadena, California, USA*, pages 1828–1833, 2008.