IMPROVED REFERENCE KEY FRAME ALGORITHM

H. A. Mohamed [1, *], A. Moussa [1, 2], M. M. Elhabiby [3], N. El-Sheimy [1]

[a] Dept. of Geomatics Engineering, University of Calgary, Calgary, AB, Canada - (haytham.abdalla, amelsaye, elsheimy)@ucalgary.ca
[b] Dept. of Electrical and Computer Engineering, Port-Said University, Port-Said, Egypt
[c] Public Works Dept., Ain Shams University, Cairo, Egypt - mmelhabi@ucalgary.ca

Commission II, WG II/3

KEY WORDS: UAV, SLAM, Sensor Fusion, Laser Scan Rangefinder, Optical Flow, Extended Kalman Filter, RKF, ICP.

ABSTRACT:

The autonomous vehicles, such as wheeled robots and drones, efficiently contribute in the search and rescue operations. Specially for indoor environments, these autonomous vehicles rely on simultaneous localization and mapping approach (SLAM) to construct a map for the unknown environment and simultaneously to estimate the vehicle's position inside this map. The result of the scan matching process, which is a key step in many of SLAM approaches, has a fundamental role of the accuracy of the map construction. Typically, local and global scan matching approaches, that utilize laser scan rangefinder, suffer from accumulated errors as both approaches are sensitive to previous history. The reference key frame (RKF) algorithm reduces errors accumulation as it decreases the dependency on the accuracy of the previous history. However, the RKF algorithm still suffers; as most of the SLAM approaches, from scale shrinking problem during scanning corridors that exceed the maximum detection range of the laser scan rangefinder. The shrinking in long corridors comes from the unsuccessful estimation of the longitudinal movement from the implemented RKF algorithm and the unavailability of this information from external source as well. This paper proposes an improvement for the RKF algorithm. This is achieved by integrating the outcomes of the optical flow with the RKF algorithm using extended Kalman filter (EKF) to overcome the shrinking problem. The performance of the proposed algorithm is compared with the RKF, iterative closest point (ICP), and Hector SLAM in corridors that exceed the maximum detection range of the laser scan rangefinder.

1. INTRODUCTION

The autonomous vehicles, such as wheeled robot and drones, efficiently participate in search and rescue operations. These vehicles save humans lives by performing the hazardous missions and tasks. Typically for indoor missions, navigation of these vehicles, in unknown environments, is addressed by the simultaneous localization and mapping approach (SLAM) (Santos et al., 2013). This is accomplished by constructing a map for the unknown environment and simultaneously estimating the vehicle's position inside this map (Thrun and Leonard, 2008) (Bresson et al., 2017) and (Aulinas et al., 2008). SLAM approaches that utilize laser rangefinders depend on a scan matching method of the successive scans (Mohamed et al., 2016). The scan matching process is a key step in the SLAM approaches. This process can adopt a local or global scan matching (Diosi and Kleeman, 2005) and (Bosse and Zlot, 2008). The result of the scan matching process has a fundamental role of the accuracy of the map construction. Both local and global scan matching approaches suffer from accumulated errors because both approaches are affected by the matching errors that occur in previous scans (Hess et al., 2016). Various examples, such as iterative closest point (ICL) (Besl and McKay, 1992), iterative matching range point (IMRP) (Lu and Milios, 1994), iterative dual correspondence (IDC) (Lu and Milios, 1994), polar scan matching (PSM) (Diosi and Kleeman, 2005), iterative closest line (ICL) (Alshawa, 2007), and Hector SLAM (Kohlbrecher et al., 2011), represent local and global scan matching methods and algorithms. All these methods and algorithms suffer from error accumulation over time.

On the other hand, reference key frame (RKF) algorithm proposed 2D real-time scan matching algorithm that depends on a sole sensor, laser scan rangefinder, and does not require external aided sensors (Mohamed et al., 2017). The RKF algorithm attempts to mitigate the accumulated errors that contaminate the local and global scan matching. The RKF algorithm reduces the dependency on the accuracy of the previous result of the scan matching process. This is achieved by computing the transformation matrix parameters with respect to the previous key frame instead of the previous scan as discussed later in Section 2.

However, laser scan matching process in long corridors, that exceed the maximum detection range of the laser scan rangefinder, is a challenging task for vehicle's navigation within unknown environment (Diosi and Kleeman, 2005). The orientation and the lateral movement of the vehicle can be estimated from the scan matching process, while the estimation of the longitudinal movement is challenging as there is no clue for the forward movement. Although the RKF algorithm mitigates the accumulated errors, the algorithm suffers from scale shrinking problem during long corridors. This is due to the shortage of detection of two non-parallel lines as required by the algorithm. As a result, the RKF algorithm depends on the ICP algorithm alone during the non-parallel lines-outage period. Thus, the shrinking in some corridors comes from the unsuccessful estimation of the longitudinal movement from the implemented ICP algorithm and the unavailability of this information from external source as well. This paper proposes an improved version of the RKF algorithm (IRKF) to overcome the shrinking problem. This is achieved by integrating the outcomes of the optical flow with the RKF algorithm using extended Kalman filter (EKF).

This paper is organized as follows: Section 2 summarizes the RKF algorithm. Section 3 illustrates an overview of the integrated navigational system. The used methodology is

---

* Corresponding author

demonstrated in Section 4. The experimental results are presented in Section 5 and finally the conclusion is given in Section 6.

2. REFERENCE KEY FRAME ALGORITHM (RKF) BACKGROUND

(Mohamed et al., 2017) proposed the reference key frame (RKF) algorithm. This algorithm depends on extracting lines from the laser scanner point cloud. Then, two non-parallel lines are elected from the extracted lines to be the mapping reference lines. The scan matching process is performed with the current reference lines and their matching lines in each successive scan. As a result of the vehicle's movement, new two non-parallel lines are elected when the previous reference lines are not detected anymore. Hence, transformation parameters between the previous and current reference lines are computed to preserve the mapping continuity. Thus, the scan matching process is performed with the kernel reference lines during the entire mission. This process is carried out as long as non-parallel lines are detected. In case of lines outage, the ICP algorithm is conducted as a point-to-point scan matching technique. In addition to the computation of the transformation parameters of the entire map, the transformation parameters during the ICP period are computed separately. This is performed to preserve continuity for the new reference lines, once detected, with the entire map. The algorithm switches back to the RKF algorithm once linear features are detected as shown in Figure 1.
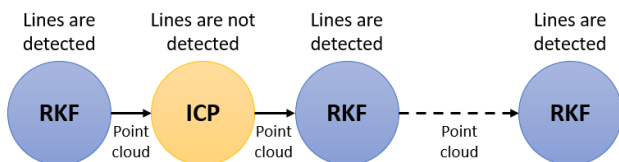

Figure 1. Alternating structure of the RKF algorithm.

3. OVERVIEW OF THE INTEGRATED NAVIGATIONAL SYSTEM

Orientation-compensated translational velocities in metric scale are provided using optical flow sensor called PX4FLOW (Honegger et al., 2013). These velocity values aid the proposed IRKF algorithm to estimate the longitudinal movement of the unmanned aerial vehicle (UAV) especially in the long corridors. Therefore, the laser scan rangefinder and optical flow sensors are fused in an integrated navigational system of the UAV. As a result, the estimated 2D position and heading of the UAV from the laser scan rangefinder are fused with the velocities computed from the optical flow sensor by EKF to provide 2D navigational solution of the UAV and the bias of the optical flow sensor as shown in Figure 2.
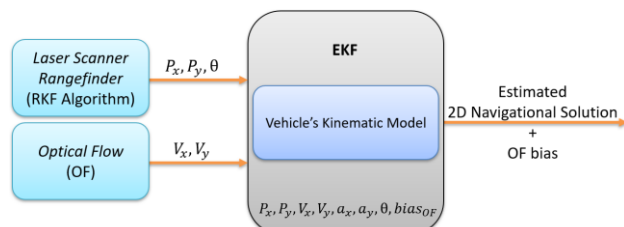

Figure 2. System architecture diagram of the estimation of the 2D navigational solution

3.1 UAV Hardware System Architecture

A quadcopter of X-configuration is utilized during the test experiments. The proposed hardware architecture consists of two main sub-systems, namely navigation and autopilot sub-systems as shown in Figure 3. The navigation sub-system includes low-cost 2D laser scan rangefinder (LS), optical flow sensor (OF) accompanied with sonar, and mini PC, while the autopilot subsystem comprises NAVIO 2 autopilot and laser rangefinder (LRF) attached to raspberry pi 2 B embedded system board.
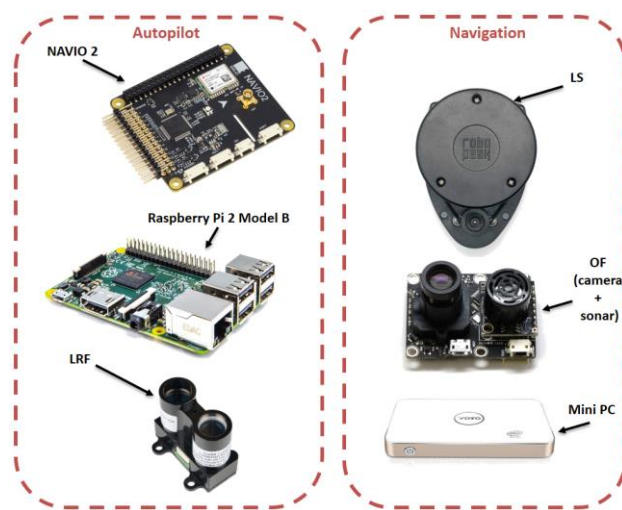

Figure 3. The two main sub-systems.

3.1.1   Navigation sub-system: The utilized low-cost laser scan rangefinder is RPLIDAR 360°. This sensor is developed by RoboPeak and produces 2D point cloud data. The maximum detection range is 6m, and the field of view is 360°. The employed optical flow sensor is PX4FLOW (pixhawk). The PX4FLOW comprises imaging sensor (camera), ultrasonic rangefinder (sonar), and low-cost MEMS 3-axes gyro. An ARM Cortex M4 processor is also mounted on the sensor board in order to process the optical flow approach at a subsampled resolution of 64x64 pixels. The optical flow sensor provides 2D velocities with respect to the body frame (Honegger et al., 2013). The optical flow measurements are computed between the consecutive image frames. The sum of absolute differences (SAD) algorithm is utilized to find the similarity between two blocks of the same size (Niitsuma and Maruyama, 2010). The data acquisition is implemented in the mini PC.

3.1.2   Autopilot sub-system: The NAVIO 2 autopilot board is equipped with dual IMU (MPU9250 9DOF and LSM9DS1 9DOF), magnetometer, barometer (MS5611), U-blox M8N GPS/Glonass/Beidou receiver. This autopilot board requires external embedded system, for instance Raspberry Pi, for code execution and development. The altitude of the UAV is estimated using PulsedLight laser rangefinder. This laser rangefinder sensor is composed of a single laser beam with maximum detection range of 40 m.

4. METHODOLOGY

4.1 Data Fusion by Extended Kalman Filter (EKF)

Extended Kalman Filter (EKF) (Gelb, 1974) have been employed to integrate sensory information from laser scan

rangefinder and optical flow sensor, to provide an improved 2D navigational solution over using individual sensor information.

The EKF describes the kinematic model of the UAV using the following first order state equation:

$$\dot{x} = Fx + Gw \qquad (1)$$

where
- x = state estimate vector
- F = dynamic matrix
- G = noise distribution matrix
- w = process noise vector

The two terms on the right-hand side in Equation (1) respectively define the dynamic model and the stochastic model. The dynamic model describes how states develop over time, while the stochastic model defines the uncertainty in the dynamic model. In this research work, eight states are estimated, as presented in Equation (2); each state is represented with respect to the mapping frame.

$$x = [P_x, P_y, V_x, V_y, a_x, a_y, \theta, bias_{OF}]^T \qquad (2)$$

where
- P_x = position of the UAV in the x direction
- P_y = position of the UAV in the y direction
- V_x = velocity of the UAV in the x direction
- V_y = velocity of the UAV in the y direction
- a_x = acceleration of the UAV in the x direction
- a_y = acceleration of the UAV in the y direction
- θ = heading of the UAV
- $bias_{OF}$ = bias of the optical flow sensor

$$
\begin{bmatrix} \dot{P}_x \\ \dot{P}_y \\ \dot{V}_x \\ \dot{V}_y \\ \dot{a}_x \\ \dot{a}_y \\ \dot{\theta} \\ \dot{bias}_{OF} \end{bmatrix}
=
\begin{bmatrix}
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
\begin{bmatrix} P_x \\ P_y \\ V_x \\ V_y \\ a_x \\ a_y \\ \theta \\ bias_{OF} \end{bmatrix}
$$

$$
+
\begin{bmatrix}
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
w
\qquad (3)
$$

Therefore, the corresponding discrete-time linear system can be expressed as follows:

$$x_k = \Phi_{k,k-1} x_{k-1} + G_{k-1} w_{k-1} \qquad (4)$$

where
- x_k = estimated state vector
- $\Phi_{k,k-1}$ = transition matrix
- G_{k-1} = noise distribution matrix
- w_{k-1} = unit variance white Gaussian noise
- k = current epoch

The transition matrix from time t_0 to t is given as $(\Phi(t, t_0))$ (Brown and Hwang, 2012) and by ignoring the high order terms, the transition matrix can be defined as:

$$
\Phi =
\begin{bmatrix}
1 & 0 & \Delta t & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & \Delta t & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & \Delta t & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & \Delta t & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}_{8x8}
\qquad (5)
$$

The laser scan rangefinder and the optical flow sensors are utilized as external observations (i.e. updates) for the system. The discrete form of the measurement equation that describes the relationship between the observations and the estimated state vector is defined as:

$$Z_k = H_k x_k + \delta n_k \qquad (6)$$

where
- Z = observations vector
- H = design matrix
- X = estimated state vector
- δn = observations noise
- k = current epoch

Figure 4 depicts the work flow diagram of the EKF algorithm. The navigation system has two updates. The first update measurements (P_x, P_y, θ) are provided by the RKF algorithm using the laser scan rangefinder, while the second update measurements (V_x, V_y) are provided by the optical flow sensor. Since the utilized RKF algorithm alternates between two algorithms; namely RKF and ICP algorithms, the covariance matrix of the measurement noise (R_{RKF}, R_{ICP}) changes according to the selected algorithm. Moreover, both algorithms have the same design matrix H_{LRF}. The optical flow update design matrix (H_{OF}) and covariance matrix of the measurement noise (R_{OF}) are given in Equations (32, 34). Otherwise, the navigation system is idle until receiving updates. The inputs of the EKF such as transition matrix, covariance matrix of the system noise, covariance matrix of the measurement noise, design matrices for both updates, and the measurements vectors, should be prepared as described in the next sub-sections.
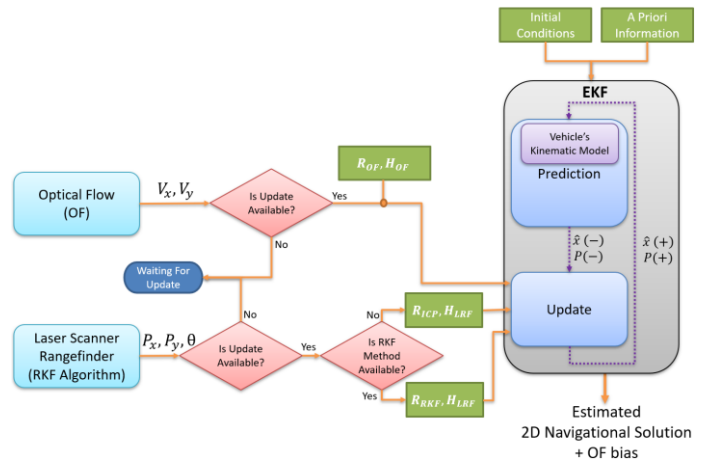

Figure 4. EKF work flow diagram.

4.2 Computation During Laser Scan Rangefinder (LS) Update

The estimated position and heading from the RKF algorithm are used as an external observation of improved accuracy. The position and heading are estimated in the mapping frame, then

the RKF algorithm observations (Z_{LRF}) are used as updates according to:

$$(Z_{LRF})_k = \begin{bmatrix} P_x \\ P_y \\ \theta \end{bmatrix}_{3x1} \tag{7}$$

where P_x = estimated position from the LS in the x direction
P_y = estimated position from the LS in the y direction
θ = estimated heading from the LS

Whereas, the corresponding design matrix during laser scan rangefinder update is given as:

$$H_{LRF} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}_{3x8} \tag{8}$$

where H_{LRF} = design matrix of the LS

4.2.1   Covariance Matrix of RKF algorithm: As mentioned before, the proposed RKF algorithm comprises two methods, namely RKF and ICP algorithms. According to the experiments results in (Mohamed et al., 2017), the RKF algorithm exhibits more robustness than the ICP algorithm. Therefore, each method has its corresponding covariance matrix of the measurement noise (R). Since the RKF algorithm shows lower uncertainty in measurements, the RMSE between the successive scans, after performing the scan matching process, are used to parameterize the covariance matrix of the measurement noise (R_{RKF}).

$$(R_{RKF})_k = \begin{bmatrix} RMSE & 0 & 0 \\ 0 & RMSE & 0 \\ 0 & 0 & RMSE \end{bmatrix}_{3x3} \tag{9}$$

where R_{RKF} = covariance matrix of the measurement noise of the RKF algorithm
$RMSE$ = root mean square error between successive scans

4.2.2   Covariance Matrix of ICP algorithm: The ICP algorithm is employed during the unavailability of two non-parallel lines, for instance in corridors, or lines outage environment. In contrast, the ICP algorithm shows high uncertainty in measurements especially in the longitudinal direction and low uncertainty in measurements elsewhere. This is due to, in contrary to feature-to-feature scan matching, the point-to-point scan matching is prone to outlier association (Aghamohammadi and Taghirad, 2007).

Furthermore, the lack of evidence that supports the longitudinal movement negatively affects the certainty of the measurements. As a result, for the localization, the new positions of the UAV appear to be stationary, which is not the real situation. Thus, the next point clouds are approximately accumulated at the same position, causing shrinking in the corridor, until a new evidence of the forward direction is caught. In order to assist in tackling the shrinking problem of the long corridor, the Eigenvalues and Eigenvectors of one of the two parallel lines (i.e., corridor) are used to compute the covariance matrix of the line after transforming the Eigenvectors from the body frame to the mapping frame. The computed covariance matrix is lent to the covariance matrix of the position measurement noise as shown in Figure 5. In this figure, the yellow rectangle represents the unchanged position of the UAV (i.e., shrinking area), while the blue dashed oval represents the lent covariance matrix. The implemented trajectory of the UAV is represented by the red

dashed line. Figure 5 also demonstrates that the corridor is not necessarily aligned with the mapping frame axes. Thus, the covariance matrix of the position measurement noise is fully populated as given in Equation (24).
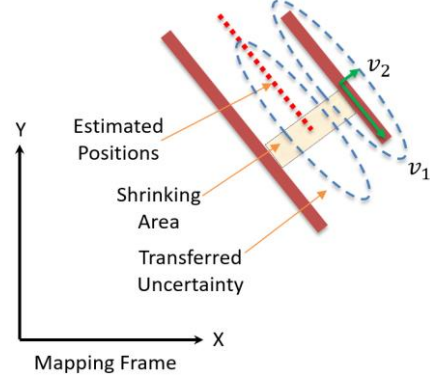

Figure 5. Confidence ellipse in corridors.

The covariance matrix of the position measurement noise is computed using the Eigenvalue and Eigenvector equation as follows:

$$Av_1 = \lambda_1 v_1 \tag{10}$$

$$Av_2 = \lambda_2 v_2 \tag{11}$$

$$A = \begin{bmatrix} a & c \\ c & b \end{bmatrix} \tag{12}$$

$$v_1 = \begin{bmatrix} v_{11} \\ v_{21} \end{bmatrix} \tag{13}$$

$$v_2 = \begin{bmatrix} v_{12} \\ v_{22} \end{bmatrix} \tag{14}$$

$$v_{11}a + v_{21}c = \lambda_1 v_{11} \tag{15}$$

$$v_{11}c + v_{21}b = \lambda_1 v_{21} \tag{16}$$

$$v_{12}a + v_{22}c = \lambda_2 v_{12} \tag{17}$$

$$v_{12}c + v_{22}b = \lambda_2 v_{22} \tag{18}$$

Solving Equations (19), (20), and (21) yields:

$$c = \frac{\lambda_2 v_{12} - \lambda_1 v_{12}}{\left(v_{22} - \frac{v_{21} v_{12}}{v_{11}}\right)} \tag{19}$$

$$a = \frac{\lambda_1 v_{11} - v_{21} c}{v_{11}} \tag{20}$$

$$b = \frac{\lambda_1 v_{21} - v_{11} c}{v_{21}} \tag{21}$$

$$\lambda_1 = RMSE * scale \tag{22}$$

$$\lambda_2 = RMSE \tag{23}$$

where v_1 = the corresponding Eigenvector of the largest Eigenvalue
v_2 = the corresponding Eigenvector of the smallest Eigenvalue
λ_1 = the largest Eigenvalue corresponds to the longitudinal movement (high uncertainty)
λ_2 = the smallest Eigenvalue corresponds to the lateral movement (low uncertainty)
A = the covariance matrix

Therefore, the covariance matrix of the measurement noise is given as:

$$(R_{ICP})_k = \begin{bmatrix} a & c & 0 \\ c & b & 0 \\ 0 & 0 & RMSE * scale \end{bmatrix}_{3x3} \tag{24}$$

where R_{ICP} = covariance matrix of the measurement noise of the ICP algorithm

$RMSE$ = root mean square error between successive scans

From Equation (28), the scale of the heading variance is manually tuned according to the environment due to the high uncertainty of the scan matching result from the ICP algorithm.

4.3 Computation During Optical Flow (OF) Update

On the other hand, the computed velocities from the optical flow sensor are used as second external observation. Since the velocities are computed in the body frame, the optical flow observations are transformed to the mapping frame using the computed transformation matrix from the RKF algorithm. Thereafter, the transformed velocities are used as updates according to:

$$v'_x = v_x + bias_{OF} \qquad (25)$$

$$v'_y = v_y + bias_{OF} \qquad (26)$$

$$(Z_{OF})_k = \begin{bmatrix} v'_x \\ v'_y \end{bmatrix}_{2x1} \qquad (27)$$

where
v'_x = computed velocity by the OF in the x direction
v'_y = computed position by the OF in the y direction
v_x = the OF sensor's true velocity in the x direction
v_y = the OF sensor's true velocity in the y direction
$bias_{OF}$ = bias of the OF sensor
Z_{OF} = observations of the OF

Whereas, the corresponding design matrix during optical flow sensor update is given as:

$$H_{OF} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}_{2x8} \qquad (28)$$

where H_{OF} = design matrix of the optical flow

4.3.1 Covariance Matrix of Optical Flow (OF): In addition to the provided velocities of the optical flow sensor, the quality of the velocities is provided with each observation as well. The quality strength is encapsulated in one byte; the quality value ranges from 0 to 255. The highest quality is 255 and the quality reduces until reach 0. Therefore, the corresponding covariance matrix of the measurement noise is computed as follows:

$$r_{OF} = (256 - quality) * (scale/255) \qquad (29)$$

$$(R_{OF})_k = \begin{bmatrix} r_{OF} & 0 \\ 0 & r_{OF} \end{bmatrix}_{2x2} \qquad (30)$$

where
R_{OF} = covariance matrix of the measurement noise of the OF
$quality$ = the received quality from the sensor
$scale$ = variance adjustment

5. EXPERIMENTAL RESULTS

A comparison has been performed to study the performance of the proposed algorithm amongst various algorithms, such as ICP algorithm, Hector SLAM, corner registration, and RKF algorithm, in corridors that exceed the maximum detection range of the laser scan rangefinder, 6m. The dataset is gathered at the University of Calgary CCIT building. All the experimental figures are to scale representation. The actual length of the corridor is 25 m.

The mapping and position results of the ICP algorithm are illustrated in Figure 6. The blue points represent the constructed map, while the red asterisks represent the vehicle's trajectory. The ICP algorithm suffers from the shrinking problem because of the lack of evidence for longitudinal movement. Furthermore, the impact of the accumulated errors leads up to inclination in the corridor. The ICP algorithm estimates 70% of the actual length.
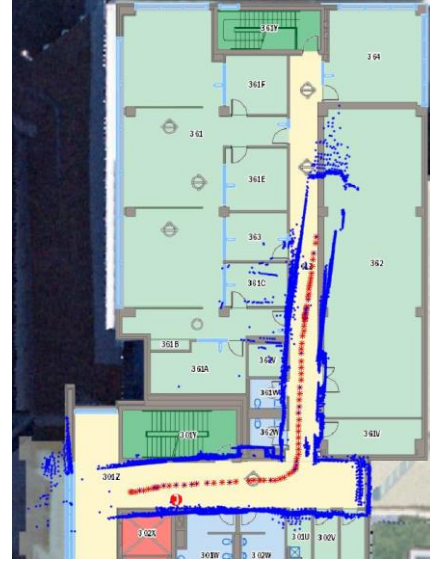

Figure 6. Mapping and position results of the ICP algorithm.

Three level multi-resolution map representations, using grid cell dimensions 20, 10, and 5 cm, have been utilized to evaluate Hector SLAM approach. Figure 7 shows the mapping and position results using Hector SLAM of resolution 5 cm. The blue points represent the constructed map, while the red asterisks represent the vehicle's trajectory. It also suffers from the same problem and for the same reason. The Hector SLAM method estimates 65.2% of the actual length.

Figure 7. Mapping and position results of the Hector SLAM.

Figure 8 depicts the mapping and position results for the RKF algorithm. The blue points represent the constructed map, while the red asterisks represent the vehicle's trajectory. Although the

RKF algorithm exhibits improvement in the corridor scale, it also suffers from the shrinking problem. The improvement comes from the idea of matching of two non-parallel lines (i.e. reference frame), the black arrows show the places that the RKF algorithm gets the second line with the main line of the corridor. The RKF algorithm estimates 75.2% of the actual length.
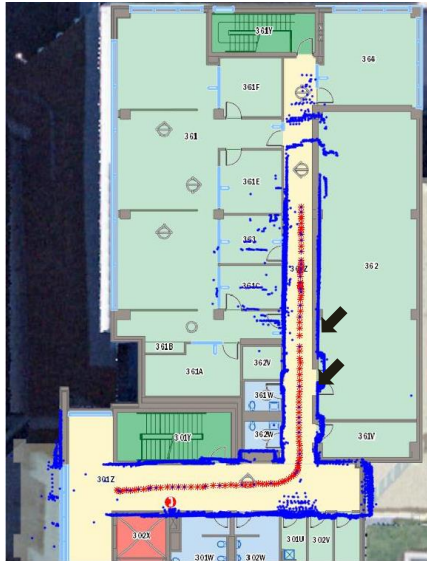

Figure 8. Mapping and position results of the RKF algorithm.

Figure 9 shows the mapping and position results for the proposed IRKF algorithm. The blue points represent the constructed map, while the red asterisks represent the vehicle's trajectory. It is obvious that the shrinking problem does not occur in the vertical corridor since the optical flow sensor aids the proposed IRKF algorithm by providing measurements of the longitudinal movement especially in the corridor. Furthermore, the certainty of the laser scan rangefinder update has also been maintained for the lateral direction only. The proposed algorithm estimates 99.2% of the actual length.
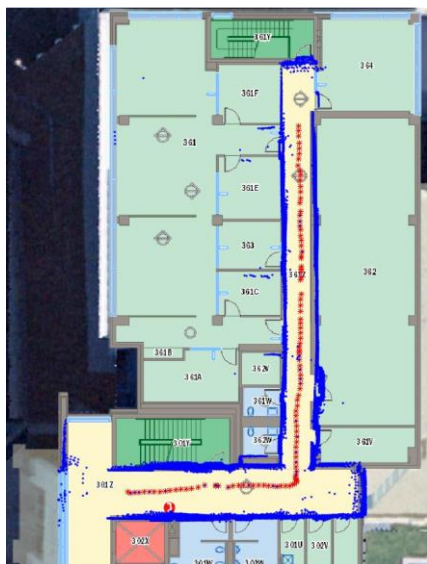

Figure 9. Mapping and position results of the proposed IRKF algorithm.

Table 1 illustrates the estimated length and the corresponding error of the proposed algorithm amongst various methods such as ICP algorithm corner registration, Hector SLAM method, and RKF algorithm.

Method	ICP	Corner Registration	Hector SLAM	RKF	Proposed Algorithm
Estimated Length [m]	17.5	17.6	16.3	18.8	24.8
Error [m]	7.5	7.4	8.7	6.2	0.2

Table 1. Estimated length of the corridor using different methods.

6. CONCLUSION

In long corridors that exceeds the maximum detection range of the laser scan rangefinder, the probability of detecting non-parallel lines is typically low. Thus, the RKF algorithm suffers from scale shrinking problem during scanning such corridors. The shrinking in long corridors comes from the lack of evidence in the forward direction and the unavailability of this information from external source as well. Consequently, the next detected point clouds are approximately accumulated at the same position during the outage of two non-parallel lines. Once a forward clue is detected by the laser scan rangefinder, the estimated position of the UAV starts to move forward again. Thus, this outage period causes a shrink in the corridor structure and therefore affects the entire dimensions of the map. Therefore, extra source such as optical flow sensor is added to the navigational system. The optical flow sensor provides orientation compensated velocities in the x and y directions in the body frame. After transforming the compensated velocities to the mapping frame, EKF algorithm is used to fuse the computed velocities with the estimated 2D position and the heading of the UAV from the RKF algorithm using laser scan rangefinder. The computed velocities aid in estimating the position of UAV especially during navigation in the long corridors. Therefore, the integration between the RKF algorithm and the observations of the optical flow sensor overcomes the shrinking problem. The percentage of the estimation error of the proposed algorithm is 0.8%, while the estimation error of the ICP algorithm, corner registration, Hector SLAM, and RKF are 30%, 29.6%, 34.8%, 24.8% respectively.

REFERENCES

Aghamohammadi, A., Taghirad, H., 2007. Feature-Based Laser Scan Matching For Accurate and High Speed Mobile Robot Localization. EMCR 1–6.

Alshawa, M., 2007. lCL: Iterative closest line A novel point cloud registration algorithm based on linear features. Ekscentar 53–59.

Aulinas, J., Petillot, Y., Salvi, J., Lladó, X., 2008. The SLAM problem: a survey, in: The 11th International Conference of the Catalan Association for Artificial Intelligence, CCIA. Sant Martí d'Empúries, Spain, pp. 363–371.

Besl, P.J., McKay, N.D., 1992. A method for registration of 3-D shapes. IEEE Trans. Pattern Anal. Mach. Intell. 14, 239–256.

Bosse, M., Zlot, R., 2008. Map Matching and Data Association for Large-Scale Two-dimensional Laser Scan-based SLAM. Int'l

J. Robot. Res. 27, 667–691.

Bresson, G., Alsayed, Z., Yu, L., Glaser, S., 2017. Simultaneous Localization and Mapping: A Survey of Current Trends in Autonomous Driving. IEEE Trans. Intell. Veh. 2, 194–220.

Brown, R.G., Hwang, P.Y.C., 2012. Introduction to Random Signals and Applied Kalman Filtering with MATLAB Exercises, 4th Editio. ed. John Wiley & Sons, Inc.

Diosi, A., Kleeman, L., 2005. Laser scan matching in polar coordinates with application to SLAM, in: 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems. pp. 3317–3322.

Gelb, A., 1974. Applied Optimal Estimation. Techniques.

Hess, W., Kohler, D., Rapp, H., Andor, D., 2016. Real-time loop closure in 2D LIDAR SLAM, in: 2016 IEEE International Conference on Robotics and Automation (ICRA). pp. 1271–1278.

Honegger, D., Meier, L., Tanskanen, P., Pollefeys, M., 2013. An open source and open hardware embedded metric optical flow CMOS camera for indoor and outdoor applications, in: Proceedings - IEEE International Conference on Robotics and Automation. IEEE, pp. 1736–1741.

Kohlbrecher, S., Stryk, O. von, Meyer, J., Klingauf, U., 2011. A flexible and scalable SLAM system with full 3D motion estimation, in: 2011 IEEE International Symposium on Safety, Security, and Rescue Robotics. pp. 155–160.

of IEEE Conference on Computer Vision and Pattern Recognition. pp. 935–938.

Mohamed, H., Moussa, A., Elhabiby, M., El-Sheimy, N., Sesay, A., 2017. A Novel Real-Time Reference Key Frame Scan Matching Method. Sensors 17, 1060–1088.

Mohamed, H.A., Moussa, A.M., Elhabiby, M.M., El-Sheimy, N., Sesay, A.B., 2016. Improved Real-Time Scan Matching Using Corner Features. ISPRS - Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci. 41B5, 533–539.

Niitsuma, H., Maruyama, T., 2010. Sum of Absolute Difference Implementations for Image Processing on FPGAs, in: 2010 International Conference on Field Programmable Logic and Applications. IEEE, pp. 167–170.

Santos, J.M., Portugal, D., Rocha, R.P., 2013. An evaluation of 2D SLAM techniques available in Robot Operating System, in: 2013 IEEE International Symposium on Safety, Security, and Rescue Robotics, SSRR 2013. IEEE, pp. 1–6.

Thrun, S., Leonard, J.J., 2008. Simultaneous Localization and Mapping, in: Prof, B.S., Prof, O.K. (Eds.), Springer Handbook of Robotics. Springer Berlin Heidelberg, pp. 871–889.

Lu, F., Milios, E.E., 1994. Robot pose estimation in unknown environments by matching 2D range scans, in: 1994 Proceedings