2014 IEEE International Conference on Robotics & Automation (ICRA)
Hong Kong Convention and Exhibition Center
May 31 - June 7, 2014. Hong Kong, China

Triangulation Toolbox: Open-source Algorithms and Benchmarks for Landmark-based Localization

Sunglok Choi

***Abstract*— This paper introduces about an open-source project, *Triangulation Toolbox*, for landmark-based localization. This project aims to share various algorithms and evaluate their performance in public. At first, landmark-based localization is briefly reviewed with respect to its types of input measurements. From its generalized problem formulation, we adopt a common interface for algorithms in the toolbox. Each algorithm is based on diverse types of input measurements (e.g. distance or bearing angle) in different dimension of space (e.g. 2D or 3D). To support capricious requirements, the toolbox follows the most general convention on measurements and space. In our performance evaluation, nine algorithms were experimentally compared in the view of position and orientation accuracy. The evaluation also leads helpful comments on landmark-based localization and its applications. Since Triangulation Toolbox is designed to be simple and flexible, many developers and researchers can refer implemented algorithms for their applications and also evaluate their own algorithm through the benchmark.**

I. Introduction

Localization is essential and important in various applications such as location-based services (LBS), mobile robots, and intelligent transportation systems. There have been a number of localization techniques due to their importance and diverse operating environments [6] [7]. As shown in Figure 1, source of measurements gives useful categorization to understand characteristic of localization techniques. Many localization techniques have utilized measurements acquired from external environments. These measurements include proximity to objects, observation from landmarks, and local appearance of the environments. For example, detection of RFID tags teaches location through proximity. WiFi access points are a kind of landmarks because they provide pseudo-distance through radio signal strength. Moreover, range data by a laser scanner locally describe the given environment so that their matching to the given map enables to estimate pose on the map. They mostly give absolute position and orientation of a target object because the environments can provide the fixed world coordinate. It is a significant advantage to other techniques by observing only internal motion.

This paper focuses on landmark-based localization, especially a kind of lateration and angulation. Lateration is a process to calculate position of a target object using distances from the object to known landmarks. Similarly, angulation is a process to estimate position and orientation of the object using angular measurements from the object to landmarks. In contrast to temporal-domain approaches based on filtering and tracking, lateration and angulation only rely on spatially and instantaneously observable measurements. *The temporally-independent property* provides simplicity and convenience to many applications (e.g. GPS, cellular positioning, sensor network, and embedded localization sensors). They are still actively investigated in computer vision as a perspective-n-point (PnP) problem [9] [10]. Moreover, they are utilized as a part of the spatial-temporal approaches in the concept of sensor fusion. Even though many researchers have proposed algorithms for landmark-based localization, there has been no open-source implementation to share and evaluate the algorithms. There have been few evaluations only for lateration [11].

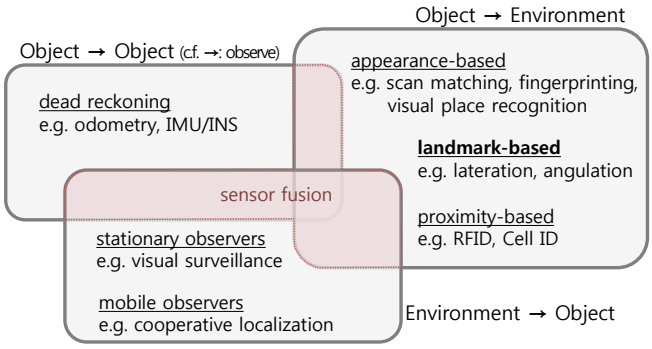
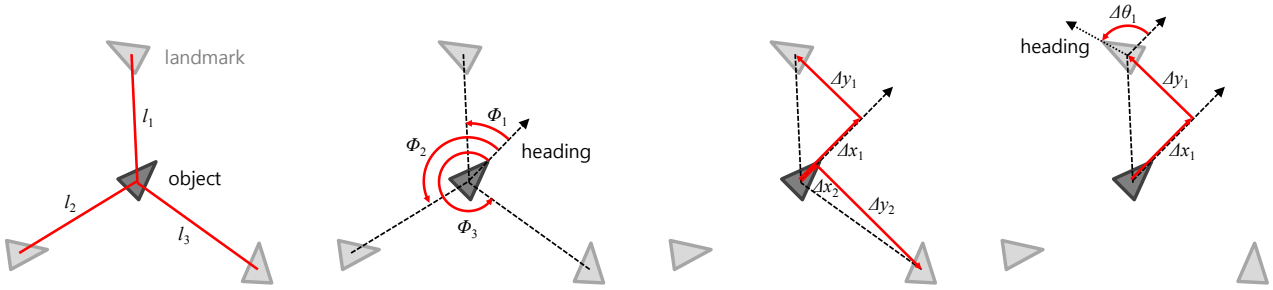


Fig. 1: Localization techniques can be categorized into three groups with respect to their source of measurements: observing internal motion (a.k.a. proprioception), observing environments (a.k.a. exteroception), and being observed by third view-points. Each group shares its major advantages and disadvantages [8].

This paper introduces an open-source project, *Triangulation Toolbox*[1], which aims to share algorithms for landmark-based localization and evaluate their performance. It is implemented in MATLAB script language for readability and portability. However, it would be easily converted to other programming language if its linear algebra library is available (e.g. Eigen [12] for C++). Landmark-based localization is mathematically formulated in Section II. This section also contains a brief introduction on nine included algorithms with respect to their types of input measurements. Triangulation Toolbox and its auxiliaries are based on generalized interface and convention to support various algorithms and

Sunglok Choi is a research scientist with Intelligent Cognitive Technology Department, Electronics and Telecommunications Research Institute (ETRI), Daejeon, South Korea (E-mail: sunglok@hanmail.net).

[1]Homepage: http://sites.google.com/site/sunglok/tt



(a) Distance (b) Bearing angle (c) Displacement (d) Relative pose

Fig. 2: Four types of geometric measurements are described in a two-dimensional space. Especially, (a) and (b) show situations of lateration (a.k.a. range-based localization) and angulation (a.k.a. bearing-based localization), respectively [7].

their requirements, which is explained in Section III. This section also introduces the proposed performance criteria for evaluating algorithms. The nine algorithms were compared in the perspective of position and orientation accuracy while controlling two variables: the magnitude of noise and number of landmarks. Finally, Section V summarizes the paper and leads helpful comments learned from the performance evaluation.

II. Landmark-based Localization

A. Problem Formulation

Landmark-based localization is a process to estimate position, sometimes also orientation, of a target object using measurements from known landmarks. This problem is basically represented by

$$(\hat{\mathbf{p}}_r, \hat{\mathbf{o}}_r) = f(\mathscr{D}; \mathscr{M}), \tag{1}$$

where $\hat{\mathbf{p}}_r$ and $\hat{\mathbf{o}}_r$ are the estimated position and orientation of the target object, $\mathscr{D}$ is a set of measurements from known landmarks, $\mathscr{M}$ is a landmark map which describes pose of each landmark in the given environment, and f is a function to perform localization. More specifically, the function, f, is usually formulated as optimization as follows:

$$f(\mathscr{D}; \mathscr{M}) = \underset{\mathbf{p}_r, \mathbf{o}_r}{\arg\min} \sum_{i=1}^{N} g\big(\mathbf{d}_i - h(\mathbf{p}_r, \mathbf{o}_r; \mathbf{p}_i, \mathbf{o}_i)\big), \tag{2}$$

where $\mathbf{d}_i$ is a measurement from the i-th landmark included in $\mathscr{D}$, $\mathbf{p}_i$ and $\mathbf{o}_i$ is pose of the i-th landmark included in the map $\mathscr{M}$, h is an observation function which simulates a measurement based on pose of the object and landmark, g is a appropriate cost function for types of data, and N is the number of given landmarks.

B. Measurements and Algorithms

There are various types of geometric measurements to be observed from landmarks as shown in Figure 2. Distance from the object to landmarks, l, is one of the simplest measurements. A pair of radio beacon and its receiver is a representative distance sensor. It provides pseudo-distance from its raw measurement such as received signal strength (RSS) and time-of-arrival (TOA). Bearing angle, ϕ, is another measurement to describes geometric relationship between the object and landmarks. A radio receiver with an antenna array determines the direction of signal through angle-of-arrival (AOA). A calibrated camera is also a bearing sensor because bearing angle of each pixel is derived from the pinhole camera model. Displacement is a multi-dimensional measurement to represent distance and bearing angle together. In a two-dimensional space, displacement can be written in (l, ϕ) in the polar coordinate or $(\Delta x, \Delta y)$ in the rectangular coordinate. A laser scanner and depth camera are examples of displacement sensors. Moreover, if a sensor can recognize pose of landmarks, its measurement is expressed in relative pose such as $(\Delta x, \Delta y, \Delta\theta)$ in a two-dimensional space. Since artificial visual landmarks are usually designed to be easily identified with its pose, their detectors are usually relative pose sensors. In addition to four geometric measurements, there are other kinds of measurements such relative distance (e.g. time-difference-of-arrival; TDOA) and relative bearing angle [13].

Table I enumerates various localization algorithms with its measurements. It is clear that algorithms need less number of landmarks when their measurements have more information. In a two-dimensional space, localization using distances or bearing angles needs at least three landmarks. However, at least two landmarks are enough for localization using displacements, and only one landmark is sufficient for localization using relative pose. Since distance is observed regardless of orientation of a target object, lateration cannot estimate orientation of the object. Triangulation Toolbox currently equips nine algorithms as shown in Table I.

C. An Example: 3D Lateration

As an example of landmark-based localization, this subsection explains about 3D lateration simply extended from Sayed et al. [1]. When a target object is at $\mathbf{p}_r = [x_r, y_r, z_r]^\top$ and the i-th landmark is at $\mathbf{p}_i = [x_i, y_i, z_i]^\top$, distance between them, l_i, has a constraint as

$$l_i^2 = (x_i - x_r)^2 + (y_i - y_r)^2 + (z_i - z_r)^2. \tag{3}$$

When we subtract the first constraint from the i-th constraint, it becomes

$$l_i^2 - l_1^2 = \bar{x}_i^2 - 2\bar{x}_i\bar{x}_r + \bar{y}_i^2 - 2\bar{y}_i\bar{y}_r + \bar{z}_i^2 - 2\bar{z}_i\bar{z}_r, \tag{4}$$

where $\bar{\mathbf{p}} = [\bar{x}, \bar{y}, \bar{z}]^\top$ is transformed position, $\bar{\mathbf{p}} = \mathbf{p} - \mathbf{p}_1$, based on the new origin $\mathbf{p}_1$. Therefore, N landmarks compose $N-1$

Name [Ref]	Dim.	Measurement	Output	N	Time Comp.	Function in Triangulation Toolbox
Sayed05-TOA2D [1]	2D	distance	$\mathbf{p}$	≥3	$O(N^3)$	`localize2d_sayed05_toa`
Sayed05-TDOA [1]	2D	relative distance	$\mathbf{p}$	≥3	$O(N^3)$	`localize2d_sayed05_tdoa`
Betke97 [2]	2D	bearing angle	$\mathbf{p}, \mathbf{o}$	≥3	$O(N)$	`localize2d_betke97`
Shim02-Alg [3]	2D	bearing angle	$\mathbf{p}, \mathbf{o}$	≥3	$O(N^3)$	`localize2d_shimshoni02_algebraic`
Shim02-Imp [3]	2D	bearing angle	$\mathbf{p}, \mathbf{o}$	≥3	$O(N^3)$	`localize2d_shimshoni02_improved`
Se05 [4]	2D	displacement	$\mathbf{p}, \mathbf{o}$	≥2	$O(N^3)$	`localize2d_se05`
Sayed05-AOA [1]	2D	displacement	$\mathbf{p}, \mathbf{o}$	≥2	$O(N^3)$	`localize2d_sayed05_aoa`
Sayed05-TOA3D [1]	3D	distance	$\mathbf{p}$	≥4	$O(N^3)$	`localize3d_sayed05_toa`
Thomas05 [5]	3D	distance	$\mathbf{P}$	3*	O(1)	`localize3d_thomas05`

TABLE I: Nine landmark-based algorithms are now included in Triangulation Toolbox. N is the number of landmarks necessary for localization. `Shim02-Alg` is simple triangulation which minimizes algebraic error, and `Shim02-Imp` is its improved version with normalization. `Sayed05-TOA3D` is 3D version of lateration by Sayed et al. [1] which is described in Section II-C. When the number of given landmarks is more than its minimum N, algorithms except `Thomas05` give the optimal solution in least-squares sense. When `Thomas05` has more than three landmarks, it uses only three landmarks for localization and uses the fourth landmark to select unique solution.

linear equations. They are simultaneously represented by

$$\mathrm{H}\bar{\mathbf{p}}_r = \mathbf{b} \tag{5}$$

where the i-th row of H and $\mathbf{b}$ are

$$\mathrm{H}_i = [\bar{x}_i \quad \bar{y}_i \quad \bar{z}_i] \quad \text{and} \quad b_i = \frac{1}{2}\left(\bar{x}_i^2 + \bar{y}_i^2 + \bar{z}_i^2 + l_1^2 - l_i^2\right). \tag{6}$$

The linear equations are solved by $\bar{\mathbf{p}}_r = \mathrm{H}^\dagger \mathbf{b}$, where $\mathrm{H}^\dagger$ is pseudo inverse of H. From the solution, position of the target object $\mathbf{p}_r$ is easily derived as $\bar{\mathbf{p}}_r + \mathbf{p}_1$. This method does not use the first constraints,

$$l_1^2 = \bar{x}_r^2 + \bar{y}_r^2 + \bar{z}_r^2, \tag{7}$$

because this quadratic equation is not directly solved with linear equations (5). That is why this algorithm needs at least four landmarks to lead the unique solution. Even though more than four landmarks are given, the rank of $\mathbf{H}$ can be less than three (underdetermined) if the landmarks are coplanar or collinear.

III. Triangulation Toolbox

A. Convention and Interfaces

Triangulation Toolbox should be based on common convention and generalized interfaces to take account of diverse algorithms and their requirements. For example, some algorithms aim at two-dimensional spaces, but others tackle three-dimensional spaces. Moreover, some algorithms use distance as its input, but others utilize other measurements such as bearing angle, displacement, and relative pose.

Pose and Map Triangulation Toolbox is based on a three-dimensional space because a two-dimensional space is a special case of three-dimensional space. For example, 2D position $[x,y]^\top$ can be regarded as $[x,y,0]^\top$ if the 2D plane is $z=0$. In the toolbox, pose of an object and landmark is represented in a three-dimensional space. In other words, position is written in $\mathbf{p} = [x,y,z]^\top$, and orientation is represented by ZYX Euler angles, $\mathbf{o} = [\theta_x, \theta_y, \theta_z]^\top$. The toolbox follows the right-hand rule for rotation, so counterclockwise rotation along each axis is defined as positive. From ZYX Euler angles, its 3D rotation matrix is derived[2] as follows:

$$\mathrm{R}(\mathbf{o}) = \mathrm{R}_z(\theta_z) \cdot \mathrm{R}_y(\theta_y) \cdot \mathrm{R}_x(\theta_x) \tag{8}$$

A landmark map contains all necessary information about landmarks in the given environment. Since landmark-based localization only depends on pose of landmarks, the map is represented as a series of landmark pose as follows:

$$\mathscr{M} = \begin{bmatrix} \mathbf{p}_1 & \cdots & \mathbf{p}_M \\ \mathbf{o}_1 & \cdots & \mathbf{o}_M \end{bmatrix}, \tag{9}$$

where M is the number of all landmarks in the environment.

Measurements Triangulation Toolbox uses a flexible matrix for representing measurements because its dimension is different according to its types. Distance in a three-dimensional space is same with a two-dimensional case, which is a scalar value, l. However, other measurements have different size in a three-dimensional space. Bearing angle from the object toward a landmark is described by azimuthal and polar angles, $[\phi, \varphi]^\top$, as shown in Figure 3. On the 2D plane, bearing angle is represented as only azimuthal angle ϕ. Displacement from the object to a landmark is $[\Delta x, \Delta y, \Delta z]^\top$. Similarly with a two-dimensional case, displacement can be expressed with distance and bearing angles, $[l, \phi, \varphi]^\top$, in the spherical coordinate. Relative pose is written as $[\Delta x, \Delta y, \Delta z, \Delta\theta_x, \Delta\theta_y, \Delta\theta_z]^\top$. Similarly with a landmark map, a set of measurements is described as a following example:

$$\mathscr{D} = \begin{bmatrix} \mathbf{d}_1 & \cdots & \mathbf{d}_N \end{bmatrix} = \begin{bmatrix} \phi_1 & \cdots & \phi_N \\ \varphi_1 & \cdots & \varphi_N \end{bmatrix}, \tag{10}$$

where N is the number of the observed landmark.

Algorithms Triangulation Toolbox adopts the general formulation of landmark-based localization as common interface of algorithms. It is mathematically represented in

[2]This transformation is implemented in `tran_rad2rot`, and its inverse is implemented in `tran_rot2rad`.

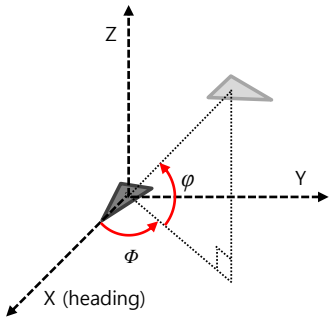


Fig. 3: In a three-dimensional space, relative bearing is represented by a pair of azimuthal angle ϕ and polar angle φ in the object-oriented coordinate. To resolve ambiguity, range of azimuthal and polar angles are defined as $[-\pi,+\pi)$ and $[0,\pi]$, respectively.

Equation (1) whose pose and measurements were defined in a three-dimensional space. As shown in Table I, the interface in MATLAB is

$$[\texttt{pose},\texttt{valid}] = \texttt{localizeXd\_nameYY}(\texttt{data},\texttt{map}), \quad (11)$$

where `data` is measurements, $\mathbf{d}_1 \cdots \mathbf{d}_N$, and `map` is their corresponding pose of landmarks. Each algorithms returns not only estimated pose, but also validity of its elements. For example, 2D angulation by Shimshoni [3] returns its estimated 6 DOF pose with validity, $[1,1,0,0,0,1]^\top$, where 1 means valid and 0 means invalid. 3D lateration explained in Section II-C returns estimated 3D position with its validity, $[1,1,1,0,0,0]^\top$. Through the validity, algorithms is possible to share the common interface in spite of their working dimension and output are different.

B. Performance Criteria

Algorithms can be evaluated in the view of position and orientation accuracy. When the true position $\mathbf{p}_r^*$ is available, position accuracy is quantified by Euclidean distance[3] between two position vectors as

$$e_p(\hat{\mathbf{p}}_r;\mathbf{p}_r^*) = \|\mathbf{p}_r^* - \hat{\mathbf{p}}_r\|. \quad (12)$$

Similarly, when the true orientation $\mathbf{o}_r^*$ is known, orientation accuracy is measured by angular difference[4] between two orientation. Since the first column of 3D rotation matrix means the heading direction of an object, angular difference is calculated by

$$e_o(\hat{\mathbf{o}}_r;\mathbf{o}_r^*) = \arccos\left(\mathrm{R}(\mathbf{o}_r^*)\begin{bmatrix}1\\0\\0\end{bmatrix}\cdot\mathrm{R}(\hat{\mathbf{o}}_r)\begin{bmatrix}1\\0\\0\end{bmatrix}\right). \quad (13)$$

Since these criteria stand for error from the truth, their bigger value means less accurate.

It is also possible to calculate accuracy of algorithms without the ground truth. As shown in Equation (2), most algorithms formulate the problem as optimization, so the degree of optimization is a reasonable measure for accuracy without the ground truth as follows:

$$e_d(\hat{\mathbf{p}}_r,\hat{\mathbf{o}}_r;\mathbf{d},\mathbf{m}) = \sum_{i=1}^{N}\left\|\mathbf{d}_i - h(\hat{\mathbf{p}}_r,\hat{\mathbf{o}}_r;\mathbf{p}_i,\mathbf{o}_i)\right\|^2, \quad (14)$$

which only relies on the given measurements and landmarks. Since its cost function is squared L2-norm of reprojection error, it is not applicable when measurements $\mathbf{d}_i$ has distance and angle together (e.g. displacement and relative pose).

C. Observation Functions

Each type of measurements can be simulated when poses of object and landmark are known. In Triangulation Toolbox, observation functions, h, generate such simulated measurements[5]. They can be utilized for performing simulations and calculating the degree of optimization as shown in Equation (14). Distance is calculated by Euclidean distance between the object and i-th landmark as follows:

$$h_l(\mathbf{p}_r,\mathbf{o}_r;\mathbf{p}_i,\mathbf{o}_i) = \|\mathbf{p}_i - \mathbf{p}_r\|. \quad (15)$$

Displacement is a position vector from the object to landmark in the view point of the object, so it is generated by

$$h_\Delta(\mathbf{p}_r,\mathbf{o}_r;\mathbf{p}_i,\mathbf{o}_i) = \mathrm{R}^\top(\mathbf{o}_r)(\mathbf{p}_i - \mathbf{p}_r). \quad (16)$$

From the simulated displacement, $\mathbf{q} = [q_1,q_2,q_3]^\top = h_\Delta(\mathbf{p}_r,\mathbf{o}_r;\mathbf{p}_i,\mathbf{o}_i)$, bearing angles are derived as

$$h_\phi(\mathbf{p}_r,\mathbf{o}_r;\mathbf{p}_i,\mathbf{o}_i) = \left[\arctan\left(\frac{q_2}{q_1}\right) \quad \arcsin\left(\frac{q_3}{\|\mathbf{q}\|}\right)\right]^\top, \quad (17)$$

where the arc tangent should be implemented by `atan2` to satisfy range of azimuthal angle, $[-\pi,+\pi)$. Relative pose contains position and orientation vectors in the perspective of the object. In contrast to other measurements, it depends on orientation of landmark as follows:

$$h_{\mathbf{p}}(\mathbf{p}_r,\mathbf{o}_r;\mathbf{p}_i,\mathbf{o}_i) = \begin{bmatrix}\mathbf{q}\\ \texttt{tran\_rot2rad}\left(\mathrm{R}^\top(\mathbf{o}_r)\mathrm{R}(\mathbf{o}_i)\right)\end{bmatrix}. \quad (18)$$

While all observation functions are defined in a three-dimensional space, they are easy to use in a two-dimensional space. If we make the target object and all landmarks on the same plane such as $z=0$, observation functions will generate a measurement in a three-dimensional space, but we can selectively use meaningful elements on the measurement. For example, bearing angles in a three-dimensional space have two elements: azimuthal and polar angles. Only azimuthal angle represent a bearing measurement in two-dimensional space. Polar angle is always zero due to such coplanar configuration.

[3]It is implemented in `error_position` in the toolbox.

[4]it is implemented in `error_orientation` in the toolbox.

[5]Four different observation functions are implemented in the toolbox: `observe_distance` for distance, `observe_bearing` for bearing angles, `observe_displacement` for displacement, and `observe_pose` for relative pose.

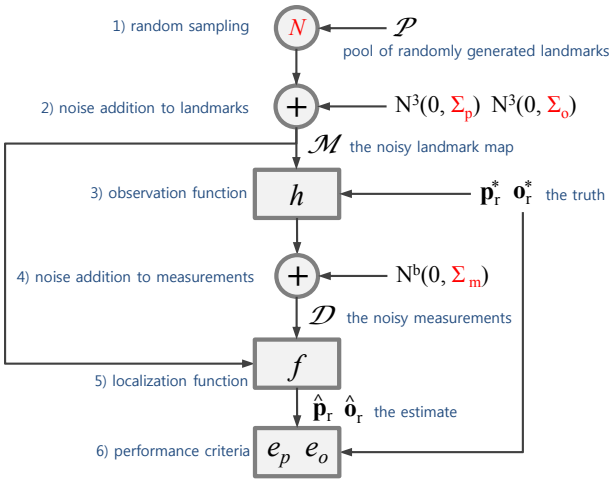


Fig. 4: In the benchmark with simulated data, each trial of the given configuration $(N, \Sigma_{\mathbf{p}}, \Sigma_{\mathbf{o}}, \Sigma_{\mathbf{m}})$ is composed of six steps: 1) sampling landmarks, 2) adding noise to the landmark map, 3) generating simulated measurements, 4) adding noise to the measurements, 5) running the given algorithms, and 6) calculating performance criteria.

D. Benchmark Frameworks

Triangulation Toolbox contains benchmark frameworks based on simulation and real datasets. The simulation is easy to control landmarks and their magnitude of noise. In the simulation, the given space was $100 \times 100 \times 50$ wide and the target object was placed at $\mathbf{p}_r^* = [50, 50, 0]^\top$ with orientation, $\mathbf{o}_r^* = [0, 0, \pi/4]^\top$ whose units are meter and radian. We randomly distributed landmarks all over the given space. Their measurements are generated through observation functions described in Section III-C. Sets of randomly generated landmarks can attain generality because the randomness prohibits biased results of algorithms come from fixed landmarks. There are two kinds of noise: measurement noise and map uncertainty. We used unbiased Gaussian noise[6] to deteriorate measurements or maps. Its magnitude of noise is defined as the standard deviation of Gaussian distribution. First, noise can be directly applied to measurements as follows:

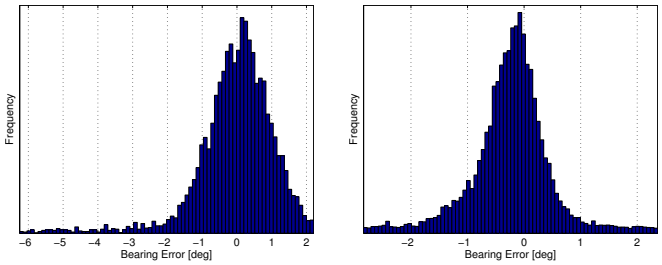
$$\mathbf{d}_i = h(\mathbf{p}_r, \mathbf{o}_r; \mathbf{p}_i, \mathbf{o}_i) + \mathrm{N}^b(0, \Sigma_{\mathbf{m}}), \tag{19}$$

where b is the dimension of measurements (e.g. 1 for distance, 3 for displacement) and $N^b(0, \Sigma_{\mathbf{m}})$ is a b-dimensional Gaussian random vector whose mean is 0 and variance is $\Sigma_{\mathbf{m}}$. Second, uncertainty can be imposed to a landmark map so that its landmarks have deteriorated pose. Measurements from the noisy map become

$$\mathbf{d}_i = h\Big(\mathbf{p}_r, \mathbf{o}_r; \mathbf{p}_i + \mathrm{N}^3(0, \Sigma_{\mathbf{p}}), \mathbf{o}_i + \mathrm{N}^3(0, \Sigma_{\mathbf{o}})\Big). \tag{20}$$

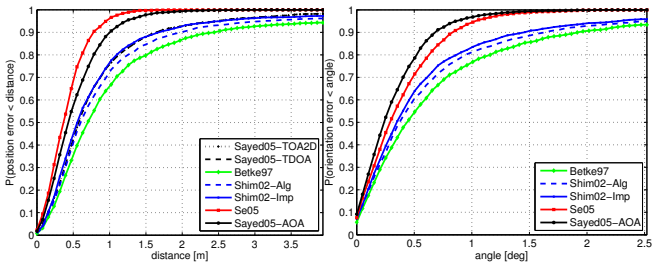
The overall procedure is presented in Figure 4. In contrast to the simulation, real datasets may contain noise comes from actual landmarks and their measurements. In addition,

[6]It is implemented in `apply_noise_gauss` in the toolbox.



(a) Bearing noise at Roh's dataset (b) Bearing noise at MRCLAM

Fig. 5: Bearing measurements in Roh's angulation dataset and UTIAS MRCLAM dataset had almost unbiased noise as shown in (a) and (b), respectively. However, distance measurements in UTIAS MRCLAM dataset were biased at 0.08 meters. They might result from inaccurate range estimation using a single camera.



(a) e_p at $N = 4$ and $\sigma_p = 0.6$ m (b) e_o at $N = 4$ and $\sigma_p = 0.6$ m

Fig. 7: In detail, position and orientation accuracies at $N = 4$ and $\sigma_p = 0.6$ m ($\sigma_o = 0$ and $\sigma_m = 0$) are presented in cumulative distributions of their error.

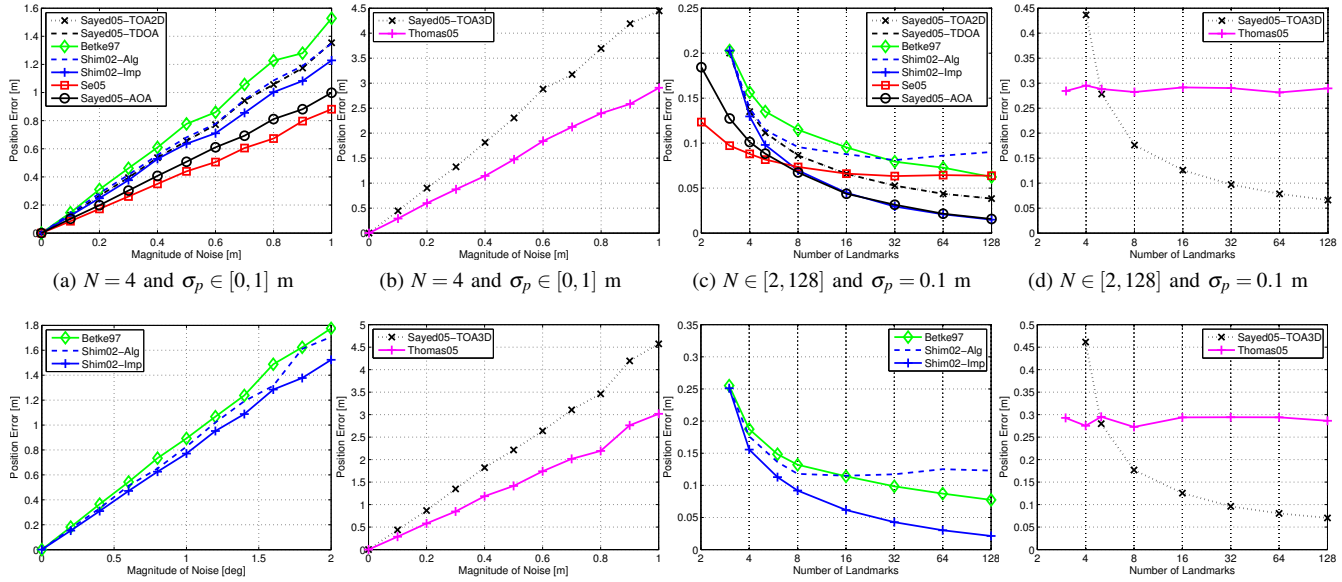
Triangulation Toolbox includes two real datasets: Roh's angulation dataset [14] and UTIAS MRCLAM dataset [15]. Figure 5 presents distributions of their measurement noise.

IV. Performance Evaluation

A. Simulated and Real Datasets

Nine algorithms presented in Table I were evaluated in the simulation mentioned in Section III-D. For statically meaningful results, each algorithm was executed 2×10^3 times in each configuration. Accuracy of each configuration were decided by median of 2×10^3 results. When an algorithm was failed (e.g. invalid or non-unique solutions), its accuracy was assigned as an infinite value not to violate finding median.

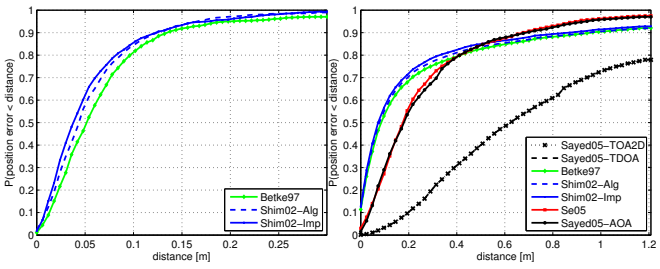
Four different experiments were performed while controlling three variables: the number of landmarks N, landmark noise $(\Sigma_{\mathbf{p}}, \Sigma_{\mathbf{o}})$, and measurement noise $\Sigma_{\mathbf{m}}$. In the first experiment, the magnitude of landmark uncertainty was varied from 0.0 to 1.0 meter with 4 random landmarks. The second experiment was performed with 2 to 128 landmarks at the fixed landmark uncertainty. Similarly, in the third experiment, the magnitude of measurement noise was changed from 0.0 to 1.0 meter and 0.0 to 2.0 degrees for



(a) $N = 4$ and $\sigma_p \in [0,1]$ m (b) $N = 4$ and $\sigma_p \in [0,1]$ m (c) $N \in [2,128]$ and $\sigma_p = 0.1$ m (d) $N \in [2,128]$ and $\sigma_p = 0.1$ m

(e) $N = 4$ and $\sigma_m \in [0,2]$ deg (f) $N = 4$ and $\sigma_m \in [0,1]$ m (g) $N \in [2,128]$ and $\sigma_m = 0.1$ deg (h) $N \in [2,128]$ and $\sigma_m = 0.2$ m

Fig. 6: Four experiments with simulated data are presented. (a) and (b) resulted from the varying magnitudes of landmark noise, $\Sigma_{\mathbf{p}} = \sigma_p^2 \, \mathrm{I}_{3\times 3}$ ($\Sigma_{\mathbf{o}} = 0$ and $\Sigma_{\mathbf{m}} = 0$). (c) and (d) came from the various numbers of landmarks, N, with the fixed landmark noise. (e) and (f) resulted from the changing magnitude of measurement noise, $\Sigma_{\mathbf{m}} = \sigma_m^2 \, \mathrm{I}_{b\times b}$ ($\Sigma_{\mathbf{p}} = 0$ and $\Sigma_{\mathbf{o}} = 0$). (g) and (h) came from the changing numbers of landmarks, N, with the fixed measurement noise. Note that angular values are represented in degree unit for intuitive analysis.



(a) e_p at Roh's dataset (b) e_p at MRCLAM dataset

Fig. 8: Two experiments with real datasets are presented in cumulative distribution of position and orientation error similarly to Figure 7. (a) presents position accuracy at Roh's angulation dataset which only includes bearing measurements. Similarly, (b) shows position accuracy at UTIAS MRCLAM dataset which contains range and bearing measurements. More details can be found in the homepage of the toolbox.

translational and rotational measurements, respectively. The last experiment was executed with 2 to 128 landmarks at the fixed measurement noise. Their results are presented in Figure 6, and their specific point was analyzed in Figure 7 in detail. The first and third experiments had almost similar results, and the second and forth experiments were also quite similar each other.

Seven 2D algorithms were evaluated with Roh's angulation dataset [14] and UTIAS MRCLAM dataset [15]. Their results are shown in Figure 8 whose results were almost similar to results with simulated data.

B. Results and Discussion

Accuracy of 2D Algorithms While the magnitude of noise was increasing, position and orientation errors were also linearly increased. It is presented in Figure 6 (a) and (e). In details, `Se05` had the best accuracy than others, especially in position. It was because `Se05` uses displacement (distance and bearing together), so it works similar to others with twice landmarks. Therefore, to achieve better accuracy with less number of landmarks, it is recommended to use a measurement which has more information. In addition, `Shim05-Imp` was slightly better than its basic version, `Shim05-Alg`. `Sayed05-TOA2D` and `Sayed05-TOA3D` had exactly same accuracy, which were almost similar to `Shim05-Alg` and `Shim05-Imp`. `Betke97` had the worst performance even though it uses bearing angle same with `Shim05-Alg` and `Shim05-Imp`.

As the number of landmarks was increasing, position and orientation errors were exponentially decreased. It is shown in Figure 6 (c) and (g). Experimentally, more than 32 landmarks did not enhance accuracy meaningfully. In details, accuracy improvement of `Shim05-Imp` was the most significant. Since `Shim05-Alg` did not take advantage of more landmarks, we can conclude that additional normalization (scaled map and bounded variance) contributes such improvement. Therefore, to achieve better accuracy with more number of landmarks, it is a good choice to apply normalization to a landmark map and measurements. `Sayed05-TOA2D`,

`Sayed05-TOA3D`, and `Betke97` had similar improvement to `Shim05-Imp`, but `Se05` was similar to `Shim05-Alg`.

Accuracy of 3D Algorithms While the magnitude of noise was increasing, position errors by 3D lateration were also increased. It is presented in in Figure 6 (b) and (f). Compared to Figure 6 (a) and (e), `Sayed05-TOA3D` had almost 3.5 times bigger error than its results on the 2D plane. In optimization aspect, the 3D space has more degree of freedom and its search space is much larger. Therefore, to attain the same accuracy with the 2D case, we need to give more information (e.g. more measurements) to restrict the search space. Experimentally, it was around 16 landmarks, which was almost 4 times more than the 2D case with similar accuracy.

Interestingly, `Thomas05` had more accurate than `Sayed05-TOA3D` at $N \leq 4$, but `Sayed05-TOA3D` became better as the number of landmarks was increasing. It was because `Sayed05-TOA3D` took an advantage of least-square approach in the over-constrained situation, $N > 4$. It is good to use `Thomas05` with 3 or 4 landmarks, but better to utilize `Sayed05-TOA3D` with more number of landmarks.

V. Conclusion

This paper introduces about Triangulation Toolbox whose purpose is to share algorithms and evaluate their performance. Since its readability and portability in MATLAB, we believe that many developers and researchers refer this toolbox for their own algorithms and applications. The evaluation was performed on nine included algorithms while controlling two variables: the magnitude of noise and number of landmarks. Their performance was quantified by position and orientation error, and thier time complexity are summarized in Table I.

Lessons Learned from Evaluation From analysis on the evaluation, we can draw following four remarks. *First, if we can choose types of measurements, it is good to select one that has more information.* For example, displacement is preferred rather than distance or bearing angle. Such selection is especially useful in the case of small number of landmarks. *Second, if there are much more landmarks than the minimum requirement, normalization of map and measurements is very effective to improve accuracy.* Similarly, this is a textbook wisdom [16] in geometric computer vision to calculate transformation (e.g. homography or fundamental matrix) between two images. Since many algorithms did not equip with normalization, additional performance verification is necessary as a further work. *Third, if we formulate our localization on 2D plane rather than 3D space, it is possible to enhance accuracy.* Experimentally, its improvement was around 3 times in our evaluation. Even though the operating environment is three-dimensional, localization can be expressed on the 2D plane if a target object and all landmarks are coplanar. Similarly with this dimensional reduction, a recent research on visual odometry [17] formulated 6 DOF pose estimation into 3 and 2 DOF problem on planes. The research reported that such reduction improves accuracy and reduces computing time significantly. *Fourth, too many landmarks does not prominently improve accuracy.* In our evaluation, more than 32 landmarks were not helpful for enhancing accuracy.

Further Works There are many remaining tasks to fulfill purpose of this project. At first, we need to include more datasets, especially measurements comes from real environments. More algorithms, especially using other types of measurements, are required to be implemented for completeness. Finally, the toolbox is now available in MATLAB, so its C++ and Java implementation will encourage many people to join and use this toolbox.

Acknowledgement

The author would like to thank Federico Thomas for sharing his lateration code and Hyun-Chul Roh, Dr. Keith Leung, Yoni Halpern, Prof. Tim Barfoot, and Prof. Hugh Liu for their real datasets. The author also appreciates Dr. Byungjae Park, Yu-Cheol Lee, Prof. Hyun Myung, and Woo-Han Yun for their helpful comments on this paper.

References

[1] A. H. Sayed, A. Tarighat, and N. Khajehnouri, "Network-based wireless location," *IEEE Signal Processing Magazine*, vol. 22, pp. 24–40, 2005.

[2] M. Betke and L. Gurvits, "Mobile robot localization using landmarks," *IEEE Transactions on Robotics and Automation*, vol. 13, pp. 251–263, 1997.

[3] I. Shimshoni, "On mobile robot localization from landmark bearings," *IEEE Transactions on Robotics and Automation*, vol. 18, pp. 971–976, 2002.

[4] S. Se, D. G. Lowe, and J. J. Little, "Vision-based global localization and mapping for mobile robots," *IEEE Transactions on Robotics*, vol. 21, pp. 364–375, 2005.

[5] F. Thomas and L. Ros, "Revisiting trilateration for robot localization," *IEEE Transactions on Robotics*, vol. 21, pp. 93–101, 2005.

[6] J. Borenstein, H. Everett, and L. Feng, *Where Am I? Sensors and Methods for Mobile Robot Positioning*. The University of Michigan, 1996.

[7] J. Hightower and G. Borriello, "A survey and taxonomy of location systems for ubiquitous computing," University of Washington, Tech. Rep. UW-CSE 01-08-03, 2001.

[8] S. Choi and J.-H. Kim, "Reducing effect of outliers in landmark-based spatial localization using MLESAC," in *Proc. of IFAC World Congress*, 2008.

[9] L. Kneip, P. T. Furgale, and R. Siegwart, "Using multi-camera systems in robotics: Efficient solutions to the NPnP problem," in *ICRA*, 2013.

[10] L. Kneip, D. Scaramuzza, and R. Siegwart, "A novel parametrization of the perspective-three-point problem for a direct computation of absolute camera position and orientation," in *CVPR*, 2011.

[11] K. Whitehouse, C. Karlof, and D. Culler, "A practical evaluation of radio signal strength for ranging-based localization," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 11, 2007.

[12] "Eigen." [Online]. Available: http://eigen.tuxfamily.org/

[13] K. Briechle and U. Hanebeck, "Localization of a mobile robot using relative bearing measurements," *IEEE Transactions on Robotics and Automation*, vol. 20, pp. 36–44, 2004.

[14] H. C. Roh, "Mobile robot localization based on active beacon system using infrared sensors in indoor environment," Master's thesis, KAIST, 2010.

[15] K. Leung, Y. Halpern, T. Barfoot, and H. Liu, "The UTIAS multi-robot cooperative localization and mapping dataset," *The International Journal of Robotics Research (IJRR)*, vol. 30, no. 8, 2011.

[16] R. Hartley, "In defense of the eight-point algorithm," *IEEE Transaction on Pattern Recognition and Machine Intelligence*, vol. 19, pp. 580–593, 1997.

[17] D. Scaramuzza, F. Fraundorfer, and R. Siegwart, "Real-time monocular visual odometry for on-road vehicles with 1-point RANSAC," in *ICRA*, 2009.