UWB/LiDAR Fusion For Cooperative Range-Only SLAM

Yang Song*, Mingyang Guan*, Wee Peng Tay, Choi Look Law and Changyun Wen

Abstract— **We equip an ultra-wideband (UWB) node and a 2D LiDAR sensor a.k.a. 2D laser rangefinder on a mobile robot, and place UWB beacon nodes at unknown locations in an unknown environment. All UWB nodes can do ranging with each other thus forming a cooperative sensor network. We propose to fuse the peer-to-peer ranges measured between UWB nodes and laser scanning information, i.e. range measured between robot and nearby objects/obstacles, for simultaneous localization of the robot, all UWB beacons and LiDAR mapping. The fusion is inspired by two facts: 1) LiDAR may improve UWB-only localization accuracy as it gives a more precise and comprehensive picture of the surrounding environment; 2) on the other hand, UWB ranging measurements may remove the error accumulated in the LiDAR-based SLAM algorithm. Our experiments demonstrate that UWB/LiDAR fusion enables drift-free SLAM in real-time based on ranging measurements only.**

I. INTRODUCTION

Simultaneous localization and mapping, also known as SLAM, has attracted immense attention in the mobile robotics literature, and many approaches use laser range finders (LiDARs) due to their ability to accurately measure range to the nearby objects. There are two basic approaches to mapping with LiDARs: feature extraction and scan matching. The scan matching approach matches point clouds directly and relate two robot's poses via map constraints. Scan matching is much more adaptable than feature extraction approach as it is much less environmentally dependent. Of all existing scan matching algorithms, GMapping [1] and HectorSLAM [2] are arguably the most well-known and widely used algorithms. GMapping needs odometry input while HectorSLAM is an odometry-less approach. Apart from their respective advantages, one common drawback is that their performance is very vulnerable to the accumulated errors. These errors may come from long run operation of odometry such as GMapping which extracts odometry information directly from odometry sensor. The accumulated errors may also come from the SLAM algorithm itself such as HectorSLAM where the error of robot's pose at current time step will be passed via scan matching procedure to the grid-map which will in turn impair the estimation of robot's pose at next time step.

To eliminate these accumulated errors, loop-closure detection approach [3] is introduced to detect whether the robot has returned to a previously visited location. This approach relies heavily on accuracy of the robot's pose estimate. One another way to remove accumulated errors and enhance the robustness in LiDAR-based SLAM is by sensor fusion [4]. In this paper, we propose to fuse LiDAR sensor with UWB sensors. The objective of fusing UWB and LiDAR is to 1) provide not just landmarks/beacons but also a detailed mapping information about surrounding environment; 2) improve the accuracy and robustness of UWB-based localization and mapping.

However, the fusion is hindered by the discrepancy between the accuracy of the UWB mapping and that of LiDAR mapping. UWB has lower range resolution than LiDAR (the laser ranging error is about 1cm which is about one-tenth of UWB ranging error) so that UWB cannot represent an environment in the same quality as LiDAR can do. In this case, fusion by building LiDAR map directly on top of UWB localization results is not a good solution. In our proposed fusion scheme, we consider to refine the robot's pose obtained from UWB ranging measurements by subjecting it to a scan matching procedure. The diagram of our proposed system is shown in Fig. 1 where the system collects all peer-to-peer UWB ranging measurements consisting of robot-to-beacon ranges and beacon-to-beacon ranges at time t, based on which the robot and beacons' 2D positions and 2D velocities are estimated using extended Kalman filter (EKF), then system feeds the state estimate to scan matching procedure in order to update map and find state offset, which is used to correct EKF's state estimate.
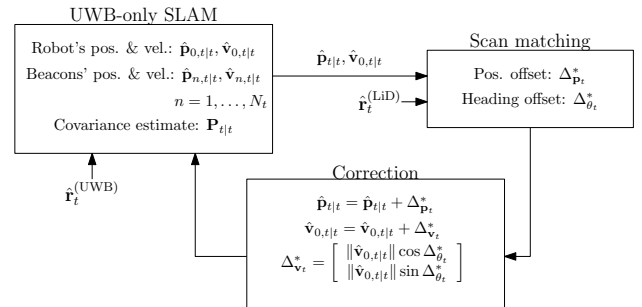

Fig. 1. The digram of proposed system.

Why do we choose UWB? In many practical environments such as enclosed areas, urban canyons, high-accuracy localization becomes a very challenging problem in the presence of multipaths and NLOS propagations. Among various wireless technologies such as the Bluetooth, the Wi-

Fi, the UWB and the ZigBee, UWB is the most promising technology to combat multipaths in cluttered environment. The ultra-wide bandwidth in UWB results in well separated direct path from the multi-paths, thus enabling more accurate ranging using time-of-arrival (TOA) of the direct path.

A. Related Work

1) Range-only SLAM: [5]–[12] propose to simultaneously localize robot(s) and static beacons based on only robot-to-beacon ranging measurements aided by control input e.g. odometry/IMU/etc.. Among these works, [7], [11], [12] use UWB technology to measure ranges. [5], [6], [13], [14] consider robot-to-beacon as well as beacon-to-beacon ranges. Our work further extends the paradigm of [6] by integrating the robot-to-obstacle ranges obtained by laser range finder. The new paradigm allows us to not just localize the robot(s) and map the landmarks (i.e. localize the beacons) but also map the obstacles around the robot. Moreover, as the robot's pose can be estimated based on UWB ranging measurements, the heading information can be derived from its estimated trajectory thus we need no control input.

2) LiDAR based SLAM: Well-known approaches such as HectorSLAM [2], GMapping [1] have pushed forward the research on LiDAR-based SLAM. Gmapping proposes an adaptive approach to learn grid maps using RaoBlackwellized particle filter (RBPF) while the number of required particles in RBPF can be dramatically reduced. HectorSLAM proposes a fast online learning of occupancy grid maps using a fast approximation of map gradients and a multi-resolution grid. Moreover, GMapping needs odometry input whereas HectorSLAM purely replies on the laser ranger finder.

3) Sensor fusion for SLAM: As LiDAR cannot see through occulusions such as smoke, fog or dust, [15] proposes to fuse LiDAR data with Radar data to handle the SLAM problem in the environments with low visibility. The fusion takes place on scan level and map level to maximize the map quality considering the visibility situation. [16] proposes a probabilistic model to fuse sparse 3D LiDAR data with stereo images to provide accurate dense depth maps and uncertainty estimates in real-time. [4] proposes to fuse UWB and visual inertial odometry to remove the visual drift with the aid of UWB ranging measurements thus improving the robustness of system. Moreover, [17] proposes to fuse UWB, inertial navigation system and 3D laser scanner based on Kalman filter to improve the overall localization system. These systems are anchor-based whereas ours is anchorless. [18] proposes a batch-mode solution to fuse ranging measurements from UWB sensors and 3D point-clouds from RGB-D sensors for mapping and localizing unmanned aerial vehicles (UAVs). This system makes the same assumption as ours where the locations of UWB anchors are priori unknown. However, our system can simultaneously localize the robot, UWB beacons and build map in real-time whereas [18] proposes to collect all the data before start doing mapping and localization.

B. Our Contributions

We propose a 2D range-only SLAM approach that combines low-cost UWB sensors and LiDAR sensor for simultaneously localizing the robot, beacons and constructing a 2D map of an unknown environment in real-time. Here are some highlights of the proposed approach:

1) no prior knowledge about the robot's initial position is required, neither is control input;
2) beacon(s) may be moved and number of beacon(s) may be varied while SLAM is proceeding;
3) the robot can move fast without accumulating errors in constructing map;
4) the map can be built even in feature-less environment.

Notations: We use $\mathbb{R}$ to denote the set of real numbers, the superscript ^{T} to represent matrix transpose, and the superscript $^{-1}$ to denote matrix inverse. We use $\mathbf{I}_n$ to represent an identity matrix of size $n \times n$ and use $\mathbf{1}_n$ to denote a column vector with n elements all being one. We use $\otimes$ to denote kronecker product, use $|\cdot|$ to take absolute value, and use $\|\cdot\|$ to denote Euclidean distance. The normal distribution with mean μ and variance σ^2 is denoted as $\mathcal{N}(0, \sigma^2)$. We use $[\mathbf{A}]_i$ to denote the ith row of matrix $\mathbf{A}$, and use $[\mathbf{A}]_{i,j}$ to denote the element at the intersection of the ith row and the jth column of matrix $\mathbf{A}$.

II. PROBLEM FORMULATION

We consider to fuse two kinds of ranging measurements which are

- the peer-to-peer ranges between UWB nodes consisting of robot-to-beacon ranges and beacon-to-beacon ranges, and
- the range between the robot and obstacles available from laser range finder.

We symbolize UWB-based ranging measurements as $\mathbf{r}^{(\text{UWB})}$ and LiDAR-based ranging measurements as $\mathbf{r}^{(\text{LiD})}$. Let

$$\mathbf{x}_t = [p_{0,x,t}, p_{0,y,t}, v_{0,x,t}, v_{0,y,t}]^T,$$
$$\mathbf{m}_t^{(\text{UWB})} = [p_{1,x,t}, p_{1,y,t}, \cdots, p_{N_t,x,t}, p_{N_t,y,t},$$
$$v_{1,x,t}, v_{1,y,t}, \cdots, v_{N_t,x,t}, v_{N_t,y,t}]^T,$$

where $\mathbf{x}_t \in \mathbb{R}^4$ contains 2D location of the robot $\mathbf{p}_{0,t} = [p_{0,x,t}, p_{0,y,t}]^T$ and 2D velocity $\mathbf{v}_{0,t} = [v_{0,x,t}, v_{0,y,t}]^T$ at time t, $\mathbf{m}_t^{(\text{UWB})} \in \mathbb{R}^{4N_t}$ which represents "UWB map" contains N_t number of UWB beacons' 2D locations $\mathbf{p}_{n,t} = [p_{n,x,t}, p_{n,y,t}]^T, n = 1, \ldots, N_t$ and 2D velocities $\mathbf{v}_{n,t} = [v_{n,x,t}, v_{n,y,t}]^T, n = 1, \ldots, N_t$, and let $\mathbf{m}_t^{(\text{LiD})}$ be the "LiDAR map" containing the locations of the observed obstacles.

Remark 1. *No control input e.g. odometry/IMU/etc., is assumed. The heading information (or equivalently 2D velocity) for robot/UWB beacons is derived purely from their trajectory.*

Given all ranging measurements $\mathbf{r}_{1:t}^{(\text{UWB})}$ and $\mathbf{r}_{1:t}^{(\text{LiD})}$ up to time t , our goal is to simultaneously estimate $\mathbf{x}_t$, $\mathbf{m}_t^{(\text{UWB})}$

and $\mathbf{m}_t^{(\mathrm{LiD})}$. We propose to decompose the problem into three coupled steps:

1) find the relative positions of robot and beacons based on $\mathbf{r}_{1:t}^{(\mathrm{UWB})}$, and derive the robot's heading information from its trajectory,
2) construct/update LiDAR map as well as UWB map using $\mathbf{r}_{1:t}^{(\mathrm{LiD})}$ and robot's pose and heading estimates and beacons' pose estimates,
3) correct robot's pose and heading as well as beacons' poses based on the map's feedback.

In what follows, we will elaborate how we implement these three steps.

III. UWB-ONLY SLAM

A. The dynamical and observational models

Let $\boldsymbol{\chi}_t = \left[\mathbf{p}_t^T, \mathbf{v}_t^T\right]^T \in \mathbb{R}^{4(N_t+1)}$ represent the compete UWB-related state consisting of locations of robot and beacons $\mathbf{p}_t = \left[\mathbf{p}_{0,t}^T, \ldots, \mathbf{p}_{N_t,t}^T\right]^T$ and velocities of robot and beacons $\mathbf{v}_t = \left[\mathbf{v}_{0,t}^T, \ldots, \mathbf{v}_{N_t,t}^T\right]^T$. The state $\boldsymbol{\chi}_t$ is evolved according to the following dynamical model:

$$\boldsymbol{\chi}_t = \mathbf{F}_t \boldsymbol{\chi}_{t-1} + \mathbf{G}_t \mathbf{w}_t,$$

where the transition matrix $\mathbf{F}_t$ equals to $\begin{bmatrix} \mathbf{I}_{2(N_t+1)} & \delta\mathbf{I}_{2(N_t+1)} \\ \mathbf{0} & \mathbf{I}_{2(N_t+1)} \end{bmatrix}$, and $\mathbf{G}_t$ equals to $\begin{bmatrix} \delta\mathbf{I}_{2(N_t+1)} \\ \mathbf{I}_{2(N_t+1)} \end{bmatrix}$, the state noise $\mathbf{w}_t$ is zero-mean and has covariance $\mathbf{Q}_t = \sigma_w^2 \mathbf{G}_t \mathbf{G}_t^T$ [19] and δ is the sampling interval.

Let $\mathbf{r}_t^{(\mathrm{UWB})} \in \mathbb{R}^{N_t(N_t+1)/2}$ be the UWB-based ranges measured at time t that consists of robot-to-beacon ranges $\left[\mathbf{r}_t^{(\mathrm{UWB})}\right]_i, i = 1, \ldots, N_t$, and beacon-to-beacon ranges $\left[\mathbf{r}_t^{(\mathrm{UWB})}\right]_i, i = N_t + 1, \ldots, N_t(N_t + 1)/2$, and they are non-linear functions of $\mathbf{p}_t$:

$$\left[\mathbf{r}_t^{(\mathrm{UWB})}\right]_i = h\left(\mathbf{p}_{j,t}, \mathbf{p}_{k,t}\right) = \|\mathbf{p}_{j,t} - \mathbf{p}_{k,t}\|, 0 \leq j < k \leq N_t,$$

where $i = 1, \ldots, N_t(N_t + 1)/2$ index the pairwise combination of UWB nodes. We assume that all peer-to-peer ranges in $[\mathbf{r}_t]_i, i = 1, \ldots, N_t(N_t + 1)/2$ are corrupted by i.i.d. additive noise $n_t \sim \mathcal{N}(0, \sigma_n^2)$, i.e., $\left[\hat{\mathbf{r}}_t^{(\mathrm{UWB})}\right]_i = \left[\mathbf{r}_t^{(\mathrm{UWB})}\right]_i + n_t$. This assumption has been widely adopted under line-of-sight (LOS) scenario [20].

B. EKF update

The update process proceeds via a standard EKF:

$$\hat{\chi}_{t|t-1} = \mathbf{F}_t \hat{\chi}_{t-1|t-1} + \mathbf{w}_t,$$
$$\mathbf{P}_{t|t-1} = \mathbf{F}_t \mathbf{P}_{t-1|t-1} \mathbf{F}_t^T + \mathbf{Q}_t,$$
$$\mathbf{S}_t = \mathbf{H}_t \mathbf{P}_{t|t-1} \mathbf{H}_t^T + \sigma_n^2 \mathbf{I}_{N_t(N_t+1)/2},$$
$$\mathbf{K}_t = \mathbf{P}_{t|t-1} \mathbf{H}_t^T \mathbf{S}_t^{-1},$$
$$\hat{\chi}_{t|t} = \hat{\chi}_{t|t-1} + \mathbf{K}_t \left(\hat{\mathbf{r}}_t^{(\mathrm{UWB})} - h\left(\hat{\chi}_{t|t-1}\right)\right),$$
$$\mathbf{P}_{t|t} = \left(\mathbf{I}_{4(N_t+1)} - \mathbf{K}_t \mathbf{H}_t\right) \mathbf{P}_{t|t-1},$$

where $\hat{\chi}_{t|t}$ and $\mathbf{P}_{t|t}$ are the updated state estimate and covariance estimate, respectively. The measurement matrix

$\mathbf{H}_t \in \mathbb{R}^{L_t \times 4(N_t+1)}$ is defined to be the $\frac{\partial h}{\partial \boldsymbol{\chi}}\Big|_{\hat{\boldsymbol{\chi}}_{t|t-1}}$. As the measurements depend not on the velocity of the nodes, the partial derivatives of $h(\cdot)$ w.r.t. the velocities of robot and beacons are all zeros.

Remark 2. *The estimated velocity of the robot won't be accurate if robot stops because it is derived from robot's trajectory.*

C. Elimination of location ambiguities

As we are dealing with an infrastructure-less localization problem where no prior information about location of nodes is assumed, we can only derive the relative geometry of all UWB nodes based on peer-to-peer ranging measurements. Such relative geometry, as shown in Fig. 2, however, can be arbitrarily translated and rotated. For constructing "LiDAR
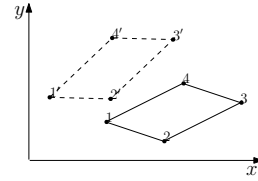

Fig. 2. Illustration of two versions of the same relative geometry of four nodes: one is translated and rotated version of the other one.

map" which will be discussed in next section, we need to eliminate all ambiguous copies of the relative geometry except one. To do this, we translate and rotate the state estimate available $\hat{\chi}_{t|t}$: $\mathbf{R}_t(\theta) \otimes \mathbf{I}_2 \begin{bmatrix} \hat{\mathbf{p}}_{t|t} \\ \hat{\mathbf{v}}_{t|t} \end{bmatrix} + \begin{bmatrix} \mathbf{T}_t \\ \mathbf{0} \end{bmatrix}$, where $\mathbf{R}_t(\theta)$ is a 2D rotation matrix and $\mathbf{T}_t \in \mathbb{R}^{2 \times (N_t+1)}$ is a translation matrix. We find $\mathbf{R}_t(\theta)$ and $\mathbf{T}_t$ such that one beacon node is always located at origin $[0, 0]^T$ and one another beacon node is always located on positive x-axis. In addition, we force the velocity of these two chosen beacon nodes to be zero. These two beacon nodes will set up a local frame under which the LiDAR map can be built and updated.

D. Detection/removal of ranging outliers

NLOS propagation is crucial for high-resolution localization systems [21], [22] because non-negligibly positive biases will be introduced in range measurements, thus degrading the localization accuracy. A rule-of-thumb suggested by [23], [24] has been practised for NLOS detection. This rule says if the power difference, i.e. the received power minus the first-path power, is greater than 10dB the channel is likely to be NLOS. The NLOS ranges are ignored in the EKF update.

Remark 3. *We have tried to detect the outliers using hypothesis testing [25] on the residuals of the EKF, i.e. $\boldsymbol{\epsilon}_t = \hat{\mathbf{r}}_t^{(\mathrm{UWB})} - h\left(\hat{\mathbf{p}}_{t|t-1}\right)$, which is a very common way to remove outliers in the infrastructure-based localization system. In absence of errors, $[\boldsymbol{\epsilon}_t]_i \sim \mathcal{N}\left(0, [\mathbf{S}_t]_{i,i}\right)$. Given confidence intervals for each peer-to-peer ranging measurement, if it's violated, this range is considered an outlier and is ignored. This detection method doesn't work well in relative positioning because in the relative positioning where*

localization of all nodes purely relies on peer-to-peer ranges if some ranges measured at time $t-1$ are outliers, the ranging error will be propagated to all nodes whereby the prediction ranges $\hat{\mathbf{r}}_{t|t-1}^{(\mathrm{UWB})}$ may deviate away from the ranges $\hat{\mathbf{r}}_t^{(\mathrm{UWB})}$ measured at time t even if $\hat{\mathbf{r}}_t^{(\mathrm{UWB})}$ contains no outliers thus resulting in many false alarm detections.

IV. MAP UPDATE

Due to the non-negligible accuracy gap in ranging measurement between UWB and LiDAR, constructing LiDAR map directly on top of UWB localization results is not a promising solution. For example, if the robot keeps scanning the same object for a while, all the scan endpoints should converge to the area where the object is located. However, if the robot's pose and heading are subject to some uncertainties, say 20cm and 3 degrees, respectively, which are quite normal in UWB-only SLAM, mapping the scan endpoints according to robot's pose/heading estimates without any further process would end up with scan endpoints being diverged thereby dramatically reducing the map quality. Fig. 4(a) show such an example. This motivates us to align the scan endpoints by refining the robot's pose/heading estimates. Then, we turn our focus to the scan matching procedure. HectorSLAM [2] proposes a scan matching procedure based on Gauss-Newton method [26] that matches the beam endpoints observed at time t with the latest map $\mathbf{m}_{t-1}^{(\mathrm{LiD})}$ in order to find the optimal transformation/rotation leading to the best match. The scan matching proceeds in a multi-resolution grid starting from grid maps with lower resolution to the grid maps with higher resolution so that it's more likely to find the global solution other than being trapped in local solutions. Our proposed scan matching method as will be discussed below is grounded on [2].

The pose estimates of robot and beacons $\hat{\mathbf{p}}_{t|t}$ and the velocity estimate of the robot $\hat{\mathbf{v}}_{0,t|t}$, which are obtained from EKF-based UWB-only SLAM, would be used for initializing the scan matching procedure. Note that the robot's heading information θ_t could be derived from the 2D velocity as $\hat{\theta}_t = \arctan\left([\hat{\mathbf{v}}_{0,t|t}]_2, [\hat{\mathbf{v}}_{0,t|t}]_1\right)$.

Let $\boldsymbol{\xi}_t = \left[\left(\hat{\mathbf{p}}_{t|t}\right)^T, \hat{\theta}_t\right]^T$. We propose to find the optimal offset $\Delta_{\boldsymbol{\xi}_t} = [\Delta_{\mathbf{p}_t}, \Delta_{\theta_t}]^T$ such that the following objective is minimized:

$$
\begin{aligned}
&\Delta_{\boldsymbol{\xi}_t}^* \\
&= \underset{\Delta_{\boldsymbol{\xi}_t}}{\arg\min} \frac{1}{2} \sum_{i=1}^n \left[1 - \Gamma(f_i\left(\hat{\mathbf{p}}_{0,t|t} + \Delta_{\mathbf{p}_{0,t}}, \hat{\theta}_t + \Delta_{\theta_t}, \hat{\mathbf{r}}_t^{(\mathrm{LiD})}\right))\right]^2 \\
&+ \frac{\gamma}{2} \sum_{0 \le i < j \le N_t} \left[h\left(\hat{\mathbf{p}}_{i,t|t} + \Delta_{\mathbf{p}_{i,t}}, \hat{\mathbf{p}}_{j,t|t} + \Delta_{\mathbf{p}_{j,t}}\right) \right. \\
&\left. \qquad\qquad\qquad\qquad - \left[\hat{\mathbf{r}}_t^{(\mathrm{UWB})}\right]_k\right]^2,
\end{aligned} \tag{1}
$$

where $h(\mathbf{a}, \mathbf{b}) = \|\mathbf{a} - \mathbf{b}\|$ and k indexes the range measured between node i and node j and $\Delta_{\mathbf{p}_t} = \left[\Delta_{\mathbf{p}_{0,t}}^T, \ldots, \Delta_{\mathbf{p}_{N_t,t}}^T\right]^T$.

Remark 4. *As discussed in Section III-C, two UWB beacons are used to establish the UWB coordinate system i.e. one node is always located at $[0,0]^T$ and the other node is fixed on the positive x-axis by letting its coordinate along y-axis be zero. Hence, the values corresponding to these three coordinates in $\Delta_{\boldsymbol{\xi}_t}$ are fixed to zero and we don't update them.*

Now let us take a close look at (1).

LiDAR map matching: The first term in (1) intends to match the laser scan with learnt map by offsetting the robot's pose and heading. The n is the number of laser scan endpoints, and the function $f_i(\cdot)$ maps by transforming/rotating the scan endpoints, say $\mathbf{p}_i^{(\mathrm{sc})}, i = 1, \ldots, n$ which are computed based on $\hat{\mathbf{r}}_t^{(\mathrm{LiD})}$, to their corresponding coordinates $\mathbf{p}_i^{(\mathrm{UWB})} = f_i\left(\mathbf{p}_i^{(\mathrm{sc})}\right), i = 1, \ldots, n$ under UWB coordinate system, and the $\Gamma(\cdot)$ function returns the occupancy probabilities at coordinates $\mathbf{p}_i^{(\mathrm{UWB})} = [x_i, y_i]^T, i = 1, \ldots, n$. The occupancy probability is the probability of a grid cell where $f_i\left(\mathbf{p}_i^{(\mathrm{sc})}\right)$ is in being occupied. Follow the way proposed in [2], the occupancy probability $\Gamma\left(\mathbf{p}_i^{(\mathrm{UWB})}\right) \in [0, 1]$ for the ith scan endpoint is approximated by bilinear interpolation using its four closest integer neighbour points in the grid map, say $\mathbf{p}_{i,j} = [x_i, y_j]^T, i, j = \{0, 1\}$. This bilinear interpolation can be written as

$$
\begin{aligned}
&\Gamma\left(\mathbf{p}_i^{(\mathrm{UWB})}\right) \\
&\approx \frac{y_i - y_0}{y_1 - y_0}\left(\frac{x_i - x_0}{x_1 - x_0}\Gamma(\mathbf{p}_{1,1}) + \frac{x_1 - x_i}{x_1 - x_0}\Gamma(\mathbf{p}_{0,1})\right) \\
&+ \frac{y_1 - y_i}{y_1 - y_0}\left(\frac{x_i - x_0}{x_1 - x_0}\Gamma(\mathbf{p}_{1,0}) + \frac{x_1 - x_i}{x_1 - x_0}\Gamma(\mathbf{p}_{0,0})\right).
\end{aligned}
$$

The gradient of $\nabla\Gamma\left(f_i(\boldsymbol{\xi}_t)\right)$ is $\left[\frac{\partial\Gamma\left(f_i(\boldsymbol{\xi}_t)\right)}{\partial x_i}, \frac{\partial\Gamma\left(f_i(\boldsymbol{\xi}_t)\right)}{\partial y_i}\right]^T$. For the detailed computation of the gradient, we refer the readers to [2].

Remark 5. *The UWB-only SLAM defines a relative coordinate system (Section III-C explains how it's done) on which the LiDAR map is built/updated. That's why the scan endpoints are all mapped to UWB coordinate system.*

UWB map matching: The second term in (1) is a nonlinear least square function depending on all LOS peer-to-peer UWB ranging measurements. Minimizing this function would refine the robot and beacons location estimates where γ is a tradeoff parameter. When $\gamma = 0$, (1) degenerates to (7) in [2] where the matching is done based on LiDAR only.

To find closed-form solution of (1), we approximate functions $\Gamma(\boldsymbol{\xi}_t + \Delta_{\boldsymbol{\xi}_t})$ and $h(\boldsymbol{\xi}_t + \Delta_{\boldsymbol{\xi}_t})$, respectively, by first-order Taylor expansion at point $\boldsymbol{\xi}_t$ as $\Gamma(\boldsymbol{\xi}_t + \Delta_{\boldsymbol{\xi}_t}) \approx \Gamma(\boldsymbol{\xi}_t) + \Delta_{\boldsymbol{\xi}_t}^T\nabla\Gamma(\boldsymbol{\xi}_t)$ and $h(\boldsymbol{\xi}_t + \Delta_{\boldsymbol{\xi}_t}) \approx h(\boldsymbol{\xi}_t) + \Delta_{\boldsymbol{\xi}_t}^T\nabla h(\boldsymbol{\xi}_t)$. Then, taking derivative of the objective in (1) w.r.t. $\Delta_{\boldsymbol{\xi}_t}$ and equating it to zero yields

$$
\Delta_{\boldsymbol{\xi}_t}^* = \left(\mathbf{H}_t^{(\mathrm{LiD})} - \gamma\mathbf{H}_t^{(\mathrm{UWB})}\right)^{-1}\left(\mathbf{M}_t^{(\mathrm{LiD})} + \gamma\mathbf{M}_t^{(\mathrm{UWB})}\right), \tag{2}
$$

where

$$\mathbf{H}_t^{(\mathrm{LiD})} = \sum_{i=1}^n \left[\nabla\Gamma\left(f_i(\boldsymbol{\xi}_t)\right) \frac{\partial f_i(\boldsymbol{\xi}_t)}{\partial \boldsymbol{\xi}_t} \right] \left[\nabla\Gamma\left(f_i(\boldsymbol{\xi}_t)\right) \frac{\partial f_i(\boldsymbol{\xi}_t)}{\partial \boldsymbol{\xi}_t} \right]^T,$$

$$\mathbf{H}_t^{(\mathrm{UWB})} = \sum_{0 \le i < j \le N_t} \nabla h\left(\hat{\mathbf{p}}_{i,t|t} + \Delta_{\mathbf{p}_{i,t}}, \hat{\mathbf{p}}_{j,t|t} + \Delta_{\mathbf{p}_{j,t}}\right)$$
$$\nabla h\left(\hat{\mathbf{p}}_{i,t|t} + \Delta_{\mathbf{p}_{i,t}}, \hat{\mathbf{p}}_{j,t|t} + \Delta_{\mathbf{p}_{j,t}}\right)^T,$$

$$\mathbf{M}_t^{(\mathrm{LiD})} = \sum_{i=1}^n \left[\nabla\Gamma\left(f_i(\boldsymbol{\xi}_t)\right) \frac{\partial f_i(\boldsymbol{\xi}_t)}{\partial \boldsymbol{\xi}_t} \right] \left[1 - \Gamma\left(f_i(\boldsymbol{\xi}_t)\right) \right],$$

$$\mathbf{M}_t^{(\mathrm{UWB})} = \sum_{0 \le i < j \le N_t} \nabla h\left(\hat{\mathbf{p}}_{i,t|t} + \Delta_{\mathbf{p}_{i,t}}, \hat{\mathbf{p}}_{j,t|t} + \Delta_{\mathbf{p}_{j,t}}\right)$$
$$\left(h\left(\hat{\mathbf{p}}_{i,t|t} + \Delta_{\mathbf{p}_{i,t}}, \hat{\mathbf{p}}_{j,t|t} + \Delta_{\mathbf{p}_{j,t}}\right) - \left[\hat{\mathbf{r}}_t^{(\mathrm{UWB})}\right]_k \right),$$

where k indexes the range measured between node i and node j. Note that the optimal offset $\Delta_{\boldsymbol{\xi}_t}^*$ in (2) generalized the one given in (12) of [2] which considers only the LiDAR map matching.

After obtaining the optimal offset $\Delta_{\boldsymbol{\xi}_t}^* = \left[\left(\Delta_{\mathbf{p}_t}^*\right)^T, \Delta_{\theta_t}^*\right]^T$, the state estimate would be corrected as

$$\hat{\mathbf{p}}_{t|t} = \hat{\mathbf{p}}_{t|t} + \Delta_{\mathbf{p}_t}^*,$$
$$\hat{\mathbf{v}}_{0,t|t} = \hat{\mathbf{v}}_{0,t|t} + \Delta_{\mathbf{v}_t}^*,$$
$$\Delta_{\mathbf{v}_t} = \begin{bmatrix} \|\hat{\mathbf{v}}_{0,t|t}\| \cos\Delta_{\theta_t}^* \\ \|\hat{\mathbf{v}}_{0,t|t}\| \sin\Delta_{\theta_t}^* \end{bmatrix}.$$

V. EXPERIMENTAL RESULTS

We present here the experimental results that show the advantages of UWB/LiDAR fusion in SLAM by comparing our approach with the existing LiDAR-only SLAM approach. The robustness of UWB/LiDAR fusion in SLAM are demonstrated under different scenarios such as 1) the robot moves fast; 2) beacon(s) drop from the sensor network / new beacon(s) join the sensor network / beacons are moving; 3) the environment is feature-less. We also study the impact of γ, which tradeoffs the LiDAR map matching and UWB map matching, on the SLAM performance. All the experiments shown below share some common settings: the standard deviation (std.) of ranging/motion noise are $\sigma_n = 0.1$ meter and $\sigma_w = 0.1$ meter.

A. Hardware

Fig. 3(a) is our designed UWB hardware platform that integrates DWM1000 from Decawave which is used for wireless ranging and messaging. The current implementation uses 4GHz band, 850 kbps data rate at 16MHz PRF. The STM32 is controlled by the intelligence of computer/micro-computer e.g. raspberry pi, beaglebone black, etc., to execute commands given via USB. For one peer-to-peer ranging, it takes 3 milliseconds. The LiDAR sensor we use is Hokuyo UTM-30LX-EW scanning laser rangefinder. The robot i.e. a wheelchair equipped with one UWB sensor and one LiDAR sensor is shown Fig. 3(b). The wheelchair is pushed manually in our experiments.
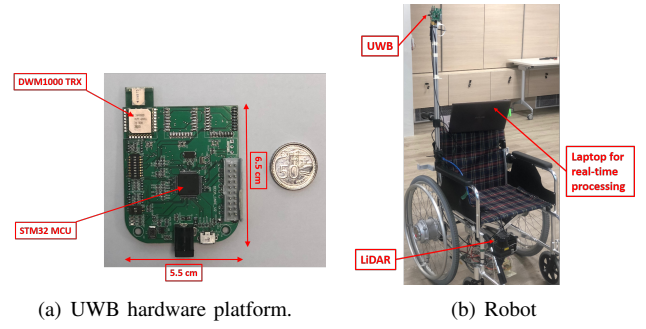

(a) UWB hardware platform.   (b) Robot

Fig. 3. Our UWB/LiDAR SLAM system.

B. SLAM in a workshop of size 12×19 m^2

Five UWB beacons are placed at unknown locations and the robot moves at a speed about 0.8m/s along a predefined trajectory for three loops which takes about 150 seconds. Fig. 4 shows the UWB/LiDAR-based SLAM results. Comparing Fig. 4(a) with Figs. 4(b) to 4(d), we see that when there is no scan matching step the quality of the LiDAR map dramatically decreases. This explains why we cannot build LiDAR map by directly mapping the laser scan endpoints to the UWB map. Comparing Fig. 4(b) and Fig. 4(c) where the former one has no correction step (i.e. correction of the UWB map using the offset obtained in the scan matching procedure) whereas the latter one has, we can see the correction step significantly improves the estimate of robot's trajectory while the estimates of beacons' positions are roughly the same for both cases. It makes sense since the robot's pose is directly affected by LiDAR map matching, i.e. the first term in (1), which however has no direct influence on beacons' poses. Fig. 4(d) ignores the UWB map matching, i.e. the second term in (1), by setting γ to be a negligibly small value 10^{-6}. Comparing Fig. 4(d) with Fig. 4(c), we can see the localization for both robot and beacons are distorted when the role of second term in (1) is downplayed. Table I gives the averaged (over time steps and over five nodes) positioning error and std. of UWB beacons. The estimates of five beacons' poses are recorded while robot is moving. From the table we can see choosing $\gamma = 0.65$ and enabling the correction step gives best results among the three settings. In the following experiments, we empirically choose $\gamma = 0.65$ but we believe the choice of γ is environmentally dependent. A simple rule-of-thumb is that let γ be large when the environments have less features and/or there are sufficient LOS UWB ranges, and let γ be small when the environments are feature-sufficient and/or there are only few LOS UWB ranges.

TABLE I

AVERAGED ERRORS/STDS. OF FIVE UWB BEACONS' POSE ESTIMATES.

	$\gamma = 0.65$ no correction	$\gamma = 0.65$ w/ correction	$\gamma = 10^{-6}$ w/ correction
err. in meters	0.213	0.076	0.206
std. in meters	0.136	0.122	0.149

Fig. 5 demonstrates our system is capable of handling dynamical scenarios where the beacons may drop/join/move.

(a) no scan matching/correction

(b) $\gamma = 0.65$, no correction

(c) $\gamma = 0.65$, w/ correction

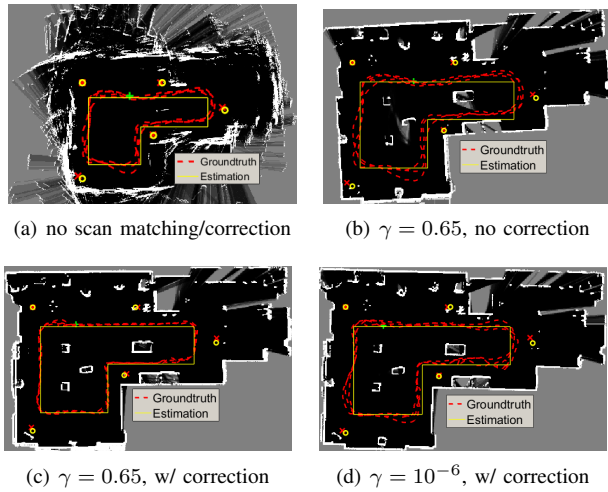
(d) $\gamma = 10^{-6}$, w/ correction

Fig. 4. UWB/LiDAR fusion in four cases: a) no scan matching, no correction; b) with UWB/LiDAR scan matching where $\gamma = 0.65$, no correction; c) with UWB/LiDAR scan matching where $\gamma = 0.65$ and correction; d) with LiDAR-only scan matching where $\gamma = 10^{-6}$ and correction. The green "+" denotes the final position of the robot.

We divide the SLAM process into three time slots: a) For $t < 135$ (135 time steps not 135 seconds), there are four static beacons; b) We power on a new beacon (with ID. 5) at $t = 135$ at the "start pos." shown in Fig. 5(a), then this new beacon is moved along the trajectory of the robot until we place it to the "end pos." at $t = 311$; c) We power off one existing beacon (with ID. 3) at $t = 311$ and relocate it to the "start pos." shown in Fig. 5(b), then power it on at $t = 466$ and move it along the trajectory of the robot until we place it to "end pos." at $t = 626$. Fig. 5(c) shows how the number beacons varies over time steps. In Fig. 6, we compare UWB/LiDAR-based SLAM

(a) $135 \leq t \leq 311$

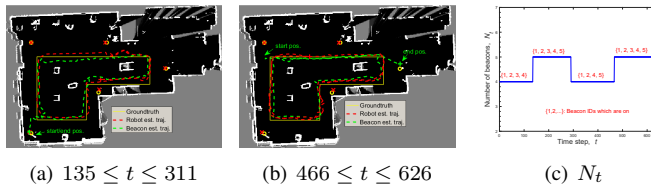(b) $466 \leq t \leq 626$

(c) N_t

Fig. 5. Beacon(s) drops/joins/moves while SLAM is proceeding.

with HectorSLAM (LiDAR-only SLAM) when robot moves at different speeds. When the robot moves at about 0.4m/s, HectorSLAM is working properly and a high-quality map shown in Fig. 6(a) is constructed. However, when the robot moves faster at about 0.8m/s, HectorSLAM becomes error-prone where the errors mainly come from fast turnings as HectorSLAM cannot accommodate a sudden change of robot's behaviour especially the change of heading, and these errors would affect the subsequent scan matching steps thus being accumulated over time. This can be seen in Fig. 6(b). Such errors can be dramatically reduced due to additional inputs from UWB sensors. In Fig. 6(c) where robot moves as fast as the one in Fig. 6(b), we observe no drift-errors in UWB/LiDAR-based SLAM.


(a) HectorSLAM at 0.4m/s   (b) HectorSLAM at 0.8m/s   (c) UWB/LiDAR at 0.8m/s

Fig. 6. Comparison of our UWB/LiDAR-based SLAM with HectorSLAM [2] at different robot's speeds. To build the maps, the robot moves for one loop of the same trajectory as shown in Fig. 4.

C. SLAM in a corridor of length 22.7 meters

HectorSLAM is known to work poorly in feature-less environments such as corridor and Fig. 7(a) shows such an example where the length of corridor in its map is 8.43m whereas the actual length is 22.7m. Fusing LiDAR with UWB, we may regard UWB sensors as additional "features" that provide precise information about robot's location thus robustifying SLAM in such feature-less environments at the cost of slight degradation of map quality as shown in Fig. 7(b).
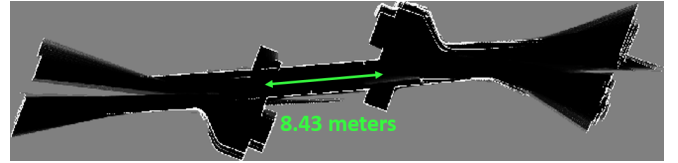
(a) HectorSLAM
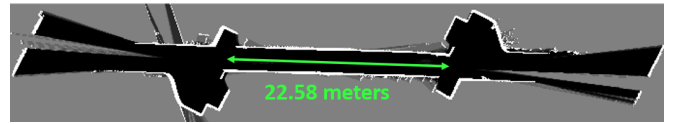
(b) UWB/LiDAR SLAM

Fig. 7. Comparison of our UWB/LiDAR-based SLAM with HectorSLAM [2] in a corridor of length 22.7m.

Remark 6. *The quality of the map that UWB/LiDAR-based SLAM builds is compromised due to the tradeoff between UWB and LiDAR ranging accuracy which is reflected from (1). Moreover, to balance the UWB and LiDAR ranging accuracy, we add some Gaussian noise with zero mean and std. of 1cm/2cm to the laser ranger finder.*

VI. CONCLUSION

We have proposed a fusion scheme that utilizes both UWB and laser ranging measurements for simultaneously localizing the robot, the UWB beacons and constructing the LiDAR map. This is a 2D range-only SLAM and no control input is required. In our fusion scheme, a "coarse" map i.e. UWB map is built using UWB peer-to-peer ranging measurements. This "coarse" map can guide where the "fine" map i.e. LiDAR map should be built. Through a new scan matching procedure, the LiDAR map can be constructed while the UWB map is polished. Our proposed system suits infrastructure-less and ad-hoc applications where no prior information about the environment is known and the system needs to be deployed quickly.

REFERENCES

[1] G. Grisettiyz, C. Stachniss, and W. Burgard, "Improving grid-based SLAM with Rao-Blackwellized particle filters by adaptive proposals and selective resampling," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2005, pp. 2432–2437.

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

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

[4] C. Wang, H. Zhang, T.-M. Nguyen, and L. Xie, "Ultra-wideband aided fast localization and mapping system," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 1602–1609.

[5] J. Djugash, S. Singh, G. Kantor, and W. Zhang, "Range-only SLAM for robots operating cooperatively with sensor networks," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2006, pp. 2078–2084.

[6] J.-L. Blanco, J. Gonzalez, and J.-A. Fernandez-Madrigal, "A pure probabilistic approach to range-only SLAM," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2008.

[7] T. Deißler and J. Thielecke, "UWB SLAM with Rao-Blackwellized Monte Carlo data association," in *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2010.

[8] G. Vallicrosa, P. Ridao, D. Ribas, and A. Palomer, "Active range-only beacon localization for AUV homing," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2014, pp. 2286–2291.

[9] F. Herranz, A. Llamazares, E. Molinos, and M. Ocaña, "A comparison of SLAM algorithms with range only sensors," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 4606–4611.

[10] L. Génévé, O. Kermorgant, and É. Laroche, "A composite beacon initialization for EKF range-only SLAM," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015, pp. 1342–1348.

[11] J. Tiemann, A. Ramsey, and C. Wietfeld, "Enhanced UAV indoor navigation through SLAM-augmented UWB localization," in *IEEE International Conference on Communications Workshops (ICC Workshops)*, 2018.

[12] C. Gentner, M. Ulmschneider, and T. Jost, "Cooperative simultaneous localization and mapping for pedestrians using low-cost ultra-wideband system and gyroscope," in *IEEE/ION Position, Location and Navigation Symposium (PLANS)*, no. 1197-1205, 2018.

[13] J. Djugash and S. Singh, "A robust method of localization and mapping using only range," *Experimental Robotics*, vol. 54, pp. 341–351, Jul. 2008.

[14] E. Menegatti, A. Zanella, S. Zilli, F. Zorzi, and E. Pagello, "Range-only SLAM with a mobile robot and a wireless sensor networks," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2009.

[15] P. Fritsche and B. Wagner, "Modeling structure and aerosol concentration with fused radar and LiDAR data in environments with changing visibility," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 2685–2690.

[16] W. Maddern and P. Newman, "Real-time probabilistic fusion of sparse 3D LIDAR and dense stereo," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 2181–2188.

[17] K. Li, C. Wang, S. Huang, G. Liang, X. Wu, and Y. Liao, "Self-positioning for UAV indoor navigation based on 3D laser scanner, UWB and INS," in *IEEE International Conference on Information and Automation (ICIA)*, 2016.

[18] F. J. Perez-Grau, F. Caballero, L. Merino, and A. Viguria, "Multi-modal mapping and localization of unmanned aerial robots based on ultra-wideband and RGB-D sensing," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 3495–3502.

[19] X. R. Li and V. P. Jilkov, "A survey of maneuvering target tracking: dynamic models," in *SPIE Conference on Signal and Data Processing of Small Targets*, 2000.

[20] H. Wymeersch, J. Lien, and M. Z. Win, "Cooperative localization in wireless networks," *Proceedings of the IEEE*, vol. 97, no. 2, pp. 427–450, Feb. 2009.

[21] D. Dardari, A. Conti, U. Ferner, A. Giorgetti, and M. Z. Win, "Ranging with ultrawide bandwidth signals in multipath environments," *Proceedings of the IEEE*, vol. 97, no. 2, pp. 404–426, Feb. 2009.

[22] S. Marano, W. M. Gifford, H. Wymeersch, and M. Z. Win, "NLOS identification and mitigation for localization based on UWB experimental data," *IEEE Journal on Selected Areas in Communications*, vol. 28, no. 7, pp. 1026–1035, Aug. 2010.

[23] K. Gururaj, A. K. Rajendra, Y. Song, C. L. LAW, and G. Cai, "Real-time identification of NLOS range measurements for enhanced UWB localization," in *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2017.

[24] *https://www.decawave.com/sites/default/files/resources/dwm1000-datasheet-v1.3.pdf*.

[25] J. D. Hol, F. Dijkstra, H. Luinge, and T. B. Schon, "Tightly coupled UWB/IMU pose estimation," in *IEEE International Conference on Ultra-Wideband*, 2009.

[26] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proceedings of the 7th international joint conference on Artificial intelligence, Vancouver, BC, Canada*, Aug. 1981, pp. 674–679.