Inspection of Penstocks and Featureless Tunnel-like Environments using Micro UAVs

Tolga Özaslan, Shaojie Shen, Yash Mulgaonkar, Nathan Michael and Vijay Kumar

Abstract Micro UAVs are receiving a great deal of attention in many diverse applications. In this paper, we are interested in a unique application, surveillance for maintenance of large infrastructure assets such as dams and penstocks, where the goal is to periodically inspect and map the structure to detect features that might indicate the potential for failures. Availability of architecture drawings of these constructions makes the mapping problem easier. However large buildings with featureless geometries pose a significant problem since it is difficult to design a robust localization algorithm for inspection operations. In this paper we show how a small quadrotor equipped with minimal sensors can be used for inspection of tunnel-like environments such as seen in dam penstocks. Penstocks in particular lack features and do not provide adequate structure for robot localization, especially along the tunnel axis. We develop a Rao-Blackwellized particle filter based localization algorithm which uses a derivative of the ICP for integrating laser measurements and IMU for short-to-medium range pose estimation. To our knowledge, this is the only study in the literature focusing on localization and autonomous control of a UAV in 3-D, featureless tunnel-like environments. We show the success of our work with results from real experiments.

1 Introduction

Recently, micro UAVs have attracted significant attention in a variety of civilian applications due to their low cost and superior mobility. One possible application is

Tolga Özaslan, Shaojie Shen, Yash Mulgaonkar and Vijay Kumar
University of Pennsylvania, 19104, Philadelphia, PA,
Nathan Michael
Carnegie Mellon University, 15213, Pittsburgh, PA,
e-mail: {ozaslan}{shaojie}{yashm}{kumar}@seas.upenn.edu, nmichael@cmu.edu

the use UAVs in inspection of large buildings such as dams and penstocks. Penstocks are constructions that require regular maintenance, and this in turn requires visual inspection of the interior. However penstocks are dark, long and featureless tunnels that slope steeply down hillsides. Because of this, it is hard for humans to climb up penstocks and perform visual inspection. We propose, as an alternative, the use of quadrotors equipped with minimal sensors that can fly through the tunnels and collect data for remote inspection.

In order to reduce the operator workload, we require a high level of autonomy of the quadrotor. This in turn requires that the robot is able to localize itself with respect to features in the environment. In our case, we are given engineering drawings of the penstock, which can be converted into a map of the environment. Hence we focus on solving the problem of pose estimation (localization) and autonomous control in penstocks (tunnel-like) buildings. This work allows us to build autonomous UAVs that can collect imagery from inside penstocks for inspection with only high-level user commands.

Penstocks are almost perfectly cylindrical in cross section, and have two long, non-parallel, straight portions as shown in Fig. 2. In most penstocks, the first portion is on a horizontal plane and the second part slopes upwards. The interior of the penstock is built with rectangular shaped steel plates of approximately 6 square meters each bent into a cylindrical geometry. Using the given engineering drawings (the map), IMU data and scanner readings, it is always possible to determine the orientation, height and lateral position of the quadrotor. However the position along the tunnel axis cannot be always derived due to the special geometry of the map and the available sensors. For example, when the distance between the robot and the junction of the tunnel is greater than the maximum measurement range of the laser scanner, the position along the tunnel cannot be determined. However, during the transition between the horizontal and inclined portions of the tunnel, scanner readings show significant differences which can help in localizing the robot along the axis of the tunnel at that particular region.

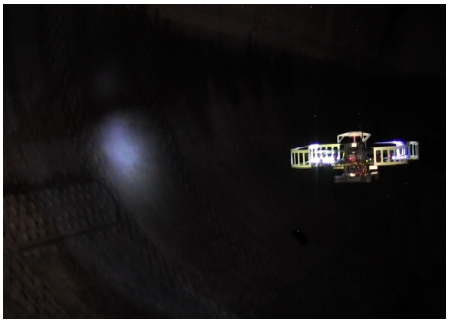


Fig. 1 A quadrotor flying inside a penstock at Allatoona Dam, GA. Lighting is provided by a portable spot light. The quadrotor is equipped with a 1.6 GHz Atom Intel processor, Hokuyo [2] laser scanner and an IMU unit. Note that the installation of lights for illuminating the entire tunnel is impractical. Therefore, we equip our quadrotor with LED lights as shown in the figure.

We stress that it is difficult for ground robots to operate in the tunnel. While a slope of 23 degrees (see Fig. 2) can be easily negotiated by tracked vehicles, the tunnel is very slippery and smooth and it is difficult to use vehicles that require traction in the tunnel. Furthermore, since the tunnel is not illuminated, inspection of ceilings and walls can be difficult and may require auxiliary lighting equipment

which can be heavy and require a lot of power. Fortunately, since our quadrotor can fly close to the walls (and the ceiling) it can use energy efficient LED lights and obtain illumination for collecting imagery. In addition to lights, the platform can be equipped with different inspection sensors such as infrared cameras.

As shown in Fig. 1, we customize a Pelican quadrotor by Ascending Technologies [1]. The robot is equipped with a 1.6 GHz Atom Intel processor, a Hokuyo [2] laser scanner, an IMU unit and LED lights (Fig. 3).

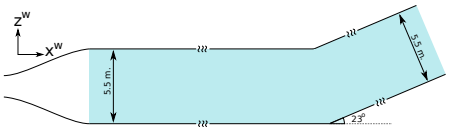


Fig. 2 Side view of a representative penstock (exact diameter and slope from the Carter Dam penstock).

A hallmark reference [7] introduced the framework of Monte Carlo Localization (MCL). Variants of such localization algorithms can be seen for museum guide robots [21] , human operated backpack [16] and robot with 3-D laser scans [15]. However, the highly symmetric and feature-less tunnel environment poses problems for existing localization algorithms. Furthermore, processing a large amount of data using low power, light weight on-board computer proves to be challenging. Also, algorithms relying on GPS are not practical for quadrotors flying inside a tunnel.

There is also extensive literature on localization using cameras. [3] fuses stereo vision, GPS, and IMU to perform outdoor localization. In another outdoor localization study, [14] tests image-based localization with wide angle cameras. Scale-invariant features are used in [19] to both localize and build 3D maps of office environments. However, it is hard to apply this method in real-time due to the limited on-board computation. Further, none of the above approaches will work in a penstock due to poor lighting conditions. In our case, although we use lighting, we need it only for detecting rust and cracking in the interior surface and not for localization.

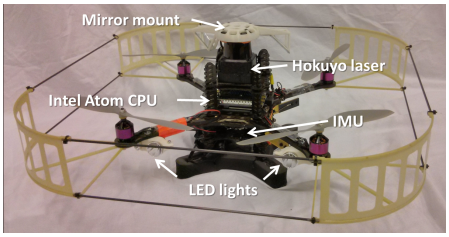


Fig. 3 Our quadrotor prototype equipped with a Hokuyo laser scanner, on-board IMU and two flash lights. A 3-D printed laser mount redirects some of the laser beams upward and downward.

The rest of the paper is organized as follows: We start by reviewing basic background in Section 2. We then present our system for localization of a quadrotor in the penstock in Section 3. The key contributions in this paper are the novel measurement models those are designed based on the unique geometry of the penstock and semi-autonomous operation in featureless tunnels. These are both presented in Section 3. Finally, field experimental results are presented in Section 4.

2 Background

2.1 Quadrotor Dynamics

Quadrotors are basically helicopters with four propellers located at corners of a square shape. A schematic of a quadrotor is given in Fig. 4. Each propeller is located at equal distances from the geometric center of the quadrotor. Motors mounted on opposite sides rotate in the same direction, while the others in the opposite direction. Ideally, while the quadrotor is stationary, moments due to the propellers rotating in opposite directions cancel each other so that the yaw is kept constant.

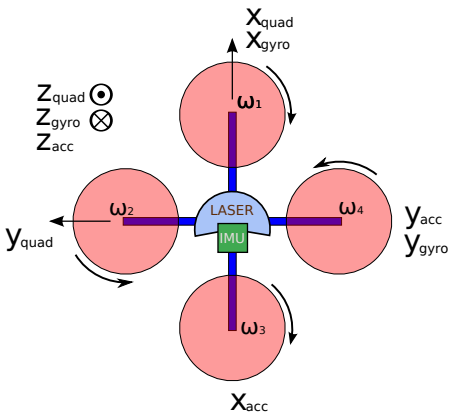


Fig. 4 Coordinate frame definitions of quadrotor. Due to the manufacturer, gyroscope and accelerometer has different orientations which are shown with subscripts of *gyro* and *acc*. And body frame is denoted by the subscript *quad*. Dot in a circle means a vector pointing out of the paper plane and an $\times$ means the opposite [17].

As the standard reference triad (SRT) for inertial frame, we use $\{\hat{x}^{\mathcal{W}}, \hat{y}^{\mathcal{W}}, \hat{z}^{\mathcal{W}}\}$ basis vectors. Then a vector in this frame is represented by the vector $\left[x^{\mathcal{W}}, y^{\mathcal{W}}, z^{\mathcal{W}}\right]^T$. Whereas SRT of the body frame is defined with the basis vectors $\{\hat{x}^{\mathcal{B}}, \hat{y}^{\mathcal{B}}, \hat{z}^{\mathcal{B}}\}$ and a vector in this frame is represented as $\left[x^{\mathcal{B}}, y^{\mathcal{B}}, z^{\mathcal{B}}\right]^T$. $\hat{x}^{\mathcal{B}}$ is the heading direction of the quadrotor which can be selected arbitrarily. $\hat{z}^{\mathcal{B}}$ is preferably selected as the upwards direction when the quadrotor is hovering and $\hat{z}^{\mathcal{W}}$ is selected to be pointing in the opposite direction to the gravitation (see Fig. 4 and its caption for illustration). Rotation between these two frames is carried through multiplication with a rotation matrix $R \in SO(3)$ and denoted by ${}^{\mathcal{B}}R_{\mathcal{W}}$. Subscript is the frame from which the vector will be transformed and the pre-superscript is the goal frame.

We use $Z-X-Y$ Euler angles to represent rotation from world to body frame [18]. Yaw, pitch and roll angles are denoted as ψ, θ and ϕ respectively. Angular velocity in the body frame is denoted by the vector $\left[p, q, r\right]^{\mathcal{B}}$

We refer to the work by Mellinger [18] where detailed derivations of dynamic equations are given. They also linearize about the hover state and present a linear controller based on this model.

2.2 Robot Localization

Robot localization, environment mapping and the merging of these two problems, Simultaneous Localization and Mapping (SLAM), has been studied extensively [4, 5, 8, 9, 20, 21]. Filtering based approaches are commonly used for solving the localization problem. Two mostly used approaches are based on the Kalman filter and the particle filter.

For systems that satisfy the Gaussian uncertainty model, the Kalman filter and its nonlinear variants (referred to as KF from this point on) yield efficient and robust results. We choose the Unscented Kalman Filter (instead of the standard Kalman filter) due to its ability to approximate the propagation of Gaussian random vectors through nonlinear functions via the propagation of stochastic linearization points [20].

On the other hand, there are many systems with multi-modal, widely spread, and other uncertainty models that are cannot be modeled as Gaussian distributions. For such distributions, the nonparametric particle filter-based approach and variants (referred to as PF from this point on), also known as Monte Carlo methods [7, 21], provide approximate representations of arbitrary probabilistic distributions. They are more powerful compared to the the parametric KF-based approaches. However, for systems with relatively large number of degrees of freedom (such as quadrotors), the number of particles that is required to accurately represent the distribution can be prohibitively large.

The Rao-Blackwellized particle filter decomposes the configuration space in order to reduce the dimension of the particle-based distribution approximation. The main goal is to reduce the required particle count for the particle filter [8, 12] by designing a hybrid filter achieved by merging the PF and the KF. That is, for some of the parameters, estimation is done through KF and for others PF is used. In our application, since a robot moves through a featureless tunnel, the localization uncertainty for the position along the axis of the tunnel is high and it is hardly a proper Gaussian distribution. However , the uncertainties in position for the other two directions are small and they can be well approximated by Gaussian distributions. For the former case, use of PF is meaningful and in the latter case KF is a reasonable choice.

2.3 Controller Design

We use the linear controller design of [18]. Since our target application requires mostly stable flight with minimum linear acceleration, linearization of dynamic equations around the hover position can be justified. Our controller utilizes a backstepping architecture that consists of a position controller and an attitude controller. The high level position controller generates desired orientations based on user specified way-points and the on-board localization feedback. The low level attitude controller drives the robot to the desired orientation by adjusting motor RPMs.

As shown in Fig. 6, a trajectory generator is used to generate a trajectory from the current pose to the goal pose. At this level we can also incorporate constraints such as closest distance to walls, maximum linear and rotational speeds, and other constraints.

3 Methodology

3.1 Process and Observation Models

We define the process model with the equation

$$\mathbf{x}_{t+\Delta t} = f(\mathbf{x}_t, \mathbf{u}_t, \Delta t) \tag{1}$$

where $\mathbf{x}$ is the state vector and $\mathbf{u}$ is the control input derived from IMU. Vectors $\mathbf{x}$ and $\mathbf{u}$ are defined as:

$$\mathbf{x}^T = [x, y, z, \dot{x}, \dot{y}, \dot{z}, \psi, \theta, \phi]^{\mathcal{W}}, \tag{2}$$

$$\mathbf{u}^T = [\ddot{x}, \ddot{y}, \ddot{z}, p, q, r]^{\mathcal{B}}. \tag{3}$$

The process model implements dynamics of a quadrotor which have detailed explanations in [18]. As it is the case for MEMs sensors, our IMU has both bias and random errors. Then the true IMU data becomes

$$\mathbf{u}^* = \mathbf{u} - \mathbf{u}_{bias} - \mathbf{u}_{rnd} \tag{4}$$

where $\mathbf{u}_{rnd}$ is a random vector drawn from a normal distribution and $\mathbf{u}_{bias}$ is the bias error. The process noise in the $\hat{x}^{\mathcal{W}}$ direction is modeled by an additive random disturbance which is distributed normally with known variance.

We are using a Hokuyo laser scanner [2] which can take measurements with a 180 degrees span in the $x^{\mathcal{B}} - y^{\mathcal{B}}$ plane. A 3-D printed dual-mirror mount is fixed on top of the laser scanner to reflect rays in upward ($+\hat{z}^{\mathcal{B}}$) and downward ($-\hat{z}^{\mathcal{B}}$) directions [13] (Fig. 3). These measurements together with the orientation estimate and the knowledge of the map are used to localize robot on the $y^{\mathcal{W}} - z^{\mathcal{W}}$ plane of the tunnel using a derivative of the ICP algorithm. This algorithm uses rays emanating in the four directions $\pm\hat{z}^{\mathcal{W}}$ and $\pm\hat{y}^{\mathcal{W}}$. Note that no rays might be exactly in these directions due to the orientation of the robot, in case which we select the closest rays. In following explanations we will call these vectors with $\mathbf{u}^{\mathcal{W}}, \mathbf{d}^{\mathcal{W}}, \mathbf{r}^{\mathcal{W}}$ and $\mathbf{l}^{\mathcal{W}}$ which refer to laser beams closest to the upwards, downwards, rightwards and leftwards directions in the world frame.

We do ray-casting to determine the intersections of the above four sets of vectors ($\mathbf{u}^{\mathcal{W}}, \mathbf{d}^{\mathcal{W}}, \mathbf{r}^{\mathcal{W}}$ and $\mathbf{l}^{\mathcal{W}}$) with the map. We call these as $\mathbf{u}_c^{\mathcal{W}}, \mathbf{d}_c^{\mathcal{W}}, \mathbf{r}_c^{\mathcal{W}}$ and $\mathbf{l}_c^{\mathcal{W}}$. Casting is done against an occupancy grid map with resolution of 5 cm. After ray-casting, we update robot $y^{\mathcal{W}}$, $z^{\mathcal{W}}$ positions such that the discrepancy between the measured

rays and the casted rays reduces. A snapshot of this procedure is illustrated in Fig. 7. Also Algorithm 1 explains this method. Due to the convexity of the tunnel cross-section, this algorithm is guaranteed to converge to the correct position.

The on-board attitude estimator supplies roll and pitch data with drift correction; but the yaw needs to be corrected using the laser because the IMU cannot measure the global yaw angle. However, due to the metal interior of the tunnel, we cannot use the magnetometer output as a global reference to the yaw angle. For this reason, we estimate the yaw angle with respect to the tunnel using laser scans.We propose a geometric solution to this problem using the fact that intersection of a cylinder (tunnel) and a plane is always an ellipse. It is easy to see that the intersection of a plane with a cylindrical tunnel can result in three different curves which are circle, ellipse and two parallel lines. This curve is a circle only when $\hat{x}^{\mathcal{W}}$ and $\hat{z}^{\mathcal{B}}$ are aligned, which is very unlikely to happen in our case. Other two cases are more likely to be observed and both can be treated as an ellipse since two parallel lines correspond to the special case of an ellipse with infinite major axis length. So we fit an ellipse to scans and then orientation of the major axis gives negative of the yaw angle up to π radians ambiguity. While we define $\psi = 0$ to be the case when $+\hat{x}^{\mathcal{W}}$ and $+\hat{x}^{\mathcal{B}}$ are coincident, the source of ambiguity is due to the lack of any clues to distinguish whether a scan is taken when robot's heading is $\psi = \psi_0$ or $\psi = \psi_0 + \pi$. In both cases the curve due to the laser has the exact same shape. We choose the yaw measurement that is closest to the current UKF yaw estimate for measurement update.

As seen in Fig. 5, laser data can be noisy due to unmodeled obstacles in the environment, inherent noise in the laser scanner and complete failures. A direct fit to such data is very probable to give wrong estimates which we experienced several times during experiments in development stage and caused crashes. In order to get rid of this problem, we use RANSAC [10] which obviously improves fit quality. Since we do not make fast maneuvers, we make a reasonable assumption that quadrotor is almost in hover state, in other words $\phi \approx 0$ and $\theta \approx 0$. Otherwise resultant ellipse fit would also reflect the effect of non-zero ϕ and θ angles and we would need to decouple these effects to obtain the actual yaw angle. We leave the details of ellipse fitting algorithm to [11].

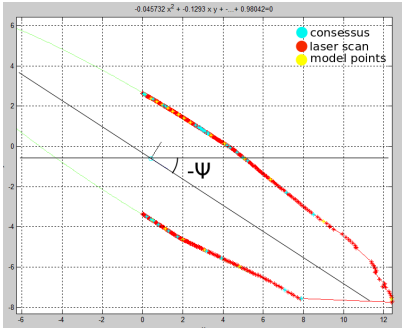


Fig. 5 A sample laser scan data. Ellipse is fit using the method in [11]. In order to eliminate outliers, we use RANSAC. Outliers are due to operators moving together with the quadrotor, noise and laser failures.

3.2 Rao-Blackwellized Particle Filter Design

In this model, we carry the well-known UKF prediction using the IMU output. Measurement updates for positions and velocities in the $y^{\mathcal{W}} - z^{\mathcal{W}}$ directions, as well as the roll, pitch, and yaw orientation are performed within the UKF framework as well.

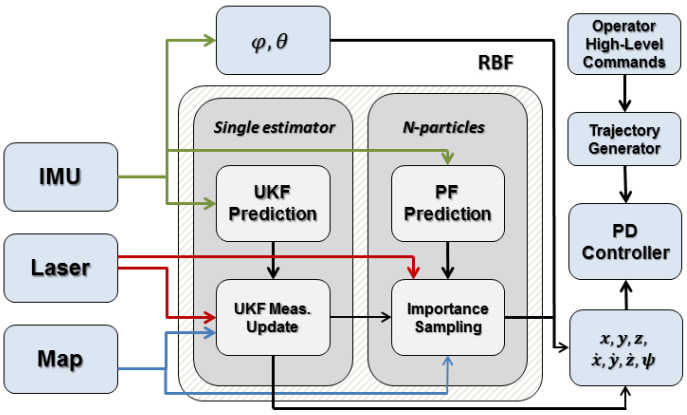


Fig. 6 The estimator based on the Rao Blackwell Filter and the PD controller for autonomous flight in a tunnel of known cross section. A particle filter with N particles is used to model the propagation of state estimates and the uncertainty in the $\hat{x}^{\mathcal{W}}$ direction, while a UKF is used to estimate the remaining states.

A particle filter is used to estimate $x^{\mathcal{W}}$ position of robot (Fig. 6) . That is, during the prediction step of UKF we make use of gyroscope and accelerometer data and in the measurement update stage we integrate information from the measurement models above. The reason using orientation information from the IMU twice which are the gyroscope data (angular velocity) and the on-board roll-pitch estimation, is because of the computational constraints. IMU supplies estimates (roll and pitch) at a rate of 100Hz which we know to be reliable due to the drift correction. But making measurement updates at this rate consumes valuable CPU time. Instead we integrate them at the same rate of laser scanner (30Hz) and carry the low-cost prediction update at 100Hz using the gyroscope data (angular velocity). Note that running a measurement update (UKF update) requires calculation of matrix square root which is of complexity $O(n^3)$. With our current setup, we have chosen not to spend CPU power with frequent measurement updates.

The overall system design is shown in Fig. 6. We run a particle filter for estimating the position and velocity along $\hat{x}^{\mathcal{W}}$ and an UKF common to all particles to estimate the remaining state variables which are $y^{\mathcal{W}}$, $z^{\mathcal{W}}$ and their derivatives and the three Euler angles, ψ, θ, ϕ. The inputs are data from the laser scanner, the IMU and a grid map. Unless we are close to the junction region of the horizontal and inclined

portions of the tunnel, we don't have measurements to estimate $x^{\mathcal{W}}$. This implies that in such cases uncertainty along this direction can be in any form which may not be have a closed-form representation. However for all the other states, including lateral and vertical positions and orientation, we always have laser measurements. We expect a unimodal uncertainty model for these states and use the UKF to estimate them.

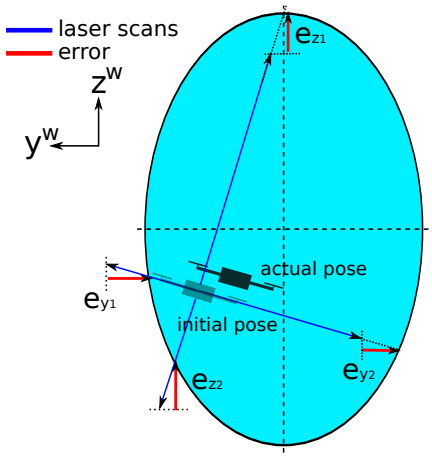


Fig. 7 Starting from an initial pose, ICP iteratively refines $y^{\mathcal{W}} - z^{\mathcal{W}}$ positions to reduce discrepancy between laser data and robot pose. Red vectors are the error vectors to be minimized. Although in the horizontal region of the tunnel cross-section is circular, in inclined region it will be an ellipse as seen by the robot.

When the robot is away from the junction region of the two portions of the penstock, laser scanner cannot make any readings since the closest wall is farther than the maximum range of the laser scanner. This invalidates the measurement model explained for $y^{\mathcal{W}} - z^{\mathcal{W}}$ estimation. Instead we use Algorithm 2 as the measurement model to calculate the weight for each particle. When there are valid measurements, particles consistent with them will be given more importance hence will survive in the importance sampling. Otherwise all particles are given the same weight and importance sampling favors them equally. As we get consecutive measurement failures, distribution of the particles spread out widely according to the IMU noise model. Note the power in representing arbitrary distributions with particles is obviously not achievable with a Gaussian assumption.

In Algorithm 2, to find the weight of a particle, similar to what we do in Algorithm 1, we define a set of vectors, $\mathbf{f}^{\mathcal{W}}$, which are the closest laser beams to $\hat{x}^{\mathcal{W}}$ direction. Then we cast these vectors against the grid map to obtain $\mathbf{f}_c^{\mathcal{W}}$. The weight of a particle is the reciprocal of $|(\mathbf{f}^{\mathcal{W}} - \mathbf{f}_c^{\mathcal{W}})_x|^2$. In case we don't have a valid reading, we assign a non-significant weight.

Depending on the availability of valid laser measurements along the axis of the tunnel, we constrain the regions to resample particles in. In case of valid measure-

Algorithm 1 $[h_{yz}, \Sigma_{yz}]$ =measurement_model_yz($laser, map$)

$iter \leftarrow 0$

$\mathbf{r}^{\mathcal{W}} \leftarrow$ get_beams_in_dir($-\hat{y}^{\mathcal{W}}, laser$) ; $\mathbf{l}^{\mathcal{W}} \leftarrow$ get_beams_in_dir($+\hat{y}^{\mathcal{W}}, laser$)

$\mathbf{u}^{\mathcal{W}} \leftarrow$ get_beams_in_dir($+\hat{z}^{\mathcal{W}}, laser$) ; $\mathbf{d}^{\mathcal{W}} \leftarrow$ get_beams_in_dir($-\hat{z}^{\mathcal{W}}, laser$)

while $(err_x > thres \wedge err_y > thres) \vee iter < iter_{max}$ **do**

$\mathbf{u}_c^{\mathcal{W}} \leftarrow$raycast($\mathbf{u}^{\mathcal{W}}, map$); $\mathbf{d}_c^{\mathcal{W}} \leftarrow$raycast($\mathbf{d}^{\mathcal{W}}, map$)

$\mathbf{l}_c^{\mathcal{W}} \leftarrow$raycast($\mathbf{l}^{\mathcal{W}}, map$); $\mathbf{r}_c^{\mathcal{W}} \leftarrow$raycast($\mathbf{r}^{\mathcal{W}}, map$)

$err_y \leftarrow (\mathbf{l}_{c,y}^{\mathcal{W}} - \mathbf{l}_y^{\mathcal{W}}) + (\mathbf{r}_{c,y}^{\mathcal{W}} - \mathbf{r}_y^{\mathcal{W}})$; $err_z \leftarrow (\mathbf{u}_{c,z}^{\mathcal{W}} - \mathbf{u}_z^{\mathcal{W}}) + (\mathbf{d}_{c,z}^{\mathcal{W}} - \mathbf{d}_z^{\mathcal{W}})$

$\mathbf{p}_y^{\mathcal{W}} \leftarrow \mathbf{p}_y^{\mathcal{W}} + 1/2 err_y$; $\mathbf{p}_z^{\mathcal{W}} \leftarrow \mathbf{p}_z^{\mathcal{W}} + 1/2 err_z$

$iter \leftarrow iter + 1$

end while

$h_{yz} = \mathbf{p}_{y,z}$

$$\Sigma = Q^T \begin{bmatrix} err_x^2 & 0 \\ 0 & err_y^2 \end{bmatrix} Q$$

$Q^T \Sigma Q$ transforms residual errors of ICP to its corresponding covariance matrix [6]

ments, resamping is done only in the region close to the junction. Similarly, failure of laser implies robot is away from the junction and particles close to the junction are eliminated.

Algorithm 2 $[w_x]$ =measurement_model_x($laser, map$)

if $laser$ is not valid **then**

$w_x \leftarrow 1/\sigma^2$

else

$\mathbf{f}^{\mathcal{W}} \leftarrow$ get_beams_in_dir($+\hat{x}^{\mathcal{W}}, laser$)

$\mathbf{f}_c^{\mathcal{W}} \leftarrow$raycast($\mathbf{f}^{\mathcal{W}}, map$)

$w_x \leftarrow 1$ / $|\mathbf{f}_{c,x}^{\mathcal{W}} - \mathbf{f}_x^{\mathcal{W}}|^2$

end if

3.3 Control

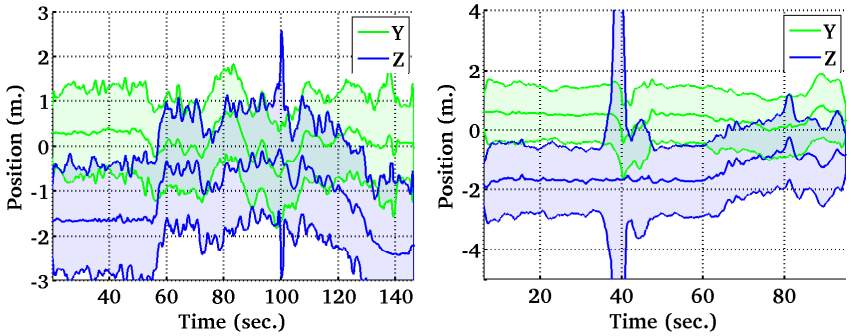
The errors in localization exhibit anisotropy. They are significant in the position coordinate along the axis of the tunnel but more constrained in the other directions.

Accordingly we advocate a semi-autonomous control scheme where the the operator goals (or goals from a planner) prescribe the yaw angle, lateral and vertical positions along the cross section of the tunnel, while the control along the axis of the tunnel is performed by the operator by directly commanding the acceleration through a joystick.

4 Experimental Work

In this section we present and interpret results of our experimental work. Data for the experimental work is collected in three different sites: the Carter Dam and the Allatoona Dam, both in Georgia, and in a long building hallway at the University of Pennsylvania.

In the visit to the Carter Dam, two datasets were collected. In the first flight the quadrotor traversed along the horizontal part of the penstock. And in the second dataset, it flew close to the junction region towards the inclined region. During these tests, the quadrotor was controlled manually. The proposed localization algorithm was run off-line using collected data sets.



(a) Experiment #1 in Allatoona Dam (b) Experiment #2 in Allatoona Dam

Fig. 8 These figures show estimation outputs for $y^{\mathcal{W}}$-$z^{\mathcal{W}}$ positions together with covariances as shaded regions. In these experiments quadrotor flew semi-autonomously. Due to reflective surfaces, laser scanner failed to return readings along $\hat{x}^{\mathcal{W}}$ direction. So we cannot estimate position along this direction. Failure was due to the distance to the junction region, wet surface and oblique surface w.r.t. ray direction.

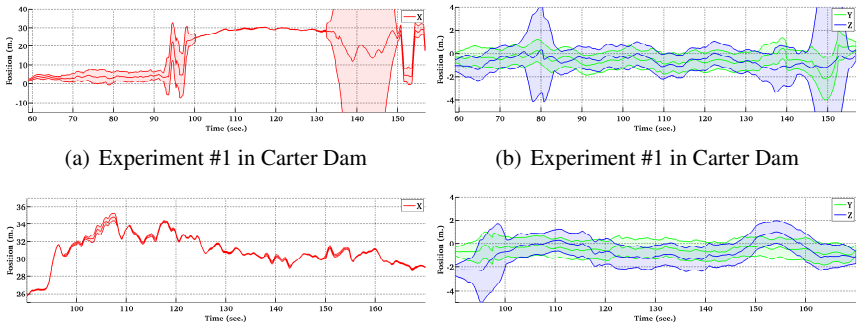
Two semi-autonomous flights were conducted in the Allatoona Dam. The operator sets the desired $y^{\mathcal{W}}$, $z^{\mathcal{W}}$, and ψ through a radio controller. Then feedback control of these parameters is carried out by our controller. The operator controls the acceleration along the $\hat{x}^{\mathcal{W}}$ direction. We believe semi-autonomy proves accuracy and stability of our estimator along the tunnel cross section. Otherwise, as opposed to a

ground robot, faults in controller or estimator would cause unrecoverable instabilities.

We conducted a third experiment in a building at the University of Pennsylvania, along a 42 meters long corridor while the quadrotor flew semi-autonomously. In the corridor experiment, although there are features, such as pillars and doors, the map we are using is a featureless rectangular prism. So there is no feature in our map that would help in estimating the $x^{\mathcal{W}}$ position. Actually those features behave as noise for yaw estimation which shows robustness of our estimator.

In Fig. 8-9-10 we give results for our Allatoona Dam, Carter Dam and university building experiments respectively. These experiments show quadrotors can be considered as a reasonable choice for inspection of tunnel-like environments. Only with a laser scanner and an IMU, as a requirement for semi-autonomy, localization along the cross-section of the tunnel can be achieved robustly. Also, when one end of the tunnel is in the range of laser scanner, localization along the tunnel axis is achieved as well.

In Fig. 8(b) at 40^{th} seconds, increase in the covariance is due to a worker walking near the quadrotor. However, we can handle such cases and estimated position is not affected. In Fig. 9(a)-10(a), periods when the covariance gets larger is when the robot is away from the end of the tunnel/corridor with the following exceptions. In Fig. 9(a) around 160^{th} seconds increase in uncertainty is due to failure of laser scanner due to water drainage behaving as a mirror. And increase in variance in Fig. 10(a) around $100^{th} - 120^{th}$ and 160^{th} seconds is because quadrotor was tilted and laser scanner sees the floor. Since the floor is tiled with marble, it behaves as a mirror and laser scanner fails.



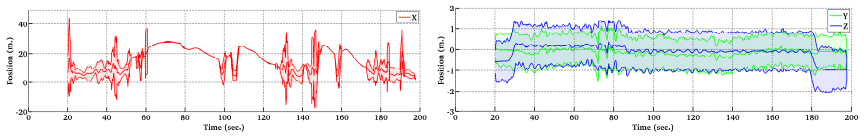
(a) Experiment #1 in Carter Dam

(b) Experiment #1 in Carter Dam

(c) Experiment #2 in Carter Dam

(d) Experiment #2 in Carter Dam

Fig. 9 These figures show estimation outputs for $x^{\mathcal{W}}$-$y^{\mathcal{W}}$-$z^{\mathcal{W}}$ positions together with covariances as shaded regions. Opposed to Allatoona Dam tests (see Fig. 8), since the walls of the penstock was not wet and reflective, we could get readings from the junction region of the tunnel. In Fig. 9(a) we can see that during a period of the flight we are able to localize along $\hat{x}^{\mathcal{W}}$ direction. In the second experiment we flew the quadrotor close to the junction region and have less time periods without valid readings along $\hat{x}^{\mathcal{W}}$ direction. This is shown in Fig. 9(c). High covariance regions in Fig. 9(a) correspond to localization failures.



(a) Experiment in university building (b) Experiment in university building

Fig. 10 These figures show results for tests carried in a corridor of length 42 meters in a building of University of Pennsylvania. Estimation outputs are given for $x^{\mathcal{W}}$-$y^{\mathcal{W}}$-$z^{\mathcal{W}}$ positions together with covariances as shaded regions. Videos of this experiment can be found at: http://mrsl.grasp.upenn.edu/tolga/FSR2013.mp4

5 Conclusion and Future Work

This work presented results of localization and semi-autonomous control of a quadrotor flying in a dam penstock. We used a Rao-Blackwellized particle filter for localization consisting of a standard particle filter for localization along the tunnel axis and a UKF to represent estimates the other five directions. This way we can represent uncertainty along the tunnel axis, which is quite significant compared to the other directions, using an non parametric distribution. Because of this anisotropy, our experiments required the human operator to specify input (acceleration) along the tunnel axis while the low-level control software provides for regulation and trajectory tracking in the other five directions.

This work is significant because it can replace the tedious and expensive process of manual inspection involving building scaffolds with human inspectors with semi-autonomous quadrotors with cameras. We believe that with some training a modestly skilled operator can fly a quadrotor through a tunnel while inspecting images from onboard cameras for defects along the tunnel walls. While our experiments were performed in penstocks that are used in dams and hydroelectric power plants, the same approach can be used for other tunnels such as those encountered in transportation networks.

Our current work is directed toward addressing more complex (but known) geometries encountered in dams near turbines and to improve the estimation of localization errors along the tunnel axis using onboard illumination sources and visual odometry algorithms.

Acknowledgements: We gratefully acknowledge the support of ONR Grants N00014-07-1-0829 ARL Grant W911NF-08-2-0004, NSF grants 0742304, 1138110, and 1113830, and collaborations with Jennifer Wozencraft and Thomas Hood of the US Army Corps of Engineers.

References

1. Ascending Technologies GmbH.
2. Hokuyo Automatic Co. Ltd.
3. M. Agrawal and K. Konolige. Real-time localization in outdoor environments using stereo vision and inexpensive GPS. In *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, volume 3, pages 1063–1068, 2006.
4. W. Burgard, A. Derr, D. Fox, and A.B. Cremers. Integrating global position estimation and position tracking for mobile robots: the dynamic Markov localization approach. In *Intelligent Robots and Systems, 1998. Proceedings., 1998 IEEE/RSJ International Conference on*, volume 2, pages 730–735 vol.2, 1998.
5. Wolfram Burgard, Armin B. Cremers, Dieter Fox, Dirk Hhnel, Gerhard Lakemeyer, Dirk Schulz, Walter Steiner, and Sebastian Thrun. Experiences with an interactive museum tour-guide robot. *Artificial Intelligence*, 114(12):3 – 55, 1999.
6. Andrea Censi. An accurate closed-form estimate of ICP's covariance. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 3167–3172, Rome, Italy, April 2007.
7. F. Dellaert, D. Fox, W. Burgard, and S. Thrun. Monte Carlo localization for mobile robots. In *Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference on*, volume 2, pages 1322–1328 vol.2, 1999.
8. Arnaud Doucet, Nando de Freitas, Kevin P. Murphy, and Stuart J. Russell. Rao-Blackwellised particle filtering for dynamic bayesian networks. *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence*, pages 176–183, 2000.
9. H. Durrant-Whyte and Tim Bailey. Simultaneous localization and mapping: part i. *Robotics Automation Magazine, IEEE*, 13(2):99–110, 2006.
10. Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.
11. A. Fitzgibbon, M. Pilu, and R.B. Fisher. Direct least square fitting of ellipses. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 21(5):476–480, 1999.
12. G. Grisetti, C. Stachniss, and W. Burgard. Improved techniques for grid mapping with Rao-Blackwellized particle filters. *Robotics, IEEE Transactions on*, 23(1):34–46, 2007.
13. S. Grzonka, G. Grisetti, and W. Burgard. Towards a navigation system for autonomous indoor flying. In *Robotics and Automation, 2009. ICRA '09. IEEE International Conference on*, pages 2878–2883, 2009.
14. Peter Hansen, Peter Corke, and Wageeh Boles. Wide-angle visual feature matching for outdoor localization. *Int. J. Rob. Res.*, 29(2-3):267–297, February 2010.
15. M. Hentschel, O. Wulf, and B. Wagner. A GPS and laser-based localization for urban and non-urban outdoor environments. In *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pages 149–154, 2008.
16. T. Liu, M. Carlberg, G. Chen, J. Chen, J. Kua, and A. Zakhor. Indoor localization and visualization using a human-operated backpack system. In *Indoor Positioning and Indoor Navigation (IPIN), 2010 International Conference on*, pages 1 –10, sept. 2010.
17. R. Mahony, V. Kumar, and P. Corke. Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor. *Robotics Automation Magazine, IEEE*, PP(99):1, 2012.
18. N. Michael, D. Mellinger, Q. Lindsey, and V. Kumar. The GRASP multiple micro-uav testbed. *Robotics Automation Magazine, IEEE*, 17(3):56 –65, 2010.
19. S. Se, D.G. Lowe, and J.J. Little. Vision-based global localization and mapping for mobile robots. *Robotics, IEEE Transactions on*, 21(3):364–375, 2005.
20. Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic Robotics*. MIT Press, 2005.
21. Sebastian Thrun, Dieter Fox, Wolfram Burgard, and Frank Dellaert. Robust Monte Carlo localization for mobile robots. *Artificial Intelligence*, 128(12):99 – 141, 2001.