Mapping GPS-denied aquatic environments

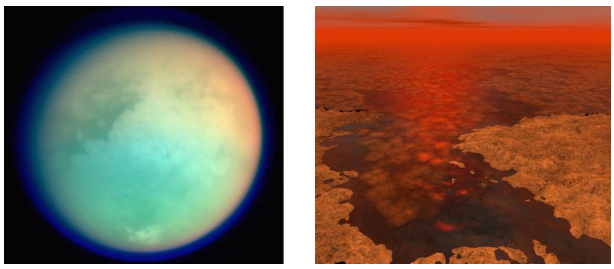
Robert Codd-Downey and Michael Jenkin
Department of Electrical Engineering and Computer Science
York University
Toronto, Ontario, M3J 1P3, Canada
{*robert, jenkin*}*@cse.yorku.ca*

***Abstract*— Building a representation of space and estimating a robots location within that space is a fundamental task in robotics known as simultaneous localization and mapping (SLAM). This work examines the problem of solving SLAM in aquatic environments using an unmanned surface vessel under conditions that restrict global knowledge of the robots pose. These conditions refer specifically to the absence of a global positioning system to estimate position, a poor vehicle motion model, and the lack of a strong stable magnetic field to estimate absolute heading. These conditions can be found in terrestrial environments where the line of sight to overhead satellites is occluded by surrounding structures and local magnetic inference affects reliable compass measurements. Similar conditions are anticipated in extra-terrestrial environments such as on Titan where the lack of a global satellite network inhibits the use of traditional positioning sensors and the lack of a stable magnetic core limits the applicability of a compass. This work develops a solution to the SLAM problem that utilizes shore features coupled with information about the depth of the water column. Theoretical results are validated experimentally using an autonomous surface vehicle utilizing omnidirectional video and a depth sounder. Solutions are compared to ground truth obtained using GPS and to solutions found when the restriction of a poor magnetic field is lifted.**

***Index Terms*— SLAM, autonomous surface vehicles, visual and depth landmarks**

I. Introduction

Most robots operate on the ground plane, although a range of interesting environments exist beyond this domain. To take but just one non-ground-plane application, consider the problem of navigating and representing the surface of bodies of water. Localization and mapping of bodies of liquids (primarily water terrestrially) finds a wide range of applications. Terrestrially, reservoirs, lakes, ponds, rivers and the like provide a range of interesting environments for robot operations. Furthermore, many of these environments are GPS denied (e.g., underground reservoirs, surface water bodies in mountainous regions, etc.) preventing the use of a global localization solution. Off-earth lakes, (not necessarily of water) on the surface of non-terrestrial planets and natural satellites are of particular importance to the future of space exploration. Titan, one of Saturn's moons, is the only known celestial body within our solar system with hydrocarbon lakes on its surface [**?**]. Other celestial bodies such as Europa and Enceladus may have interior bodies of water [1][2], however, this makes them less accessible to exploration by an autonomous robot and very recently there has been evidence of water on Mars [3]. Titan, shown in Figure 4 is of particular interest for space exploration because of its geological environment and its potential to support pre-biotic chemistry [4]. Titan's size and thus curvature presents additional problems for SLAM, being that even moderately sized lakes on Titan can have large sections in which lake shores are either not visible, or not discernible due to limitations on the resolution of standard cameras. Titan does not support a constant magnetosphere [5], limiting the usefulness of a compass in navigation and mapping. As a result, SLAM on but near the shore of the lakes of Titan must rely on egocentric sensors, including visual, LiDAR, RADAR and SONAR sensors.



(a) *(b)*

Fig. 1: **Titan**
(a) Multi spectral image of Titan. (b) Artist rendering of a Titan lake.

Terrestrial regions with poor compass performance also exist, with large metal bodies and electromagnetic fields generated by motors and the like typically lead to substantive failures in terms of compass performance. Although the work in this thesis is intended to be applicable to the SLAM task on bodies of liquid generally, one goal of this work is to investigate how the limits of Titan's sensing environment constrains SLAM algorithms operating in the environment found on Titan.

The algorithm described here relies on a combination of

*This work was supported by the Natural Sciences and Engineering Research Council (NSERC) through the NSERC Canadian Field Robotics Network (NCFRN).

Fig. 2: The Kingfisher robot used in this work. The base robot has been augmented with enhanced onboard computation and depth and omndirectional video sensors.

visual and SONAR measurements to perform SLAM on the surface of an aquatic environment. Visual measurements are captured using an omnidirectional camera which produces bearing-only measurements. SONAR measurements are obtained from a depth and temperature transducer.

II. Previous work

SLAM is a well studied problem in autonomous systems and for sufficiently well conditioned environments, sensors and plant models, the problem is considered solved. As these restrictions are loosened, existing solutions typically require tuning or other adjustments in order to enable SLAM to be solved. For a review of existing SLAM algorithms see [**?**].

One well known SLAM solutions is known as FastSLAM 2.0[12]. FastSLAM 2.0 seeks to build a conditional probability function that represents the joint probability of the environmental representation (the map) and the trajectory of the robot given the measurements obtained by the robot and its commanded motion.

A key aspect in any SLAM algorithm is the nature of the landmarks and features used. Many sensors used in autonomous systems obtain bearing (or relative bearing) to sensor features in the environment. As the goal is to build a Cartesian map of the world it becomes necessary to obtain full position information from the bearing data obtained from the sensors. The problem of initializing new landmarks from 'bearing-only' measurements is a common problem with visual SLAM algorithms and is widely addressed in the literature and many solutions have arisen. Theses solutions can be categorized broadly into two groups: delayed initialization (see [6] as an example) and un-delayed initialization (see [7], [8], [9], [10] as examples). Delayed approaches keep track of bearing measurements of a single landmark and these bearing measurements are aggregated over small motions of the robot until a realistic estimate of the full state of the landmark can be obtained. Determining the criteria for landmark initialization is complex and many solutions can be described as ad hoc.

Un-delayed initialization approaches take a somewhat different approach. Rather than waiting until enough bearing measurements have been taken to obtain a good estimate of the full state of the landmark, initialization proceeds immediately using only the bearing data. In the un-delayed approach a new landmark is initialized immediately at some distance ρ_c from the robot. The uncertainty of the distance of the landmark from the robot is set so that the uncertainty covers the entire distance range from ρ_{max} to ρ_{min}, while the variance in bearing and elevation is set from the known sensor

(a) Skyline data

(b) Sample sonar data

Fig. 3: Sample data from the sensors onboard the robot. (a) shows a sample skyline data san including SIFT features. (b) shows a trace from the SONAR depth sensor.

error properties. A covariance representation in Cartesian space is then constructed from these values. Within the un-delayed category there is a split between how the covariance is represented, simple methods (e.g., [7] and [8]) represent the entire uncertainty using a single covariance matrix, other approaches (e.g., [9] and [10]) represent the landmark uncertainty using a Gaussian sum filter where landmarks are added to the map immediately as meta landmarks. During the update phase each distribution within the Gaussian sum filter is updated using the standard EKF update process, when a single distribution's uncertainty decreases more than the others that distribution is chosen to represent the landmark which ends the landmarks meta phase.

III. SLAM Algorithm

The SLAM algorithm presented in the paper is a modified version of the multiple landmark extension introduced in[6] which in turn is an extension of the FastSLAM 2.0 algorithm[12]. The notable modifications introduced here include the landmark culling process, landmark/measurement descriptors and a probability weighting function. These additional features help to overcome the limitation of our primary sensor, an omnidirectional camera with bearing-only SIFT[13] features which are descriptively volatile and visible from a certain viewing angles. The algorithm has been validated using a real autonomous surface vessel "Eddy" which is a modifed Clearpath Kingfisher M100 vehicle (see Figure 2).

1) Shoreline features and depth landmarks: SLAM for a surface vessel must rely on stable landmarks, and such landmarks are typically associated with shore features. Here we utilize shore/skyline features obtained with an omnidirectional sensor as shoreline features. Here we utilize SIFT-features[13] obtained from a skyward facing omnidirectional sensor to capture stable shore features.

SIFT features are typically matched using .... Given the relative sparse nature of skyline features here a different matching process is used that relies on .... Sample data from the omnidirectional depth sensor is shown in Figure 3(a).

The SONAR sensor uses a standard fish-finding depth sensor to identify the local depth of the water column. The sensor used here is described in [**?**]. Sample data from the sensor is shown in Figure 3(b).

2) Culling non-useful landmarks: Given the large number of possible landmarks and the cost associated with processing

them, it is necessary to prune landmarks that are not used in the map. In order to do this the algorithm keeps track of the number of times each landmark has been sighted and the number of steps between sightings. This is done so that landmarks that are unlikely to be be seen again or have a large uncertainty can be culled from the map. This helps to reduce the number of landmarks and improve the computational efficiency of the data association process.

- $i^k_{\mathbf{m}_j,t}$: number of times the landmark $\mathbf{m}_j$ of the k-th particle has been sighted within the environment at time t.
- $TTL^k_{\mathbf{m}_j,t}$ (time to live) : number of consecutive unsighted steps at time t before the landmark $\mathbf{m}_j$ of the k-th particle becomes inconsequential and should be removed from the map of the k-th particle.
- $vol(\Sigma_{\mathbf{m}_j,t_{\mathbf{m}_j}})$: volume of the landmark $\mathbf{m}_j$'s uncertainty at time $t_{\mathbf{m}_j}$ (the time when the landmark was created).

We wish to prune landmarks that have not be 'seen' by any particle recently. To do this, we define three constants that define how each landmark's time to live $TTL^k_{\mathbf{m}_j,t}$ counter is modified in specific events. These constants are specific to the measurement type of $\mathbf{z}_{t,l}$

- $\alpha_{\mathbf{z}_{t,l}}$: multiplier to apply to an existing landmark's $TTL^k_{\mathbf{m}_i,t}$ (time to live) when it has been sighted again within the environment.
- $\beta_{\mathbf{z}_{t,l}}$: number of sightings before a landmark of the measurement type $\mathbf{z}_{t,l}$ is eligible to become a permanent fixture of the current particle's map.
- $\gamma_{\mathbf{z}_{t,l}}$: numerical value used to initialize the time to live counter of a landmark created from the measurement $\mathbf{z}_{t,l}$.

3) Distinguishing landmarks and measurements: This algorithm also incorporates landmark/measurement descriptions that are used to further discriminate between measurements and landmarks. The inclusion of these descriptors helps to rule out measurements that might otherwise be associated with a landmark given different criteria, mainly proximity.

- $\mathbf{D}_{\mathbf{z}_{l,t}}$: quantitative description of the l-th measurement observed at time t.
- $\mathbf{D}_{\mathbf{m}^k_i}$: quantitative description of the i-th landmark in the k-th particle.

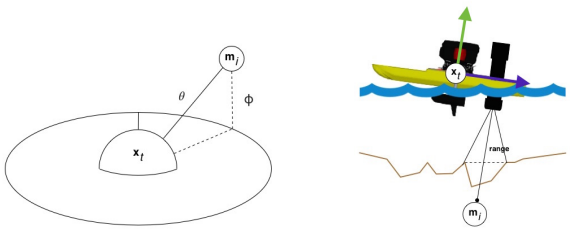
4) Probability weighting function: The non-delayed initialization of bearing-only landmarks used in our approach assigns a large amount of uncertainty to new landmarks which makes data association a difficult task. The probability weighting function w allows us to modify the contribution of the proximal likelihood $\phi_{j,l}$ based on the certainty of the landmark $\mathbf{m}^k_j$ in order to increase the reliance on the landmarks description. This weighting function is also used to augment the weights of each landmark within the map update process. This prevents landmarks with high uncertainty from having too much influence on the overall weighting of the current particle.

- $w(\phi_{j,l}) \in [0,1]$: a probability value raised to the power of w can be used to modify the value's contribution to a joint probability distribution. As the value of the weighting function reaches zero the more the original value is modified and thus its contribution is decreased.

The algorithm detailed in Alg III.1 receives a set of $\mathbf{Z}_t$ measurements at each time t, along with a control $((u)_t)$ these values are used to update and augment the previous set of particles Y_{t-1}. Each particle contains an oracle representing a possible pose of the robot $\hat{\mathbf{x}}^k_t$ and an estimation of the map as measured from the particles path. Each landmark $\mathbf{m}^k_j$ is represented by a mean $\mu^{\mathbf{k}}_{\mathbf{j,t-1}}$ and covariance $\Sigma^k_{j,t-1}$ each landmark also has an associated counter $i^k_{j,t-1}$ which represents the number of time the landmark has been sighted within its environment. The main steps of the algorithm can be compartmentalized into the following steps.

A. Measurement Model

Incorporating measurements from different sensors into a FastSLAM-based requires the definition of measurement models for each of the included sensors. The measurement model for each sensor includes three aspects, the primary aspect is a measurement function which calculates the expected measurement for a landmark from a given robot state. The other two aspects of a measurement model include the jacobian of the measurement function with respect to the robot's state and the jacobian of the measurement function with respect to the landmark's position.



(a) *(b)*

Fig. 4: **Measurement Models**
(a) Illustration of the bearing-elevation measurement model of an omnidirectional camera. (b) Illustration of the range measurement model of the single point SONAR sensor.

1) Omnidirectional camera model: The omnidirectional camera maps points from a 3-dimensional world frame into a 2-dimensional image frame (see Figure **??**), this process strips away depth cues from the measurement. Without such information the function transforming points from the world

Algorithm III.1 FastSLAM 2.0

procedure FASTSLAM
  for each $\mathbf{y}^k \in \mathbf{Y}_{t-1}$ **do**
    $\hat{\mathbf{x}}_t^k = g(\mathbf{x}_{t-1}^k, \mathbf{u}_t)$
    $\Phi = measurementsLikelihood(\hat{\mathbf{x}}_t^k, M^k, \mathbf{Z}_t)$
    $\Psi = dataAssociation(\Phi)$
    $\hat{\mathbf{x}}_t^k = robotPoseUpdate(\hat{\mathbf{x}}_t^k, M^k, \mathbf{Z}_t, \Psi)$
    $w^k, M^k = landmarksUpdate(\hat{\mathbf{x}}_t^k, M^k, \mathbf{Z}_t, \Psi)$
  end for
  $Y_t = sampling(\hat{Y}_t)$
end procedure

frame to the image frame cannot be used to obtain an exact location of the landmark m_i. The basic sensor model for a measurement $\mathbf{m}_i$ and landmark $\mathbf{x}_t$ is given in equations 1. This takes a landmark m and a known robot state x_t and computes the appropriate elevation and azimuth from these values. Let $r = \sqrt{(\mathbf{m}_{i_x} - \mathbf{x}_{t_x})^2 + (\mathbf{m}_{i_y} - \mathbf{x}_{t_y})^2}$ then

$$g(\mathbf{x}_t, \mathbf{m}_i) = \begin{bmatrix} \theta \\ \phi \end{bmatrix} = \begin{bmatrix} tan^{-1}(\frac{\mathbf{m}_{i_y} - \mathbf{x}_{t_y}}{\mathbf{m}_{i_x} - \mathbf{x}_{t_x}}) - \mathbf{x}_{t_\theta} \\ tan^{-1}(\frac{\mathbf{m}_{i_z}}{r}) \end{bmatrix} \quad (1)$$

B. SONAR sensor model

Whereas distinct visual features can be identified using some appropriate feature descriptor such as SIFT given the continuous nature of the visual sensor the same is not true for the depth sensor. Landmarks discovered by this sensor have no inherit descriptor and can only be identified by their location within the environment. Here measurement density is quite low and an alternative approach is desirable.

Before considering this, however, it is important to observe that the measurement (Figure **??**) returned from the depth sensor is impacted by its placement on the robot (its relative position and orientation) as well as the pitch, roll and yaw of the vehicle itself. Equation 2 below relates the returned depth measurement to the robot's state, assuming that the relative (x, y, z) offset (a, b, c) has already been corrected. Note that this incorporates vessel roll, pitch and yaw.

$$g(\mathbf{x}_t, \mathbf{m}_i) = \sqrt{\begin{array}{l} (\mathbf{m}_{i_x} - \mathbf{x}_x - acos(\mathbf{x}_\theta) + asin(\mathbf{x}_\theta))^2 \quad + \\ \quad (\mathbf{m}_{i_y} - \mathbf{x}_y - bcos(\mathbf{x}_\theta) - bsin(\mathbf{x}_\theta))^2 \quad + \\ \quad\quad (\mathbf{m}_z - c)^2 \end{array}} \quad (2)$$

C. Measurement likelihood

The list of data associations (Ψ) between measurements and landmarks is determined by the probability that a landmarks $\mathbf{m}_j$ corresponds to the measurement $\mathbf{z}_{l,t}$. The algorithm shown in Alg III.2 describes how the likelihood of a landmark/measurement association ($\Phi_{j,l}$) is calculated.

Algorithm III.2 Measurement Likelihood

procedure MEASUREMENT LIKELIHOOD($\hat{\mathbf{x}}_t^k, M^k, \mathbf{Z}_t$)
  for each $\mathbf{m}_j^k \in \mathbf{M}^k$ **do**
    $\hat{z}_j = h(\hat{\mathbf{x}}_t, \mathbf{m}_j^k)$
    $H_{\mathbf{x}_t} = \nabla_{\mathbf{x}_t} h(\hat{\mathbf{x}}_t, \mathbf{m}_j^k)$
    $H_{\mathbf{m}_j} = \nabla_{\mathbf{m}_j} h(\hat{\mathbf{x}}_t, \mathbf{m}_j^k)$
    for each $\mathbf{z}_l \in \mathbf{Z}_t$ **do**
      $Q_{j,l} = P_{\mathbf{z}_l} + H_{\mathbf{m}_j^k} \Sigma_{\mathbf{m}_j^k} H_{\mathbf{m}_j^k}^T$
      $\Sigma_{\mathbf{x}_t^k,j,l} = [H_{\mathbf{x}_t^k}^T Q_{j,l}^{-1} H_{\mathbf{x}_t^k} + R_t^{-1}]^{-1}$
      $\mu_{\mathbf{x}_t^k,j,l} = \Sigma_{\mathbf{x}_t^k,j,l} H_{\mathbf{x}_t^k}^T Q_{j,l}^{-1} (\mathbf{z}_j - \hat{z}_j) + \hat{\mathbf{x}}_t^k$
      $\hat{z}_{j,l} = h(\mu_{\mathbf{x}_t^k,j,l}, \mathbf{m}_j^k)$
      $\phi_{j,l} = \frac{1}{\sqrt{(2\pi)^d |Q_{j,l}|}} exp(-\frac{1}{2}(\mathbf{z}_l - \hat{z}_{j,l})^T Q_{j,l}^{-1} (\mathbf{z}_l - \hat{z}_{j,l}))$
      $\Phi_{j,l} = \phi_{j,l}^{w(\phi_{j,l})} \cdot \lambda_{j,l}$
    end for
  end for
end procedure

Each likelihood is dependant on the measurement ($\mathbf{z}_{l,t}$), the predicted measurement ($\hat{\mathbf{z}}_{l,t}$) and the measurement innovation covariance matrix ($Q_{l,t}$). The measurement innovation covariance integrates both the measurement covariance (Q_t), the previous landmark covariance ($\Sigma_{j,t-1}^k$) and both jacobian's of the measurement models. The predicted covariance of the new landmark is dependant on the control covariance (R_t) and the measurement innovation covariance ($Q_{l,t}$). An estimate of the robot's pose $\mu_{\mathbf{x}_t^k,j,l}$ integrates a correction from the assignment of the measurement $\mathbf{z}_{l,t}$ to the landmark $\mathbf{m}_j$. A predicted measurement of the the landmark $\mathbf{m}_j$ can be estimated using the new estimated of the robot's pose $\mu_{\mathbf{x}_t^k,j,l}$ and the location of the landmark's previous location. The likelihood($\phi_{j,l}$) that a landmark $\mathbf{m}_j$ corresponds to a measurement $\mathbf{z}_{l,t}$ can be modelled by the probability density function of a normal distribution with mean $\hat{\mathbf{z}}_{l,t}$ and covariance $Q_{l,t}$. The likelihood $\phi_{j,l}$ is further augmented by the the weighting function $w(\phi_{j,l})$ and the likelihood($\lambda_{j,l}$) that the description $\mathbf{D}_{\mathbf{z}_{l,t}}$ corresponds to the description $\mathbf{D}_{\mathbf{m}_j}$.

D. Robot pose update

A new estimate for the robot's pose is sampled from a proposal distribution that incorporates the input from both the motion of the robot($\mathbf{u}_t$) and the set of landmark/measurement associations. The proposal distribution is modelled as a Gaussian distribution. The mean and covariance of the Gaussian distribution starts from an initial estimate of the current robot pose $\hat{\mathbf{x}}_t^k$ and the control covariance R_t. Corrections to these initial estimates are iteratively incorporated through corrections from associations from landmarks and measurements. It is important to note that some associations are excluded from this process if the associated landmark's weighting function

Algorithm III.3 Pose update algorithm

procedure POSEUPDATE($\hat{\mathbf{x}}_t^k, M^k, \mathbf{Z}_t, \Psi$)
  if $\Sigma_{j=1}^{|\mathbf{M}^k|} == 0$ **then**
    $x_t^k \sim p(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{u}_t)$
  else
    $\Sigma_{\mathbf{x}_t,0} = R_t$
    $\mu_{\mathbf{x}_t,0} = \hat{\mathbf{x}}_t$
    for each $\mathbf{m}_j \in \mathbf{M}^k$ **do**
      if $w(\phi_{j,l}) < 0.5$ **then**
        $continue$
      end if
      if $\psi_j > 0$ **then**
        $\Sigma_{\mathbf{x}_t,j} = [H_{\mathbf{m}_j}^T Q_{j,\psi_j}^{-1} H_{\mathbf{m}_j} \Sigma_{\mathbf{x}_t,j-1}^{-1}]^{-1}$
        $\mu_{\mathbf{x}_t,j} = \mu_{\mathbf{x}_t,j-1} + \Sigma_{\mathbf{x}_t,j} H_{\mathbf{x}_t,j}^T Q_{j,\psi_j}^{-1}(z_{t,\psi_j} - \hat{z}_j)$
      end if
    end for
    $\Sigma_{\mathbf{x}_t} = \Sigma_{\mathbf{x}_t,j}$
    $\mu_{\mathbf{x}_t} = \mu_{\mathbf{x}_t,j}$
    $x_t^k \sim N(\mu_{\mathbf{x}_t}, \Sigma_{\mathbf{x}_t})$
  end if
end procedure

is greater that 0.5. If a landmark association qualifies for inclusion the proposal mean and covariance are updated to incorporate the uncertainty of the measurement with respect to the robot's state.

Finally a new pose is randomly sampled from the proposal distribution. This process introduces the variability in the FastSLAM process.

E. Map update

The landmark update process iterates through each known landmark in the current particle and evaluates them accordingly. Each landmark that has a corresponding measurement association is updated using the standard EKF update process and updates the landmarks sighting counter $i_{\mathbf{m}_j,t}^k$ and time to live counter $TTL_{\mathbf{m}_j,t}^k$ by a factor of $\alpha_{\mathbf{z}_{t,\psi_j}}$. Any landmark that has been sighted at least $\beta_{\mathbf{z}_{t,\psi_j}}$ qualifies to included in the particle's weight. Landmarks without a corresponding measurement have their time to live counter $TTL_{\mathbf{m}_j,t}^k$ decreased by 1. If a landmarks time to live counter decreases to zero it can be removed if it's sightings counter $i_{\mathbf{m}_j,t}^k$ is less than $\beta_{\mathbf{z}_{t,\psi_j}}$ and its uncertainty has decreased by at least 60%.

Any measurement that has not been associated with a known landmark is used to created a new landmark that is added to the map. New landmarks are initialized with a sighting counter $i_{\mathbf{m}_j,t}^k = 1$ and time to live counter $TTL_{\mathbf{m}_i,t}^k = \gamma_{\mathbf{z}_{t,\psi_i}}$, the mean and covariance of the landmark is based on the properties of the measurement that observed it.

Algorithm III.4 Map update algorithm

procedure MAPUPDATE($\hat{\mathbf{x}}_t^k, M^k, \mathbf{Z}_t, \Psi$)
  $w^k = 1$
  for each $\mathbf{m}_j \in \mathbf{M}^k$ **do**
    if $\psi_j > 0$ **then**
      $\bar{z}_j = h(x_t^k, \mathbf{m}_j)$
      $H_{\mathbf{m}_j} = \nabla_{\mathbf{m}_j} h(x_t^k, \mathbf{m}_j)$
      $Q_j = P_{\psi_j,t} + H_{\mathbf{m}_j} \Sigma_{\mathbf{m}_j,t-1} H_{\mathbf{m}_j}^T$
      $K = \Sigma_{\mathbf{m}_j,t-1} H_{\mathbf{m}_j}^T Q_j^{-1}$
      $\mu_{\mathbf{m}_j,t} = \mu_{\mathbf{m}_j,t-1} + K(z_{t,\psi_j} - \bar{z}_j)$
      $\Sigma_{\mathbf{m}_j,t} = (I - KH_{\mathbf{m}_j})\Sigma_{\mathbf{m}_j,t-1}$
      if $i_{j,t}^k \geq \beta_{\mathbf{z}_{t,\psi_j}}$ **then**
        $H_{\mathbf{x}_t} = \nabla_{\mathbf{x}_t} h(\hat{\mathbf{x}}_t, \mathbf{m}_j)$
        $L = H_{\mathbf{x}_t} R_t H_{\mathbf{x}_t}^T + Q_j$
        $\hat{w} = \frac{1}{\sqrt{(2\pi)^r |L|}} exp(-\frac{1}{2}(\mathbf{z}_l - \hat{z}_{j,l})^T L^{-1}(\mathbf{z}_l - \hat{z}_{j,l}))$
      else
        $\hat{w} = 1$
      end if
      $i_{\mathbf{m}_j,t}^k = i_{\mathbf{m}_j,t-1}^k + 1$
      $TTL_{\mathbf{m}_j,t}^k = TTL_{\mathbf{m}_j,t-1}^k * \alpha_{\mathbf{z}_{t,\psi_j}}$
    else
      $TTL_{\mathbf{m}_j,t}^k = TTL_{\mathbf{m}_j,t-1}^k - 1$
      if $TTL_{\mathbf{m}_j,t}^k = 0$ and $i_{\mathbf{m}_j,t}^k < \beta_{\mathbf{z}_{t,\psi_j}}$ and $vol(\Sigma_{\mathbf{m}_j,t}) > 0.4 vol(\Sigma_{\mathbf{m}_j,t_{\mathbf{m}_j}})$ **then**
        delete $\mathbf{m}_j$
      end if
      $\hat{w} = 1$
    end if
    $w^k = w^k \cdot \hat{w}^{w(\phi_{j,l})}$
  end for
  for $i = |\mathbf{M}^k|,\ i < |\psi|,\ i{+}{+}$ **do**
    if $\psi_i > 0$ **then**
      $\mathbf{m}_i = createLandmark(x_t^k, \mathbf{z}_{t,\psi_i})$
      $i_{\mathbf{m}_i,t}^k = 1$
      $TTL_{\mathbf{m}_i,t}^k = \gamma_{\mathbf{z}_{t,\psi_i}}$
    end if
  end for
end procedure

IV. RESULTS

V. CONCLUSIONS

This paper has shown how the simultaneous measurement extension of the original FastSLAM 2.0 algorithm[12] described in [6] can be further extended to incorporate measurements from both a SONAR sensor and omnidirectional camera. This extension includes a number of features that allows the algorithm to be tuned in a variety of ways. The

most notable difference between this implementation and its predecessor is the inclusion of a time to live counter associated with each landmark which can be used as the a means for removing inconsequential landmarks from the map. This accounts for assumption adopted in the previous implementation that landmarks are visible throughout the environment. In this research landmarks are considered weak until they have been observed at at least $\beta_{\mathbf{z}_{t,\psi_j}}$ times and have increased in certainty by at least 60%. This puts less pressure on a landmark's continual observance and instead puts more weight on the amount of certainty associated with the landmark. This method helps to increase the amount of weak landmarks that are removed from the map overtime. Another major change introduces a weighting function augmenting the proximal likelihood of an association between a landmark and measurement. This weighting function helps to lend more influence to the visual likelihood of a measurement association when a landmark has a high uncertainty. This improves the effectiveness of the data association process.

VI. Future Work

This research has addressed how the inclusion of the depth the water column can be used to improve localization and mapping using an aquatic surface vehicle. The most obvious extension to this research is to upgrade the single-beam SONAR sensor in favour of either a multi-beam or side-scanning SONAR sensor. These sensors provide measurements of the lakebed at much higher resolution and wider field of view. Integrating these sensors into a appropriate SLAM algorithm would allow for higher fidelity mapping of the lakebed and the ability perform scan matching between scans and/or the environment. There are also additional sensors that can be added to the robot to provide increased situational awareness. LiDAR sensors can be effective for detecting environmental landmarks at moderate distances. RADAR sensors are effective for moderate to long range detection of other vehicles, terrain and even weather patterns. The addition of these sensor types would help offset mapping performance under adverse conditions such as lighting conditions and fog, both of which prevent visual sensors obtaining accurate measurements. LiDAR can still operate effectively without ambient light and RADAR remains unaffected by all but the most severe weather.

There are also many other avenues that can be explored from an algorithmic standpoint. One such adjustment that could be made changes the assumption made about the sparsity of depth measurements. In the current formulation the environments is assumed to be sparsely populated by landmarks above and below the surface of the water. The visual landmarks within the scene can be modelled as sparse so long as they are differentiable by a mean other than there location. However SONAR measurements can be obtained from every location within the environment and may be more effective modelled as densely packed landmarks. This would require a reformulation of the SLAM algorithm introducing increased complexity with the benefit a better model of the lakebed.

Acknowledgment

The support of NSERC Canada and the NSERC Canadian Field Robotics Network (NCFRN) is gratefully acknowledged.

References

[1] NASA. (2013) Hubble sees evidence of water vapor at jupiter moon. Accessed 25-September-2015. [Online]. Available: http://www.jpl.nasa.gov/news/news.php?release=2013-363

[2] ——. (2014) Nasa space assets detect ocean inside saturn moon. Accessed 25-September-2015. [Online]. Available: http://www.jpl.nasa.gov/news/news.php?release=2013-363

[3] ——. (2015) Nasa confirms evidence that liquid water flows on todays mars. Accessed 25-September-2015. [Online]. Available: https://www.nasa.gov/press-release/nasa-confirms-evidence-that-liquid-water-flows-on-today-s-mars

[4] C. Neish, R. Lorenz, D. O Brien, C. R. Team *et al.*, "The potential for prebiotic chemistry in the possible cryovolcanic dome ganesa macula on titan," *International Journal of Astrobiology*, vol. 5, no. 1, p. 57, 2006.

[5] S. Ledvina, J. Luhmann, S. Brecht, and T. Cravens, "Titan's induced magnetosphere," *Advances in Space Research*, vol. 33, no. 11, pp. 2092–2102, 2004.

[6] C. Gamallo, M. Mucientes, and C. V Regueiro, "A fastSLAM-based algorithm for omnidirectional cameras," *Journal of Physical Agents*, vol. 7, no. 1, pp. 12–21, 2013.

[7] A. J. Davison, "Real-time simultaneous localisation and mapping with a single camera," in *Proc. 2003 IEEE International Conference on*

Computer Vision (ICCV), Oct 2003, pp. 1403–1410 vol.2.

[8] S. Thompson and A. Zelinsky, "Accurate local positioning using visual landmarks from a panoramic sensor," in *Proc. 2002 IEEE International Conference on Robotics and Automation (ICRA)*, vol. 3, pp. 2656–2661.

[9] N. M. Kwok, G. Dissanayake, and Q. P. Ha, "Bearing-only SLAM using a SPRT based gaussian sum filter," in *Proc. 2005 IEEE International Conference on Robotics and Automation (ICRA)*, April 2005, pp. 1109–1114.

[10] T. Lemaire, S. Lacroix, and J. Sola, "A practical 3d bearing-only SLAM algorithm," in *Proc. 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2449–2454.

[11] J. M. Sáez, A. Hogue, F. Escolano, and M. Jenkin, "Underwater 3d slam through entropy minimization," in *Proc. 2006 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3562–3567.

[12] M. Montemerlo, S. Thrun, D. Roller, and B. Wegbreit, "Fastslam 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges," in *Proc. 2003 International Joint Conference on Artificial Intelligence (IJCAI)*, ser. IJCAI'03. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2003, pp. 1151–1156. [Online]. Available: http://dl.acm.org/citation.cfm?id=1630659.1630824

[13] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.