Gamma-SLAM: Stereo Visual SLAM in Unstructured Environments Using Variance Grid Maps

Tim K. Marks, Andrew Howard, Max Bajracharya, Garrison W. Cottrell, and Larry Matthies

***Abstract*— We introduce a new method for stereo visual SLAM (simultaneous localization and mapping) that works in unstructured, outdoor environments. Observations use dense stereo vision to measure the variance of the heights in each cell of a 2D grid. Unlike other grid-based SLAM algorithms, which use occupancy grid maps, our algorithm uses a new mapping technique that maintains a posterior distribution over the height variance in each cell. To obtain a joint posterior over poses and maps, we use a Rao-Blackwellized particle filter: the pose distribution is estimated using a particle filter, and each particle has its own map that is obtained through exact filtering conditioned on the particle's pose. For the particle filter over pose, visual odometry (VO) provides good proposal distributions. In the analytical (exact) filter for the map, we update the sufficient statistics of a gamma distribution over the precision (inverse variance) of heights in each grid cell. We demonstrate performance on two outdoor courses, and verify the accuracy of the algorithm by comparing with ground truth data obtained using electronic surveying equipment.**

I. Introduction

The task of simultaneous localization and mapping (SLAM) is to estimate from a temporal sequence of observations both a map of the environment and the pose (position and orientation) of the observer (the robot) in this map. Like many SLAM systems (e.g., [1; 2; 3]), our system uses a Rao-Blackwellized particle filter [4]: Each of multiple particles has a single hypothesis about the robot's pose; based on its pose history, each particle uses an exact filter to obtain a map.

Most current approaches to SLAM use laser range scans for their observations. Many of these laser-based systems (such as [1]) are *landmark-based*, in that each particle's map consists of a posterior distribution over the locations of a number of salient landmarks. Others (such as [2]) are *grid-based*, meaning that each particle's map is a dense occupancy grid containing the posterior probability that each cell in the grid is occupied. In addition to SLAM algorithms that use laser range measurements, there are a growing number of examples of vision-based slam, a few of which use stereo vision [3; 5; 6]. All of the existing stereo vision SLAM algorithms are landmark-based. Futhermore, most current approaches to SLAM address either indoor environments or structured outdoor environments.

We take a new approach to stereo SLAM that can be used in unstructured outdoor environments. Since we will use our SLAM maps for autonomous robotic navigation, it is not enough for the maps to contain the locations of easily-identifiable landmarks. For planning purposes, we need a dense map for estimating the traversability of every location (every cell) in the map. Our system, which we call Gamma-SLAM, represents the world using grid-based maps. Existing grid-based SLAM systems [2] use laser range finders and binary occupancy grids (the maps contain the posterior probability that each cell is occupied). In contrast, Gamma-SLAM uses dense stereo vision, and the maps contain the posterior distribution over the variance of heights in each cell. As a result, our maps distinguish different types of objects, which is crucial for unstructured outdoor environments.

Our new mapping technique maintains a posterior distribution over the variance of the heights (z-coordinates) of points in each grid cell. For each grid cell, this posterior distribution is a gamma distribution, which is why we call our algorithm *Gamma-SLAM*. As explained in Section IV, each map cell simply contains two scalar values, which are the sufficient statistics of this gamma distribution.

II. The mobile robotic platform

The algorithm was designed for and tested on the LAGR hHerminator robot [7] (see Figure 3, top), a differential drive vehicle with 0.75 m wheel separation, 0.16 m wheel radius, and top speed 1.3 m/s. Onboard sensors include an inertial measurement unit (IMU) and two stereo camera pairs that together provide a 140° field of view. Dense stereo range data are generated from each of the two camera pairs, then fused (with the help of the IMU reading) into a cartesian map containing map cell statistics [8]. The incremental vehicle pose is estimated using a visual odometry (VO) algorithm based on [9], applying a corner detector on each frame, matching features using the stereo SAD (sum of absolute differences) scores over a local window, and then estimating the camera motion using the 3D position of the features as determined by stereo.

III. The generative model

Our Gamma-SLAM system can be conceived as performing inference on a generative model described by the graphical model in Figure 1. Given the sequence of observations made by the robot from time 1 to t, denoted $z_{1:t}$, and the sequence of control commands (determined from visual odometry), denoted $u_{1:t}$, our goal is to simultaneously infer a posterior distribution over both the map of the environment,

Tim K. Marks and Garrison W. Cottrell are at the Department of Computer Science and Engineering, University of California, San Diego, La Jolla, CA. {tkmarks, gary}@cs.ucsd.edu

Andrew Howard, Max Bajracharya, and Larry Matthies are at NASA Jet Propulsion Laboratory, Pasadena, CA. {abhoward, maxb, lhm}@robotics.jpl.nasa.gov

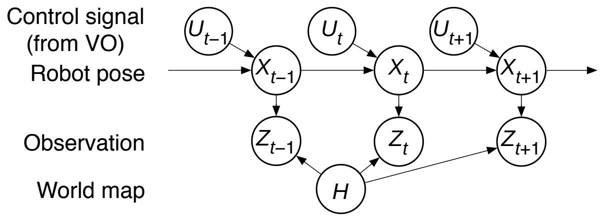

Fig. 1. The goal of Gamma-SLAM is to simultaneously infer H and $X_{1:t}$ given a sequence of observations $z_{1:t}$ and a sequence of control signals $u_{1:t}$ (obtained from visual odometry).

H, and the robot's path (pose sequence) through that environment, $X_{1:t}$. (We use uppercase to denote random variables and lowercase for the values taken by those variables.)

The world map is a grid of G square cells with side length 0.16 m. We assume that every point observed in cell g has its height drawn from a Gaussian distribution, with precision (inverse variance) h_g and unknown mean. The entire world map is $h = \{h_g\}_{g=1}^{G}$, the collection of all grid cells.

A. Motion model

The robot pose x_t consists of a 2D position, d_t, and orientation, θ_t. The control signal, u_t, consists of a robot-centered translation, Δd_t, and rotation, $\Delta\theta_t$. The motion model dictates that x_t is given by the previous pose incremented by the control signal, plus a small amount of Gaussian noise, Q_t:

$$\underbrace{\begin{bmatrix} D_t \\ \Theta_t \end{bmatrix}}^{X_t} = \overbrace{\begin{bmatrix} D_{t-1} \\ \Theta_{t-1} \end{bmatrix}}^{X_{t-1}} + \overbrace{\begin{bmatrix} \Delta D_t \\ \Delta\Theta_t \end{bmatrix}}^{U_t} + \overbrace{\begin{bmatrix} Q_{dt} \\ Q_{\theta t} \end{bmatrix}}^{Q_t}, \tag{1}$$

where $Q_{dt} \sim N(0, \sigma_d^2),\ Q_{\theta t} \sim N(0, \sigma_\theta^2)$.

B. Observation model

Each observation obtained from dense stereo vision contains a large number of points with locations (x, y, z). For each point observed, we bin the xy-locations into a square grid aligned with the world map. After the observed points are collected into cells, each cell in the grid contains a set of n heights (z-coordinates). For each cell, we assume that these heights are i.i.d. samples, $\{s_1, \ldots, s_n\}$, from a 1D Gaussian with unknown mean μ and precision h $\big($where precision $= \frac{1}{\text{variance}}\big)$. (For simplicity, we omit the subscript g from h_g.) Note that observations may have different values of n for different cells; for example, nearby cells tend to have larger n than faraway cells. We can write the likelihood of drawing these n samples as a function of the unknown population mean and precision, μ and h:

$$\begin{aligned} p(s_1, \ldots, s_n \mid \mu, h) &= \prod_{i=1}^{n} p(s_i \mid \mu, h) = \prod_{i=1}^{n} f_N(s_i; \mu, h) \\ &= (2\pi)^{-n/2} h^{n/2} e^{-\frac{1}{2}h\sum_{i=1}^{n}(s_i-\mu)^2}, \end{aligned} \tag{2}$$

where $f_N(s_i; \mu, h)$ denotes the 1D Gaussian pdf with mean μ and precision h.

We summarize the collection of data samples $\{s_i\}_{i=1}^{n}$ by its sample mean and variance, m and v:

$$m = \frac{1}{n}\sum_{i=1}^{n} s_i, \quad v = \frac{1}{k}\sum_{i=1}^{n}(s_i - m)^2, \text{ where } k = n - 1. \tag{3}$$

Now with some algebraic manipulation, we can rewrite the likelihood equation (2) in terms of m and v:

$$p(s_1, \ldots, s_n \mid \mu, h) = (2\pi)^{-n/2} e^{-\frac{1}{2}hn(m-\mu)^2} \cdot h^{n/2} e^{-\frac{1}{2}hkv}. \tag{4}$$

By integrating this likelihood over all values of $s_1, \ldots, s_n$ that have the same sufficient statistics m and v, we obtain the likelihood of observing data with mean and variance m, v when we sample n points from a Gaussian distribution whose population mean and precision are μ, h. This likelihood is given by the product of a normal distribution and a gamma distribution [10]:

$$\begin{aligned} p(m, v \mid \mu, h; n, k) &= f_N(m; \mu, hn) f_{\gamma 2}(v; h, k) \\ &= (2\pi)^{-1/2}(hn)^{1/2} e^{-\frac{1}{2}hn(m-\mu)^2} \\ &\quad \cdot \frac{1}{\Gamma\big(\frac{1}{2}k\big)} \big(\tfrac{1}{2}hk\big)^{k/2} v^{\frac{1}{2}k-1} e^{-\frac{1}{2}hkv}. \end{aligned} \tag{5}$$

As evident in the equation above, we use $f_{\gamma 2}$ as a convenient parameterization of the gamma function: $f_{\gamma 2}\big(v; h, k\big) \overset{\text{def}}{=} f_\gamma\big(v; \frac{1}{2}k, \frac{1}{2}hk\big) \overset{\text{def}}{=} \frac{1}{\Gamma\left(\frac{1}{2}k\right)}\big(\frac{1}{2}hk\big)^{k/2} v^{\frac{1}{2}k-1} e^{-\frac{1}{2}hkv}$.

Because it is difficult to accurately estimate the altitude of the robot, the mean height statistic m collected from an observation is not particularly informative. We therefore ignore the posterior distribution over μ, computing the marginal posterior distribution over h by integrating m out of (5):

$$\begin{aligned} p(v \mid h; k) &= \int_m f_N(m; \mu, hn) f_{\gamma 2}(v; h, k) dm \\ &= f_{\gamma 2}(v; h, k) = \frac{1}{\Gamma\big(\frac{1}{2}k\big)} \big(\tfrac{1}{2}hk\big)^{k/2} v^{\frac{1}{2}k-1} e^{-\frac{1}{2}hkv}. \end{aligned} \tag{6}$$

IV. Bayesian Update of a Map Cell

If we know the precise pose history (path) of the robot, $x_{1:t}$, as well as the sequence of observations $z_{1:t}$, then we can infer the posterior distribution over the world map. For each grid cell, $g = 1, \ldots, G$, the inferred map contains a distribution over the precision (inverse variance) of the Gaussian distribution of heights in that cell. The map can be thought of as combining all of the observations from time 1 to time t, aligned (rotated and translated) according to the pose at each time step, $x_{1:t}$. Making the simplifying assumption that the precisions of the grid cells are conditionally independent given the robot's path and observations, we express the map at time t as follows:

$$\underbrace{p(h \mid x_{1:t}, z_{1:t})}_{\text{map at time } t} = \prod_{g=1}^{G} p(h_g \mid x_{1:t}, z_{1:t}). \tag{7}$$

(For the rest of this paper, we omit the subscript g from h_g when discussing an individual cell.) A map is made up of cells, each of which contains the sufficient statistics

(v, k) of a gamma distribution. This can be interpreted as the distribution of our beliefs about the precision h of the normal distribution of heights from which the points in that cell were sampled. Each time we collect a new observation (a new set of data points for a map cell), we use the previous gamma distribution for the cell as our prior, then use the new data points for the cell to update our beliefs, obtaining a new gamma distribution as our posterior for that cell in the map.

Given the pose at time t, we know which points from the observation at time t correspond to each grid cell in the map. For any particular cell, at time t we observe some number $n = k+1$ of points that lie in that cell (according to the pose x_t), and we compute their data variance, v, using (3). This section explains how to use this observation (summarized by statistics v, k) to update the corresponding cell in the prior map (summarized by statistics v', k'), thus obtaining a posterior map (a posterior distribution over the precision of heights in this cell, summarized by statistics v'', k'').

A. The gamma prior

When estimating the precision h of the Gaussian distribution of heights in a cell, the conjugate prior is the natural conjugate of the likelihood distribution (6), which is a gamma distribution. We parameterize this gamma prior over h with parameters k' and v':

$$\begin{aligned} p(h \mid v', k') &= f_{\gamma 2}\big(h; v', k'\big) \\ &= \frac{1}{\Gamma\big(\frac{1}{2}k'\big)} \big(\tfrac{1}{2}k'v'\big)^{k'/2} h^{\frac{1}{2}k'-1} e^{-\frac{1}{2}hk'v'}. \end{aligned} \qquad (8)$$

This is the prior distribution before we incorporate our current data samples; the sufficient statistics k' and v' are obtained from the data collected in previous time steps. (Note that if the data mean were more informative, we could choose to calculate the posterior distribution over both the mean and precision of heights in each cell; in that case, the natural conjugate distribution would be the normal-gamma distribution [10].)

B. The gamma posterior

Suppose that for a given cell in the map, the prior distribution over h is defined by (8), with sufficient statistics (v', k'). Then we take an observation (we observe $n = k+1$ points from the same cell) that has sufficient statistics (v, k), defined by (3). By Bayes Rule, the posterior distribution for the cell in the map is proportional to the product of the prior distribution (8) and the likelihood (6):

$$\begin{aligned} p(h \mid v', k', v, k) &= \frac{p(h \mid v', k')\, p(v \mid h; k)}{\int_h p(h \mid v', k')\, p(v \mid h; k)\; dh} \\ &= \frac{f_{\gamma 2}(h; v', k')\, f_{\gamma 2}(v; h, k)}{\int_h f_{\gamma 2}(h; v', k')\, f_{\gamma 2}(v; h, k)\; dh}. \end{aligned} \qquad (9)$$

Multiplying (8) by (6) and eliminating terms that do not depend on h, we obtain:

$$p(h \mid v', k', v, k) \propto e^{-\frac{1}{2}h(k'v'+kv)}\, h^{\frac{1}{2}(k'+k)-1}. \qquad (10)$$

We can write the posterior more simply by defining new parameters k'' and v'': by letting

$$k'' = k' + k\ , \qquad v'' = \frac{k'v' + kv}{k' + k}, \qquad (11)$$

we obtain

$$p(h \mid v', k', v, k) \propto e^{-\frac{1}{2}hk''v''}\, h^{\frac{1}{2}k''-1}. \qquad (12)$$

This has the form of a gamma distribution in h. Since the posterior distribution over h is a properly normalized probability distribution, this posterior must be the following gamma distribution:

$$p(h \mid v', k', v, k) = f_{\gamma 2}(h; v'', k''). \qquad (13)$$

a) The Uninformative Prior: If no points have yet been observed in a cell, we use the uninformative prior: $k' = 0$, and $v' = 0$ (or indeed let v' equal any finite value). For this prior, the Bayesian update (11) simply produces $k'' = k$ and $v'' = v$. Thus if the incoming data are the first data to go into a cell, the sufficient statistics for the posterior distribution will simply be equal to the statistics of the data.

C. Likelihood of an observation

Suppose we have our map at time $t-1$, which is $p(h \mid x_{1:t-1}, z_{1:t-1})$. For any cell of the map, this is a gamma distribution that is summarized by the sufficient statistics v', k'. Suppose we observe k new points from this cell at time t. What is the likelihood that our observation will have a given data variance, v? (We will need the answer later to determine the relative weights of our particles.)

Suppose a cell of a prior map has statistics v', k', and we now observe $n = k+1$ new points from the same cell. The probability that our current observation will have data variance v is:

$$\begin{aligned} p(v \mid v', k'; k) &= \int_h p(h, v \mid v', k'; k)\, dh \\ &= \int_h p(v \mid h; k)\, p(h \mid v', k')\, dh. \end{aligned} \qquad (14)$$

By noting that this is the denominator of (9), we can use (9) and (13) to obtain the following:

$$\begin{aligned} p(v \mid v', k'; k) &= \frac{f_{\gamma 2}(v; h, k)\, f_{\gamma 2}(h; v', k')}{f_{\gamma 2}(h; v'', k'')} \\ &= \frac{\Gamma\big(\frac{1}{2}k''\big)}{\Gamma\big(\frac{1}{2}k\big)\Gamma\big(\frac{1}{2}k'\big)} \cdot \frac{(kv)^{k/2}(k'v')^{k'/2}}{(k''v'')^{k''/2}} \cdot v^{-1} \end{aligned} \qquad (15)$$

V. SLAM using Rao-Blackwellized particle filtering

The distribution that we estimate at each time step, which we refer to as the target distribution, is the posterior distribution over the path (pose history) of the robot and the corresponding map. Our inputs are the history of observations, $z_{1:t}$, and the visual odometry (VO) which we treat as the control signal, $u_{1:t}$. We factorize the target distribution as follows:

$$\underbrace{p(x_{1:t}, h \mid z_{1:t}, u_{1:t})}_{\text{target distribution}} = \underbrace{p(x_{1:t} \mid z_{1:t}, u_{1:t})}_{\text{filtering distribution}}\ \underbrace{p(h \mid x_{1:t}, z_{1:t})}_{\text{map}}. \qquad (16)$$

This factorization enables us to estimate the target distribution using Rao-Blackwellized particle filtering [4], a method that combines the approximate technique of particle filtering for some variables, with exact filtering (based on the values of the particle-filtered variables) for the remaining variables. We use a particle filter to approximate the first term on the right side of (16), the filtering distribution for the particle filter, using discrete samples $x_{1:t}^{[j]}$. For each of these samples (particles), we use exact filtering to obtain the second term in (16), the map, using the map updates described in Section IV.

By Bayes Rule and the probabilistic dependencies implied by the graphical model (Figure 1), we can write the filtering distribution for the particle filter as follows:

$$\underbrace{p(x_{1:t}|z_{1:t},u_{1:t})}_{\text{filtering distribution}} \propto \underbrace{p(z_t|x_{1:t},z_{1:t-1},u_{1:t})}_{\text{importance factor}} \underbrace{p(x_{1:t}|z_{1:t-1},u_{1:t})}_{\text{proposal distribution}}, \tag{17}$$

where the proportionality constant does not depend on the path $x_{1:t}$. We further factor the last term:

$$\underbrace{p(x_{1:t}|z_{1:t-1},u_{1:t})}_{\text{proposal distribution}} = \underbrace{p(x_t|x_{t-1},u_t)}_{\text{sampling distribution}} \underbrace{p(x_{1:t-1}|z_{1:t-1},u_{1:t-1})}_{\text{prior filtering distribution}}. \tag{18}$$

In combination, (17) and (18) express the filtering distribution at time t as a function of the filtering distribution at time $t-1$. The sampling distribution term in (18) is given by the motion model (1). To compute the importance factor term in (17), observe that

$$\begin{aligned} \underbrace{p(z_t\,|\,x_{1:t},z_{1:t-1},u_{1:t})}_{\text{importance factor}} &= \int_h p(z_t,h\,|\,x_{1:t},z_{1:t-1},u_{1:t})dh \\ &= \int_h \underbrace{p(z_t\,|\,h,x_t)}_{\text{likelihood of observation}} \underbrace{p(h\,|\,x_{1:t-1},z_{1:t-1})}_{\text{prior map}} dh. \end{aligned} \tag{19}$$

Since we have assumed that each cell of the map is independent, we compute the importance factor separately for each grid cell that is nonempty in both the prior map and the current observation, then take the product over all of these grid cells to obtain the importance factor for the entire observation. To see how to compute the importance factor for a grid cell, note that (19) for a single map cell is precisely the observation likelihood (14), so we can compute it using (15).

VI. The Gamma-SLAM algorithm

At time step t, estimate the effective number of particles [11] using the particle weights from $t-1$: $J_{\text{eff}} = \frac{1}{\sum_{i=1}^{J}\left(w_{t-1}^{[i]}\right)^2}$. Then do the following for each particle $j = 1, 2, \ldots, J$.

- If $J_{\text{eff}} \geq \frac{J}{2}$: Let $\tilde{x}_{1:t-1}^{[j]} = x_{1:t-1}^{[j]}$, and let $\tilde{w}_{t-1}^{[j]} = w_{t-1}^{[j]}$.
- If $J_{\text{eff}} < \frac{J}{2}$ [Resampling step]: Sample (with replacement) a particle $\tilde{x}_{1:t-1}^{[j]}$ from the previous time step's particle set, $\{x_{1:t-1}^{[i]}\}_{i=1}^{J}$, with probability $w_{t-1}^{[i]}$. Set $\tilde{w}_{t-1}^{[j]} = \frac{1}{J}$.
- **Prediction:** Sample a new pose from the proposal distribution, given by the motion model (1):

$$x_t^{[j]} \sim p(x_t\,|\,\tilde{x}_{t-1}^{[j]},u_t). \tag{20}$$

Append the new pose onto the particle's path (pose history): $x_{1:t}^{[j]} = \{\tilde{x}_{1:t-1}^{[j]}, x_t^{[j]}\}$.

- **Measurement update:** Rotate and translate the observation z_t according to pose $x_t^{[j]}$, so that the observation aligns with the particle's existing map (so the grid cell boundaries are in the same locations). For each grid cell that is observed, compute the data statistics v, k, then combine this with the prior map for this cell (statistics v', k') by updating the map using (11). This gives the statistics v'', k'', which comprise that cell of the particle's map at time t.
- **Importance weight:** For each grid cell g that is nonempty in both the current observation and the prior map, compute an importance factor, $\lambda_{tg}^{[j]}$, using the logarithm of (15):

$$\begin{aligned} \log\lambda_{tg}^{[j]} &= \log\Gamma\left(\tfrac{1}{2}k''\right) - \log\Gamma\left(\tfrac{1}{2}k\right) - \log\Gamma\left(\tfrac{1}{2}k'\right) \\ &\quad + \tfrac{1}{2}\left[k\log(kv) + k'\log(k'v') - k''\log(k''v'')\right] - \log v. \end{aligned} \tag{21}$$

To compute the importance factor for the particle, we sum over the log importance factors of all grid cells that are nonempty in both the current observation and the prior map, then divide by a constant β times the number $\tilde{G}$ of such cells[1]:

$$\log\lambda_t^{[j]} = \frac{1}{\beta\tilde{G}}\sum_g \log\lambda_{tg}^{[j]}. \tag{22}$$

The new weight of the particle is equal to the product of its old weight and its importance factor:

$$w_t^{[j]} \propto \tilde{w}_t^{[j]} \cdot \lambda_t^{[j]}, \tag{23}$$

where at the end of the time step, the particle weights $w_t^{[j]}$ are normalized so $\sum_j w_t^{[j]} = 1$.

VII. Results

The robot was driven via remote control through two courses (course A and course B) on uneven sandy ground. On both courses, we ran Gamma-SLAM with 100 particles. To test the accuracy of the Gamma-SLAM system vs. visual odometry (VO) alone, we paused the robot several times during the course of the run and measured its ground truth position using a Total Station electronic surveying device. Table I shows the root-mean-squared position errors for each course, using Gamma-SLAM vs. using VO alone.

The objects in course A (shown in Figure 2, top) included orange traffic poles, plastic storage bins, rocks, and the corners of buildings. The robot was driven back and forth

[1] If all grid cells were truly independent as we assumed, then we could simply calculate the importance factor $\lambda_t^{[j]}$ using $\log\lambda_t^{[j]} = \sum_g \log\lambda_{tg}^{[j]}$. In practice, however, this equation results in an observation likelihood that is too steep (the best particle has almost all of the weight, whereas its competitors have almost none), which is probably due (at least in part) to the independence assumption. The factor $\frac{1}{\beta\tilde{G}}$ in (22) counteracts this effect.

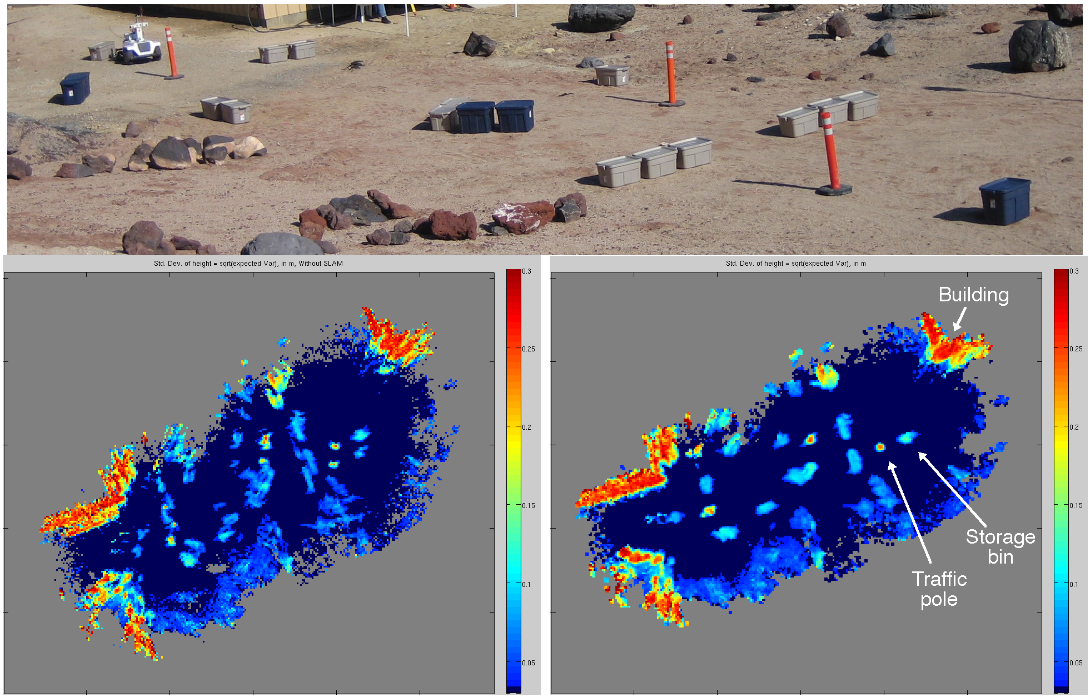
Std. Dev. of height = sqrt(expected Var), in m, Without SLAM
Std. Dev. of height = sqrt(expected Var), in m
0.3
0.25
0.2
0.15
0.1
0.05
Building
Storage bin
Traffic pole

Fig. 2. Course A. **Top:** Photo. **Lower left:** Map made using VO alone. **Lower right:** Map made by Gamma-SLAM with 100 particles. The left and right sides of the map correspond to the left and right sides of the photo, respectively. Color indicates the expected standard deviation of the heights in each grid cell. In the Gamma-SLAM map, the orange traffic poles appear as small yellow circles with orange centers. The plastic storage bins, which are shorter, appear in light blue, while the corners of the tall buildings (at the upper left and upper right of the map) appear in red-orange.

over this course three times during the course of the run. Figure 2 shows the map made using Gamma-SLAM (bottom right), as well as the map made without SLAM using the pose information from the visual odometry alone (lower left). We made this VO-only map by applying the same map update rules from Section IV using the pose taken directly from the visual odometry. In the VO-only map (lower left), there are two or three copies of each object, because each time it drove through the map, the robot's VO estimate of its position had shifted with respect to the true position. As a result, narrow gaps have been closed off in the map, which would make planning and navigation using this map virtually impossible. In contrast, the Gamma-SLAM map (lower right) provides an accurate map of the terrain.

By comparing the photo of course A (Figure 2, top) with the Gamma-SLAM map, we see that the map captures not only the locations of obstacles (as a binary grid occupancy map would), but also additional information about the heights of objects. The flatter to the ground an object is, the less height variance the system will observe in that cell, which will cause that grid cell to appear more blue in the map display. The taller an object is, the more will be the height variance of the points observed in that cell, and the cell will appear more red in the map display.

Course B (shown in Figure 3, top) consists of large loop, which the robot drove around three times during the course of the run. Course B contains four traffic poles (which roughly indicate the robot's path through the course) and a few plastic bins, but the course consists in large part of rocks of various sizes, from small rocks to boulders. Figure 3 (top) shows a photo of the course, with the robot facing one of the orange traffic poles. The map made using VO position information alone (Figure 3, lower left) is cluttered with three copies each object, because during each trip around the loop, the robot's VO estimate of its position had shifted with respect to the true position. In contrast, the map made using Gamma-SLAM (lower right) is clean. Once again, the heights of objects can be observed in the Gamma-SLAM map, from dark blue (small height variance) to red (large height variance).

TABLE I

ERROR FROM GROUND TRUTH ROBOT POSITIONS: GAMMA-SLAM VS. VISUAL ODOMETRY (VO) ALONE.

Course	**Distance traveled**	**RMS error (in m)**	
		VO only	Gamma-SLAM
Course A	164 m	1.24	**0.29**
Course B	204 m	1.12	**0.21**

VIII. Discussion

We have developed a new approach to SLAM that uses dense stereo vision and infers a map containing a posterior distribution over the variance of heights in each grid cell. Our results show significant improvement over visual odometry alone in an outdoor environment. The map is unique among SLAM systems due to the information that it contains, and the type of filter it uses to update this information. This type of map should be particularly useful in unstructured

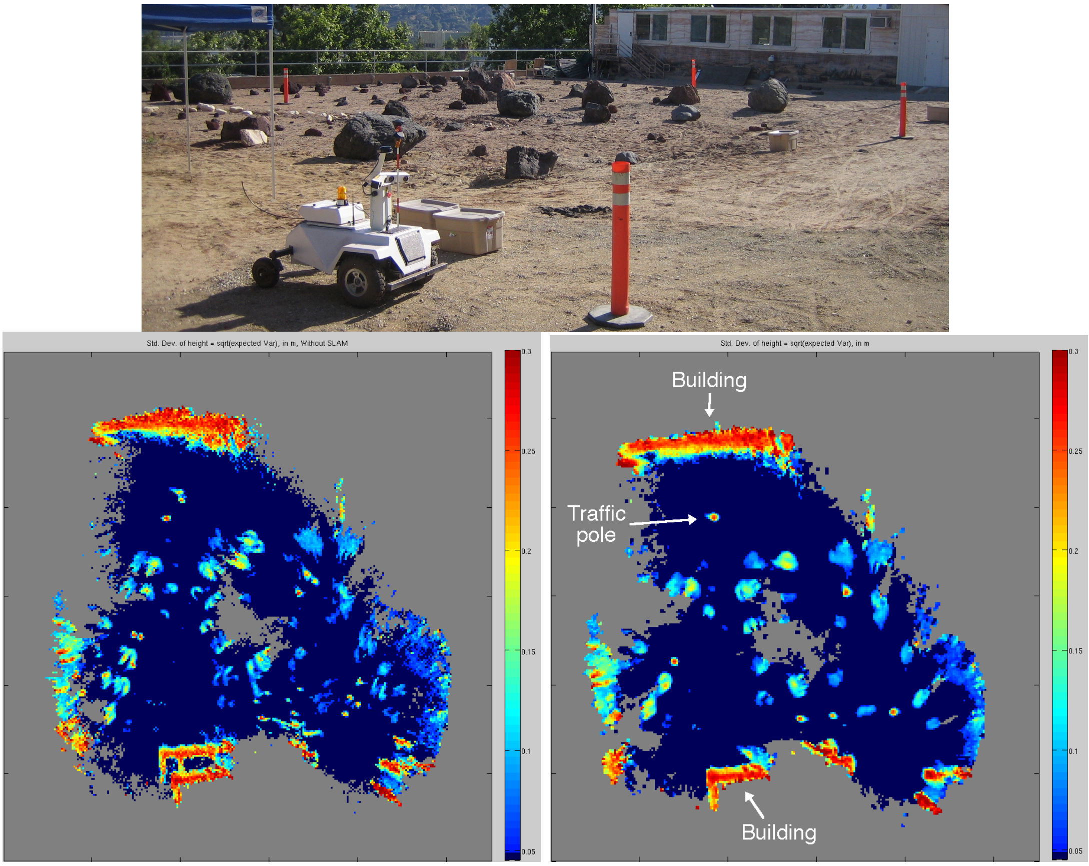
Std. Dev. of height = sqrt(expected Var), in m, Without SLAM
0.3
0.25
0.2
0.15
0.1
0.05
Std. Dev. of height = sqrt(expected Var), in m
Building
Traffic pole
Building
0.3
0.25
0.2
0.15
0.1
0.05

Fig. 3. Course B. **Top:** Photo. **Lower left:** Map made with VO alone. **Lower right:** Map made by Gamma-SLAM with 100 particles. The tall buildings (at the top and bottom of the Gamma-SLAM map) appear red-orange. The four orange traffic poles and the two white tent poles appear as small yellow circles with orange centers. The rocks vary in color depending upon their size: small rocks are dark blue, larger rocks are light blue, and the largest boulders are yellow-green.

outdoor environments, where binary occupancy maps do not provide sufficient information about the contents of a cell, and information necessary for planning may not be available in landmark-based maps. In the future, we will use Gamma-SLAM as part of a system for real-time autonomous robot navigation in unstructured outdoor environments.

IX. Acknowledgments

This work was performed for the Jet Propulsion Laboratory, California Institute of Technology, and was sponsored by the DARPA LAGR program through an agreement with the National Aeronautics and Space Administration. GWC is supported in part by NSF grant SBE-0542013.

References

[1] Michael Montemerlo and Sebastian Thrun. Simultaneous localization and mapping with unknown data association using FastSLAM. In *Proc. ICRA*, 2003.

[2] Dirk Haehnel, Wolfram Burgard, Dieter Fox, and Sebastian Thrun. An efficient FastSLAM algorithm for generating maps of large-scale cyclic environments from raw laser range measurements. In *IROS*, 2003.

[3] Pantelis Elinas, Robert Sim, and James J. Little. σSLAM: Stereo vision SLAM using the Rao-Blackwellised particle filter and a novel mixture proposal distribution. In *Proc. 2006 IEEE ICRA*, 2006.

[4] A. Doucet, N. de Freitas, K. Murphy, and S. Russell. Rao-Blackwellised particle filtering for dynamic bayesian networks. In *16th Conference on Uncertainty in AI*, pages 176–183, 2000.

[5] Stephen Se, Timothy Barfoot, and Piotr Jasiobedzki. Visual motion estimation and terrain modeling for planetary rovers. In *Proc. ISAIRAS 2005*, 2005.

[6] M. Dailey and M. Parnichkun. Simultaneous localization and mapping with stereo vision. In *Proc. ICARCV*, 2006.

[7] L. D. Jackel, Eric Krotkov, Michael Perschbacher, Jim Pippine, and Chad Sullivan. The DARPA LAGR program: Goals, challenges, methodology, and phase I results. *Journal of Field Robotics*, 24, 2007.

[8] Andrew Howard, Michael Turmon, Larry Matthies, Benyang Tang, Anelia Angelova, and Eric Mjolsness. Towards learned traversability for robot navigation: From underfoot to the far field. *Journal of Field Robotics*, 24, 2007.

[9] H. Hirschmuller, P.R. Innocent, and J.M. Garibaldi. Fast, unconstrained camera motion estimation from stereo without tracking and robust statistics. In *ICARCV'02*, pages 1099–1104, 2002.

[10] Howard Raiffa and Robert Schlaifer. *Applied Statistical Decision Theory*. Wiley Classics Library, 2000.

[11] A. Doucet, S. J. Godsill, and C. Andrieu. On sequential monte carlo sampling methods for bayesian filtering. *Statistics and Computing*, 10:197–208, 2000.