Robot-centric elevation mapping with uncertainty estimates

ROBOT-CENTRIC ELEVATION MAPPING WITH UNCERTAINTY ESTIMATES

P. FANKHAUSER*, M. BLOESCH, C. GEHRING, M. HUTTER and R. SIEGWART

Autonomous Systems Lab, ETH Zurich,
Zurich, 8092, Switzerland
**E-mail: pfankhauser@ethz.ch*
www.leggedrobotics.ethz.ch

This paper addresses the local terrain mapping process for an autonomous robot. Building upon an onboard range measurement sensor and an existing robot pose estimation, we formulate a novel elevation mapping method from a robot-centric perspective. This formulation can explicitly handle drift of the robot pose estimation which occurs for many autonomous robots. Our mapping approach fully incorporates the distance sensor measurement uncertainties and the six-dimensional pose covariance of the robot. We introduce a computationally efficient formulation of the map fusion process, which allows for mapping a terrain at high update rates. Finally, our approach is demonstrated on a quadrupedal robot walking over obstacles.

Keywords: Elevation Mapping; Terrain Estimation; Legged Robotics

1. Introduction

The ability to perceive and map an environment is an essential step to allow robots to navigate in unstructured and unknown terrain. While a lot of work has focused on obtaining a *globally*-consistent map,[1] our focus lies on retrieving a *local* map of the environment of the robot. We assume that a global path is given by a human operator or another algorithm and that the local map is used to plan the motion of the robot for a limited time horizon. Given a robot equipped with i) a range sensor and ii) state estimation capabilities, our task is to find a representation of the terrain that takes the uncertainties of the range measurements and the state estimation into account.

Since we are concerned with locally planning the robot's motion over or around obstacles, we are setting our focus on elevation mapping techniques, which simplify the terrain as a two-dimensional surface. Early work on the generation of elevation maps for autonomous legged robots was presented by Herbert et. al.[2] and Kweon et. al.[3] Their approaches build on a grid map in which each cell represents the height of the terrain in order to approximate the surface of the terrain. They use matching algorithms to find the corresponding transformation between multiple scans to build a composite elevation map, but they do not address the issues of error propagation as a result of the remaining error after matching. Cremean et. al.[4] developed an approach to fuse range measurements with uncertainties into a height map. When a range measurement is taken, cells that fall into the region of the measurement are updated based on previously stored data and the uncertainty of the measurement. Cells that do not receive measurement updates are left unchanged. This approach relies on absolute position measurements from GPS and is therefore unsuitable for our application. In the approach presented by Belter et. al.,[5] a local elevation map is used that surrounds the robot and is moved along with its motion. While we use a similar setup, their approach relies on a good pose tracking algorithm and does not address the issue of a drifting pose estimation. Our approach is similar to the work of Kleiner et. al.,[6] where the elevation map is deteriorated based on the motion of the robot. The uncertainty of the robot's position and orientation is reflected in the map by linearly growing the variance of the height estimate based on the accumulated distance and angle. This approach conservatively merges an approximation of the pose uncertainty into the height variance without taking the effect of in-plane uncertainty into account.

In contrast to related work, in which the elevation map is usually related to an inertial frame, we approach the elevation mapping from a robot-centric perspective: Our main contribution is the formulation of a mapping algorithm in which the elevation map is tightly coupled to the robot's motion. This leads to the important result that our method is able to specifically incorporate drift of the pose estimation. As opposed to methods that rely on matching new scans with the existing map, our formulation does not require (but can incorporate) a feedback-loop from the map building process to the state estimation. In Sec. 2, we formalize the elevation mapping framework with the treatment of the sensor measurement uncertainties and the six-dimensional pose covariance. Section 3 describes our implementation on an autonomous four-legged robot and presents the achieved results, before we conclude in Sec. 4. While we emphasize the application for legged robots, the presented approach is suitable for any robot that requires only local knowledge about its surrounding in order to plan its motion.

2. Method

We formulate our elevation mapping algorithm from a robot-centric view: Distance measurements are taken relative to the robot and when the robot moves, the entire elevation map is updated with information about the motion of the robot. At any time, the robot-centric elevation map is an estimate which the system has about the shape of the terrain from a local perspective. The region ahead of the robot has typically the highest precision as it is constantly updated with new measurements from the forward-looking distance sensor. On the other hand, regions which are out of the sensor's field of view (below or behind the robot) have decreased certainty due to drift of the robot's relative pose estimation.

In the following, we describe a method to obtain a robot-centric, local elevation map which consist of a two-dimensional grid in which each cell stores a height estimate $\hat{h}$ and variance $\hat{\sigma}_h^2$.

2.1. *Definitions*

To describe the framework, we introduce three coordinate frames, namely the Inertial frame I, the Sensor frame S, and the Map frame M (Fig. 1). The inertial frame I is fixed to the environment and we assume that the real terrain is stationary relative to this frame. The sensor frame S is attached to the distance sensor and is related to the inertial frame I through the translation $\boldsymbol{r}_{IS}$ and rotation $\boldsymbol{C}_{IS}$. The transformation between the inertial frame I and sensor frame S is obtained through the on-board state estimation and characterized by the six-dimensional pose covariance matrix $\boldsymbol{\Sigma}_P$. Finally, the elevation map frame M is defined through a relation to the sensor frame S with translation $\boldsymbol{r}_{SM}$ and rotation $\boldsymbol{C}_{SM}$, which are specified by the user. We choose the rotation $\boldsymbol{C}_{SM}$ such that the z-axis of the map frame M and the inertial frame I remain always aligned ($\boldsymbol{e}_z^I = \boldsymbol{e}_z^M$). The last degree of freedom, which corresponds to the yaw angle ψ between I and M (rotation around $\boldsymbol{e}_z^I$), is chosen to match the yaw angle between I and S.

2.2. *Measurement Update*

New measurements from the distance sensor are handled as points in space and mapped to the elevation map. This results in a new height measurement $\tilde{p}$ at a cell (x, y) in the height map. In the elevation map frame M, the height measurements are approximated by a Gaussian probability distribution as $\tilde{p} \sim \mathcal{N}(p, \sigma_p^2)$ with mean p and variance σ_p^2. A single measurement, given

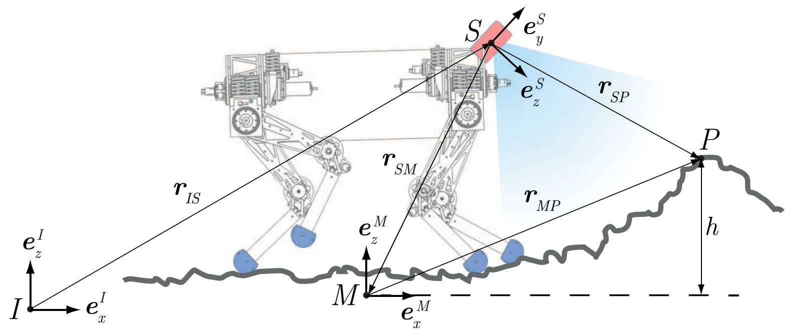

Fig. 1. Illustration of the coordinate frames used in the elevation mapping framework.

as the position ${}_{S}\boldsymbol{r}_{SP}$ in the sensor frame S, can be transformed to the corresponding height measurement p with

$$p = \boldsymbol{P}\left(\boldsymbol{C}_{SM}^{\mathsf{T}}(\boldsymbol{q})\,{}_{S}\boldsymbol{r}_{SP} - {}_{M}\boldsymbol{r}_{SM}\right) \ , \tag{1}$$

where the rotation between map frame and sensor frame $\boldsymbol{C}_{SM}$ is parameterized by the unit quaternion $\boldsymbol{q}$. The projection matrix $\boldsymbol{P} = [0\ 0\ 1]$ maps the three-dimensional measurement to the scalar height measurement p (in the map frame M). To obtain the variance of the height measurement σ_p^2, we derive the Jacobians for the sensor measurement $\boldsymbol{J}_S$ and the sensor frame rotation $\boldsymbol{J}_q$ from (1) as

$$\boldsymbol{J}_S = \frac{\partial p}{\partial\, {}_{S}\boldsymbol{r}_{SP}} = \boldsymbol{P}\boldsymbol{C}_{SM}^{\mathsf{T}}(\boldsymbol{q}) \qquad \text{and} \qquad \boldsymbol{J}_q = \frac{\partial p}{\partial \boldsymbol{q}} = \boldsymbol{P}\boldsymbol{C}_{SM}^{\mathsf{T}}\,{}_{S}\boldsymbol{r}_{SP}^{\times} \ , \tag{2}$$

where the superscript $^{\times}$ is used to denote the skew-symmetric matrix of a vector. The error propagation for the variance σ_p^2 is given as

$$\sigma_p^2 = \boldsymbol{J}_S \boldsymbol{\Sigma}_S \boldsymbol{J}_S^{\mathsf{T}} + \boldsymbol{J}_q \boldsymbol{\Sigma}_{P,q} \boldsymbol{J}_q^{\mathsf{T}} \ , \tag{3}$$

where $\boldsymbol{\Sigma}_S$ denotes the covariance matrix of the distance sensor model and $\boldsymbol{\Sigma}_{P,q}$ the covariance matrix of the sensor rotation. It is important to note, that uncertainties of the position of the sensor $\boldsymbol{\Sigma}_{P,r}$ do not need to be taken into account in this step because of our choice for the definition of the elevation map frame M (Sec. 2.1). Furthermore, an influence of the uncertainty of the yaw-rotation of the sensor ψ (rotation around $\boldsymbol{e}_z^I$) on the measurement is excluded because of the use of the projection matrix $\boldsymbol{P}$ and our definition of the map frame M.

The height measurement $(p,\, \sigma_p^2)$ is fused with the existing elevation map estimation $(\hat{h},\, \sigma_h^2)$ by means of a Kalman filter:[4]

$$\hat{h}^+ = \frac{\sigma_p^2\ \hat{h}^- + \hat{\sigma}_h^{2-}\tilde{p}}{\sigma_p^2 + \hat{\sigma}_h^{2-}} \ , \qquad \hat{\sigma}_h^{2+} = \frac{\hat{\sigma}_h^{2-}\,\sigma_p^2}{\hat{\sigma}_h^{2-} + \sigma_p^2} , \tag{4}$$

where estimates before an update are denoted with a $-$ superscript, respectively with a $+$ superscript if the update has already taken place.

If multiple measurements with different heights fall into the same cell (as it is in the case of a vertical wall), we employ an update rule similarly to the one presented by Kleiner et. al.[6] Based on the Mahalanobis distance, the rule fuses the measurements for the highest elevation and drops measurements that fall below a certain distance from the current estimate.

2.3. *Model Update*

As the elevation map frame M is defined in dependence of the motion of the sensor/robot (Sec. 2.1), the elevation map data needs to be updated whenever a motion of the robot relative to the inertial frame I has occurred. This is necessary because the real terrain is stationary in the inertial frame I and we want to approximate the terrain in the moving map frame M.

The mean $\hat{h}$ and variance values $\hat{\sigma}_h^2$ are updated according to the change of the pose estimate of the robot. Ideally, the variance and mean of each cell is updated depending on the increase of the uncertainty of the pose estimation as well as on the estimates of the surrounding cells. However, performing such an update for each cell of the map is computationally expensive. Instead, we extend the elevation map structure with information about the variance in the horizontal directions, $\hat{\sigma}_x^2$ and $\hat{\sigma}_y^2$ for the x- and y-directions, for each cell. Herewith, we can gather the full three-dimensional uncertainty for each point of the elevation map, and postpone the computationally expensive fusion (Sec. 2.4) to the time when the map is needed by the user or another algorithm.

Our goal is to derive the error propagation of the elevation map variances $[\hat{\sigma}_x^2 \ \hat{\sigma}_y^2 \ \hat{\sigma}_h^2]$ as a result of the change of the sensor pose covariance matrix $\boldsymbol{\Sigma}_P$ from time $k-1$ to k. To this end, we analyze the effect of the pose uncertainty onto the mapping from a fixed point in the inertial frame, ${}_I\boldsymbol{r}_{IP}$, to its representation in the elevation map, ${}_M\boldsymbol{r}_{MP}^k$ at timestep k:

$$ {}_S\boldsymbol{r}_{SP}^k = \boldsymbol{C}_{IS}^\mathsf{T}(\boldsymbol{q}^k)\left({}_I\boldsymbol{r}_{IP} - {}_I\boldsymbol{r}_{IS}^k\right) \ , \qquad {}_M\boldsymbol{r}_{MP}^k = \boldsymbol{C}_{SM}^\mathsf{T}\,{}_S\boldsymbol{r}_{SP}^k - {}_M\boldsymbol{r}_{SM} \ . \tag{5} $$

The Jacobians with respect to the sensor frame translation and rotation, $\boldsymbol{J}_r$ and $\boldsymbol{J}_q$, can be evaluated to

$$ \boldsymbol{J}_r = \frac{\partial_M\boldsymbol{r}_{MP}^k}{\partial_I\boldsymbol{r}_{IS}^k} = -\boldsymbol{C}_{SM}^\mathsf{T}\boldsymbol{C}_{IS}^\mathsf{T}(\boldsymbol{q}^k) \ , \tag{6} $$

$$ \boldsymbol{J}_q = \frac{\partial_M\boldsymbol{r}_{MP}^k}{\partial\boldsymbol{q}^k} = \boldsymbol{C}_{SM}^\mathsf{T}\boldsymbol{C}_{IS}^\mathsf{T}(\boldsymbol{q}^k)({}_I\boldsymbol{r}_{IP} - {}_I\boldsymbol{r}_{IS})^\times \ . \tag{7} $$

Now, the map for time k is updated based on the difference of the pose covariance matrices $\boldsymbol{\Sigma}_{P,r}$ and $\boldsymbol{\Sigma}_{P,q}$ from time $k-1$ to k according to

$$\begin{bmatrix} \hat{\sigma}_x^2 \\ \hat{\sigma}_y^2 \\ \hat{\sigma}_h^2 \end{bmatrix} = \begin{bmatrix} \hat{\sigma}_x^2 \\ \hat{\sigma}_y^2 \\ \hat{\sigma}_h^2 \end{bmatrix} + \operatorname{diag}\left(\boldsymbol{J}_r (\boldsymbol{\Sigma}_{P,r}^k - \boldsymbol{\Sigma}_{P,r}^{k-1}) \boldsymbol{J}_r^\mathsf{T} + \boldsymbol{J}_q (\boldsymbol{\Sigma}_{P,q}^k - \boldsymbol{\Sigma}_{P,q}^{k-1}) \boldsymbol{J}_q^\mathsf{T} \right) . \quad (8)$$

The estimates of the mean height $\hat{h}$ are kept unchanged as they still represent the best estimate of the mean. Note that while the change of full sensor position covariance matrix $\boldsymbol{\Sigma}_{P,r}$ is propagated on the elevation map variances, only the variances for the yaw-rotation ψ from the sensor rotation covariance matrix $\boldsymbol{\Sigma}_{P,q}$ (rotation around $\boldsymbol{e}_z^I = \boldsymbol{e}_z^M$) are propagated. This is complementary to the measurement update (Sec. 2.2), where the sensor position covariance and the sensor yaw-rotation variances are excluded from the update step.

2.4. *Map Fusion*

Whenever required, we can transform the elevation map data structure $(\hat{h}, \hat{\sigma}_h^2, \hat{\sigma}_x^2, \hat{\sigma}_y^2)$ to the corresponding original representation $(\hat{h}, \hat{\sigma}_h^2)$. This requires to infer the mean $\hat{h}$ and variance $\hat{\sigma}_h^2$ based on the data from all surrounding cells. We formulate the map fusion based on the rules for the combination of standard deviations for multiple populations as

$$\hat{h} = \frac{\sum\limits_n w_n \hat{h}_n}{\sum\limits_n w_n}, \qquad \hat{\sigma}_h^2 = \frac{\sum\limits_n w_n (\hat{\sigma}_{h,n}^2 + \hat{h}_n^2)}{\sum\limits_n w_n} - \hat{h}^2. \quad (9)$$

The weights w_n for a cell n in the neighborhood of the cell of interest is derived by considering the probability of cell n being positioned in the cell of interest as

$$w_n = \left(\Phi_x(d_x + \frac{r}{2}) - \Phi_x(d_x - \frac{r}{2}) \right) \left(\Phi_y(d_y + \frac{r}{2}) - \Phi_y(d_y - \frac{r}{2}) \right) , \quad (10)$$

where Φ_x and Φ_y denote the cumulative normal distribution with covariance σ_x and σ_y, and d_x and d_y represent the distance of cell n to the cell being updated. Variable r denotes the length of the cell side.

3. Results

We have implemented the presented elevation mapping method on the quadruped robot *StarlETH*.[7] The state estimation of the system is based on the fusion of kinematic and inertial measurements as presented by Bloesch et. al.[8] It was shown that the position and the rotation around the gravity

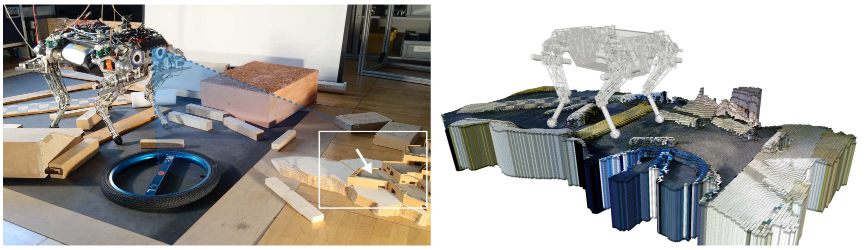

Fig. 2. The quadruped robot *StarlETH*[7] walks over obstacles (manually controlled) while the terrain is estimated with an elevation map.

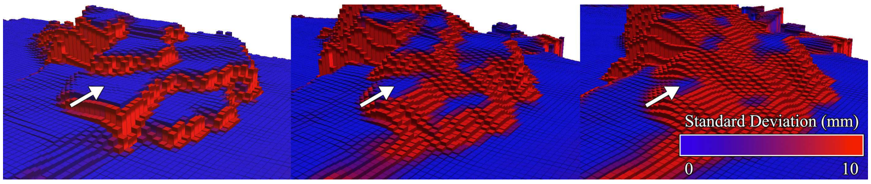

Fig. 3. The images show the development from left to right for a region of the height map (marked in Fig. 2) as the robot scans the terrain and than moves away from it.

axis (yaw angle) are in general unobservable and are therefore subject to drift. We attach a downward-facing *PrimeSense Carmine 1.09* structured light sensor in the front of the robot as distance sensor. We use a sensor measurement model that was introduced by Nguyen et. al.,[9] which models the noise in depth and lateral direction as a function of the measurement depth. The experimental setup is shown on the left of Fig. 2. We add obstacles of different heights (7–150 mm) and manually control the robot to walk over the test field in a walking gait at a speed of ∼0.05 m/s. The elevation map is set to a size of 2.5×2.5 m with a cell side length of 1 cm. The map is updated with the measurements and the model estimates at 20 Hz. For the map fusion, the process requires ∼0.3 s to fuse the entire elevation map (Intel Core i3, 2.60 GHz), but generally, only a sub-map is required.

An estimated elevation map for our setup is shown on the right of Fig. 2[a]. We conclude that the height information is well preserved when compared to the real setup. Very few holes are included in the map, which is a result of the high resolution of the sensor and the high frequency of the elevation update. One can observe that the region in front of the robot is well captured with crisp edges, while regions far behind the robot tend to smooth out with increased uncertainty. We illustrate this effect in Fig. 3, which shows a close-up of the region marked in Fig. 2. The three images of Fig. 3 depict the development over time (from left to right) as the robot scans this region and then moves away from it. One can observe that even un-

[a] A video demonstration is available at http://youtu.be/I9eP8GrMyNQ

der increased pose uncertainty of the robot, large flat regions remain at a relatively low standard deviation. This is an important feature that can be used for foothold selection in motion planning algorithms.

4. Summary and Future Work

We have presented a novel approach for elevation mapping that addresses the problem of pose estimate drift as it occurs on many autonomous robots. The presented method approaches the elevation map from a robot-centric perceptive, meaning that new measurements are introduced into the map at the precision of distance sensor and the estimation of the roll- and pitch angle of the robot. Through motion of the robot, the data in the map is updated based on the uncertainty estimates of the pose estimation of the robot. This gives the robot at any point in time an estimate of the terrain from its local perspective. We lower the computation burden of the mapping procedure by splitting the method into the data collection (measurement and model update) and the map fusion step. This enables real-time elevation mapping with high update frequencies.

We are currently working on a more detailed evaluation and comparison of the presented mapping method on different platforms.

This work was supported in part by the Swiss National Science Foundation (SNF) through project 200021_149427/1 and the National Centre of Competence in Research Robotics.

References

1. S. Thrun, Robotic mapping: A Survey, in *Exploring artificial intelligence in the new millennium*, 2002.
2. M. Herbert, C. Caillas, E. Krotkov, I. Kweon and T. Kanade, Terrain mapping for a roving planetary explorer, in *ICRA*, 1989.
3. S. Kweon and T. Kanade, *IEEE Trans. Pattern Anal. Machine Intell.* (1992).
4. L. Cremean and R. M. Murray, Uncertainty-based Sensor Fusion of Range Data for Real-time Digital Elevation Mapping (RTDEM), in *ICRA*, 2005.
5. D. Belter, P. Labecki and P. Skrzypczyski, Estimating terrain elevation maps from sparse and uncertain multi-sensor data, in *ROBIO*, 2012.
6. A. Kleiner and C. Dornhege, *Journal of Field Robotics* (2007).
7. M. Hutter, C. Gehring, M. Bloesch, M. A. Hoepflinger, C. D. Remy and R. Siegwart, StarlETH: A compliant quadrupedal robot for fast, efficient, and versatile locomotion, in *CLAWAR*, 2012.
8. M. Bloesch, C. Gehring, P. Fankhauser, M. Hutter, M. A. Hoepflinger and R. Siegwart, State Estimation for Legged Robots on Unstable and Slippery Terrain, in *IROS*, 2013.
9. C. V. Nguyen, S. Izadi and D. Lovell, Modeling Kinect Sensor Noise for Improved 3D Reconstruction and Tracking, in *3DIMPVT*, 2012.