Direct Generation of Regular-Grid Ground Surface Map From In-Vehicle Stereo Image Sequences

Shigeki Sugimoto Kouma Motooka Masatoshi Okutomi
Tokyo Institute of Technology
2–12–1–S5–22 Ookayama, Meguro-ku, Tokyo, 153–0834 Japan
shige@ok.ctrl.titech.ac.jp, kmotooka@ok.ctrl.titech.ac.jp, mxo@ctrl.titech.ac.jp

Abstract

We propose a direct method for incrementally estimating a regular-grid ground surface map from stereo image sequences captured by nearly front-looking stereo cameras, taking illumination changes on all images into consideration. At each frame, we simultaneously estimate a camera motion and vertex heights of the regular mesh, composed of piecewise triangular patches, drawn on a level plane in the ground coordinate system, by minimizing a cost representing the differences of the photometrically transformed pixel values in homography-related projective triangular patches over three image pairs in a two-frame stereo image sequence. The data term is formulated by the Inverse Compositional trick for high computational efficiency. The main difficulty of the problem formulation lies in the instability of the height estimation for the vertices distant from the cameras. We first develop a stereo ground surface reconstruction method where the stability is effectively improved by the combinational use of three complementary techniques, the use of a smoothness term, update constraint term, and a hierarchical meshing approach. Then we extend the stereo method for incremental ground surface map generation. The validity of the proposed method is demonstrated through experiments using real images.

1. Introduction

3D reconstruction of ground surfaces is one of the fundamental problems of mobile robotics, especially for the vehicles and robots traversing off-road environments. An important requirement from mobile robotics is the regular square grid representation of the ground surfaces, so-called Digital Elevation Map (DEM) [9, 17], where the ground geometry is represented by the vertex heights of a regular squared grid drawn on a level plane in the world (ground) coordinate system. A DEM can directly provide per-unit-length gradients of the ground, which is desirable for various real-time robot applications, including traversable area detection, path planning, and map extension. Our ultimate goal here is to develop an efficient method for incrementally estimating a DEM of a large ground area, no matter whether the target ground is off-road or not, from stereo image sequences captured by in-vehicle nearly-front-looking stereo cameras.

A DEM can be estimated by fitting a point cloud to a regular-grid surface model. Considered with the requirement of computational efficiency in robotics, a possible choice is to fit a regular-grid surface to the sparse point cloud estimated by an efficient stereo SLAM technique (*e.g.* [4, 8]), or to the denser point cloud estimated at each frame by a fast dense stereo method (*e.g.* [6, 11, 15]) followed by a visual odometry technique (*e.g.* [7]). Although these methods are effective for building 3D maps in urban environments, the validity for ground 3D geometry estimation is not sufficiently considered. For estimating ground surfaces, a global stereo approach is preferable to the local/semi-global approaches adopted in the previous methods, since roads and off-roads often have weakly textured surfaces and repeated patterns (*e.g.* wheel tracks). More importantly, for fast cost aggregation, these fast stereo techniques implicitly assume that the target surfaces are nearly front-parallel. This assumption is completely corrupted for the ground surface observed from in-vehicle front-looking cameras, as clearly depicted in [16].

Other than obtaining point clouds, our choice is to directly estimate a DEM and camera motion at each frame from a stereo image sequence. Its core idea lies in the previous stereo methods [12, 14] for static scenes, in which surface model parameters are estimated by direct image alignment. In these previous methods, however, the objective parameters are the depths of the regular-mesh vertices (or control points) drawn on a reference image, engendering a ground mesh irregularity such that the surface is more detailed near to the camera and rough far from the camera, when we simply apply the previous methods to the ground surface viewed from front-looking cameras. Moreover, in outdoor environments, extending the standard direct methods to the simultaneous estimation of a ground surface and camera motion from a stereo image sequence often fails in image registration, because the scene brightness changes dynamically and dramatically, especially when the sunny condition varies under scattered clouds.

In this paper we propose an efficient direct method for incrementally estimating a DEM of a large ground area using stereo image sequences captured by in-vehicle nearly-front-looking stereo cameras, while taking illumination changes on all images into consideration. At each frame, we estimate the visible vertex heights of a square grid mesh, in which each square is divided into two triangular patches, drawn on a level plane in the ground coordinate system, along with camera motion parameters and photometric parameters. Our cost function includes a data term representing the sum of the squared differences of photometrically transformed pixel values in homography-related projective triangular patches over three image pairs in the two-frame stereo image sequence. For improving the computational efficiency, the data term is formulated by the inverse compositional trick [1, 2, 12] for all objective parameters (*i.e.* surface, photometric, and motion parameters).

The main difficulty of this problem formulation lies in the instability of the height estimation of the distant vertices from the camera. This is because the pixel numbers in the image projection triangles of the distant patches are too small to contribute to the vertex height measurements. Although an additional smoothness constraint term somewhat improves the estimation stability, the use of only two terms cannot control *flaps* of the surface in the distant part through iteration optimization. Therefore, we first develop a stereo direct method [13] for robustly estimating a DEM (described in Section 2). We show that the stability can be effectively improved by the combinational use of an additional update constraint term and a hierarchical meshing approach. We also demonstrate the usability of the stereo method for mobile robots by showing traversable area detection results on the estimated ground surfaces. Then we extend this method for incremental DEM estimation (in Section 3).

2. Stereo Ground Surface Reconstruction

2.1. Preliminaries

We define coordinate systems as shown in Fig. 1. The coordinate relationships of the ground $\boldsymbol{x} = (x, y, z)^T$, a reference camera $\boldsymbol{x}_0 = (x_0, y_0, z_0)^T$, and the other camera $\boldsymbol{x}_1 = (x_1, y_1, z_1)^T$ are expressed by $\boldsymbol{x}_0 = \boldsymbol{R}\boldsymbol{x} + \boldsymbol{t}$ and $\boldsymbol{x}_1 = \boldsymbol{R}_s\boldsymbol{x}_0 + \boldsymbol{t}_s$, where $\boldsymbol{R}_s, \boldsymbol{t}_s, \boldsymbol{R}$, and $\boldsymbol{t}$ are assumed to be known.

We set a square grid mesh composed of triangular patches on the x-y plane of the ground system. Each patch and vertex position in the mesh are respectively denoted by $S_n, (n = 1, \cdots, N)$ and $\boldsymbol{x}_v = (x_v, y_v, z_v)^T, (v = 1, \cdots, V)$ where x_v and y_v are known. Let $\boldsymbol{z} \equiv (z_1, z_2, \cdots, z_V)^T$ represent the surface parameter vector to be estimated.

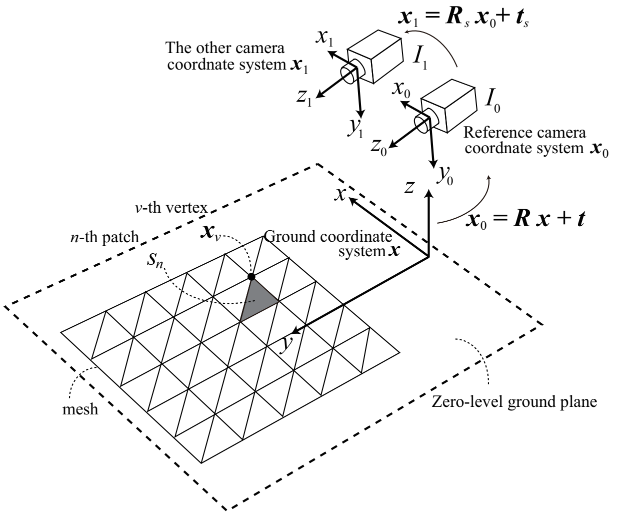

$\boldsymbol{x}_1 = \boldsymbol{R}_s\,\boldsymbol{x}_0 + \boldsymbol{t}_s$
The other camera coordnate system $\boldsymbol{x}_1$
x_1
I_1
z_1
y_1
x_0
I_0
Reference camera coordnate system $\boldsymbol{x}_0$
z_0
y_0
z
x
$\boldsymbol{x}_0 = \boldsymbol{R}\,\boldsymbol{x} + \boldsymbol{t}$
v-th vertex
n-th patch
$\boldsymbol{x}_v$
Ground coordinate system $\boldsymbol{x}$
S_n
y
mesh
Zero-level ground plane

Figure 1. Coordinate systems and ground mesh.

Let $I_0[\boldsymbol{u}]$ and $I_1[\boldsymbol{u}]$ be the pixel values (gray levels) of the reference image I_0 and the other image I_1, respectively, where $\boldsymbol{u} = (u, v)^T$ denotes an image point. For avoiding complexity, let $\boldsymbol{u}_0$ and $\boldsymbol{u}_1$ be in the *canonical* image configuration under the assumption that the stereo cameras are fully calibrated.

Considered with possible instability in the estimation of a large number of surface parameters, the pixel value differences between $I_0[\boldsymbol{u}_0]$ and $I_1[\boldsymbol{u}_1]$ at corresponding points $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$, mainly caused by the differences of device characteristics, are undesirable, even if the differences are several gray-levels, since the ground often have weakly textured surfaces. We adopt a widely-used photometric transformation represented by $I_0[\boldsymbol{u}_0] = \alpha I_1[\boldsymbol{u}_1] + \beta$, where α and β denote gain and bias, respectively. We write $I_0[\boldsymbol{u}_0] = \mathcal{P}(I_1[\boldsymbol{u}_1]; \boldsymbol{\alpha})$, where $\boldsymbol{\alpha} \equiv (\alpha, \beta)^T$ represent the photometric parameter vector also to be estimated.

2.2. Cost Function

We define a cost function as

$$C(\boldsymbol{z}, \boldsymbol{\alpha}) = C_D(\boldsymbol{z}, \boldsymbol{\alpha}) + C_S(\boldsymbol{z}), \qquad (1)$$

where C_D and C_S denote a data term and smoothness constraint term, respectively.

The data term C_D is the SSD (sum of squared differences) of the photometrically transformed pixel values in the homography-related projective triangular patches between I_0 and I_1.

$$C_D = \sum_{S'_n} \sum_{\boldsymbol{u} \in S'_n} \kappa_{\boldsymbol{u}} \big(I_0[\boldsymbol{u}] - \mathcal{P}\left(I_1[\boldsymbol{w}_n(\boldsymbol{u}; \boldsymbol{z})]; \boldsymbol{\alpha} \right) \big)^2, (2)$$

where S'_n represents the projection of the n-th triangular patch on the reference image, and $\kappa_{\boldsymbol{u}}$ denotes a binary mask indicating whether the pixel $\boldsymbol{u}$ is available or not (we set $\kappa_{\boldsymbol{u}}$ by checking the visibility of each patch and whether the norm of the image gradient vector at the pixel is smaller than a pre-defined threshold or not. The former checking is done in each iteration process). We denote by $\boldsymbol{w}_n(\boldsymbol{u}; \boldsymbol{z})$ a homography transform function of a reference image point $\boldsymbol{u}$, dropped in the projected triangle of the n-th surface patch. The function $\boldsymbol{w}_n(\boldsymbol{u}; \boldsymbol{z})$ is represented by a homography matrix $\boldsymbol{H}_n$ expressed by

$$\boldsymbol{H}_n = \boldsymbol{R}_s + \boldsymbol{t}_s \boldsymbol{m}_n^T(\boldsymbol{z}), \tag{3}$$

where $\boldsymbol{m}_n$ denotes the n-th patch's plane parameter vector define in the reference camera's coordinate system. We express by writing $\boldsymbol{m}_n(\boldsymbol{z})$ that $\boldsymbol{m}$ is a function of the n-th patch's three vertex heights in $\boldsymbol{z}$.

Since the mesh grid is regular, we adopt a simple smoothness term representing the sum of the squared Laplacian convolution outputs over the mesh.

$$C_S = \lambda_S |\boldsymbol{F}\boldsymbol{z}|^2, \tag{4}$$

where λ_S denotes a user-defined weight, and $\boldsymbol{F}$ denotes a $V \times V$ matrix whose v-th row $\boldsymbol{f}_v^T$ contains a 8-neighbor discrete Laplacian kernel for the v-th vertex. More specifically, the row vector $\boldsymbol{f}_v^T$ has an element 1 at the v-th vertex position, elements $-1/8$ at the 8-neighbors, and zeros at the others.

Since the smoothness term is defined in the surface parameter space, the smoothness constraints work uniformly over the mesh. On the other hand, the difference of the sizes of projected triangles between a surface patch near from the cameras and that far from the cameras results in the stronger contributions of the data term to the vertex height measurements for near patches. This is actually somewhat desirable because we can reconstruct in high precision the surface shape in the front area, which is more important for robot safety than a distant area, while the shape in the distant area is over smoothed but robustly estimated. However, the simple use of the two terms is insufficient for solving this problem. This is because there is an ambiguity in the surface reconstruction, such that the heights of neighboring distant vertices can go up and down at the same time while keeping flatness (*i.e.* both two terms take small values) during iterative cost minimization. We effectively improve the estimation stability by the additional use of update constraints and a hierarchical meshing approach as described later.

2.3. Optimization

2.3.1 Inverse compositional formulation

For high computational efficiency, we formulate the data term by using the inverse compositional trick [1, 2] for both surface and photometric parameters and apply a Gauss-Newton optimization algorithm.

We follow the work [12] for formulating the inverse compositional expression for the ground surface vector $\boldsymbol{z}$. We define the additive update rule for the surface parameter vector as $\bar{\boldsymbol{z}} \leftarrow \bar{\boldsymbol{z}} + \Delta\boldsymbol{z}$, where $\bar{\boldsymbol{z}}$ and $\Delta\boldsymbol{z}$ respectively denote a current estimate and update of $\boldsymbol{z}$. We denote by $\Delta\boldsymbol{w}_n(\boldsymbol{u}_0, \bar{\boldsymbol{z}}; \Delta\boldsymbol{z})$ a local homography transformation, which is only valid for a small parameter space around a given surface vector $\bar{\boldsymbol{z}}$. Under the compulsion that

$$\left(\boldsymbol{w}_n(\bar{\boldsymbol{z}}) \circ \Delta\boldsymbol{w}_n(\bar{\boldsymbol{z}}; \Delta\boldsymbol{z})^{-1} \right)(\boldsymbol{u}_0) \quad = \quad \boldsymbol{w}_n(\boldsymbol{u}_0; \boldsymbol{z}), \tag{5}$$

where $\circ$ denotes functional composition, the function $\Delta\boldsymbol{w}_n(\boldsymbol{u}_0, \bar{\boldsymbol{z}}; \Delta\boldsymbol{z})$ is represented by a homography matrix $\Delta\boldsymbol{H}_n$ as

$$\Delta\boldsymbol{H}_n = \boldsymbol{I} - a_n \boldsymbol{R}_s^T \boldsymbol{t}_s \Delta\boldsymbol{m}_n^T(\bar{\boldsymbol{z}}, \Delta\boldsymbol{z}), \tag{6}$$

$$\text{where } a_n = \frac{1}{1 + \left(\boldsymbol{m}_n(\bar{\boldsymbol{z}}) + \Delta\boldsymbol{m}_n(\bar{\boldsymbol{z}}, \Delta\boldsymbol{z})\right)^T \boldsymbol{R}_s^T \boldsymbol{t}_s} \tag{7}$$

Herein $\Delta\boldsymbol{m}_n(\bar{\boldsymbol{z}}, \Delta\boldsymbol{z})$ denotes an additive update of $\boldsymbol{m}_n$, as a function of a current and update vectors of the three vertices of the n-th patch. The additive update rule gives $\Delta\boldsymbol{m}_n \simeq (\partial\boldsymbol{m}_n / \partial\boldsymbol{z})\Delta\boldsymbol{z}$.

On the other hand, we follow [2] and adopt the inverse compositional update rule, $\bar{\boldsymbol{\alpha}} \leftarrow (\frac{\bar{\alpha}}{1+\Delta\alpha}, \frac{\bar{\beta}-\Delta\beta}{1+\Delta\alpha})^T$, where $\bar{\boldsymbol{\alpha}}$ and $\Delta\boldsymbol{\alpha}$ respectively denote a current estimate and update of $\boldsymbol{\alpha}$. For a pixel value p this update rule meets

$$\left(\Delta\mathcal{P}(\Delta\boldsymbol{\alpha})^{-1} \circ \mathcal{P}(\bar{\boldsymbol{\alpha}}) \right)(p) = \mathcal{P}(p; \boldsymbol{\alpha}), \tag{8}$$

where

$$\mathcal{P}(p; \bar{\boldsymbol{\alpha}}) = \bar{\boldsymbol{\alpha}} p + \bar{\boldsymbol{\beta}}, \tag{9}$$

$$\Delta\mathcal{P}(p; \Delta\boldsymbol{\alpha}) = (1 + \Delta\boldsymbol{\alpha})p + \Delta\boldsymbol{\beta}. \tag{10}$$

Then we rewrite the data term as

$$C_D(\Delta\boldsymbol{z}, \Delta\boldsymbol{\alpha}) = \sum_{S'_n} \sum_{\boldsymbol{u} \in S'_n} \kappa(\boldsymbol{u}) \big(\Delta\mathcal{P}\left(I_0[\Delta\boldsymbol{w}_n(\boldsymbol{u}, \bar{\boldsymbol{z}}; \Delta\boldsymbol{z})]; \Delta\boldsymbol{\alpha} \right) - \mathcal{P}\left(I_1[\boldsymbol{w}_n(\boldsymbol{u}; \bar{\boldsymbol{z}})]; \bar{\boldsymbol{\alpha}} \right) \big)^2 . (11)$$

2.3.2 Constraint on Δz and hierarchical meshing

The iterative estimation only with the data term and smoothness term still engenders *flaps* of distant parts of the surface in each iteration. This is because there is an ambiguity due to the recession of the data term in the estimation of distant vertex heights. For improving the estimation stability, we add the update constraint term representing the norm of $\Delta\boldsymbol{z}$.

$$C_U(\Delta\boldsymbol{z}) = \lambda_U|\Delta\boldsymbol{z}|^2, \tag{12}$$

where λ_U is a user-defined weight. It is possible to use a set of weights so as to enforce harder update constraints on more distant vertices. Eq. (12) indicates that we simply set identical weights considering that the recession of the data term actually plays a similar role as stronger update constraints on distant vertices.

A possible drawback of the update constraint term would be slow convergence. However, this simple scheme works very well for rapidly obtaining preferable reconstruction results, when we combine the term with a hierarchical meshing approach. We first roughly estimate the target ground surface using a mesh with large squares and the level-of-detail of the mesh is increased in stages (see Fig. 3) while keeping the same λ_S and λ_U. In the hierarchical meshing strategy, a current level-of-detail is initialized by using the result from the previous rougher level-of-detail, in which the data term is more dominant than the current level-of-detail. Therefore, the role of the update constraint term is not only to prevent the surface from flapping during iterative minimization, but also to keep the current surface as similar as possible to the surface estimated by the previous level-of-detail with a more dominant data term.

2.3.3 Computation of Δz

The Gauss-Newton Hessian of the stereo ground surface reconstruction is the summation of three Hessian matrices derived from the three terms. Although the data term is formulated by the inverse compositional trick, unfortunately, the Hessian of the data term should be re-computed in each iteration process, since the local function $\Delta\boldsymbol{w}_n(\boldsymbol{u}_0,\bar{\boldsymbol{z}};\Delta\boldsymbol{z})$ depends on a current estimate $\bar{\boldsymbol{z}}$ (Hessian matrices from the other two terms are constant). However, when we write $\partial I_0/\partial\Delta\boldsymbol{z} = (\partial I_0/\partial\Delta\boldsymbol{h}_n)(\partial\Delta\boldsymbol{h}_n/\partial\Delta\boldsymbol{z})$ where $\boldsymbol{h}_n$ denotes 9-vector of the homography parameters of the n-th patch, the pixel-dependent Jacobian matrix $\partial I_0/\partial\Delta\boldsymbol{h}_n$ is constant in each iteration, as indicated by [12]. On the other hand, $\partial\Delta\boldsymbol{h}_n/\partial\Delta\boldsymbol{z}$, derived from Eq. (6), is not constant but independent of pixel coordinates $\boldsymbol{u}$. That is, thanks to the inverse compositional trick, we do not need per-pixel Jacobian computations but per-patch Jacobian computations, which result in a much more computationally efficient algorithm than the case without the trick.

The computational cost at each iteration is increased as the mesh is detailed because of the increase of dimension of the linear system to be solved and the number of per-patch Jacobian computations. On the other hand, the number of iteration we need at each meshing level is generally quite few except the first roughest meshing level, since the resultant surface of the previous level is always a good initializer for a current one. In addition, for all meshing levels, the per-pixel Jacobians are constant, and thus we do not need any pre-computations at each meshing level except the first one.

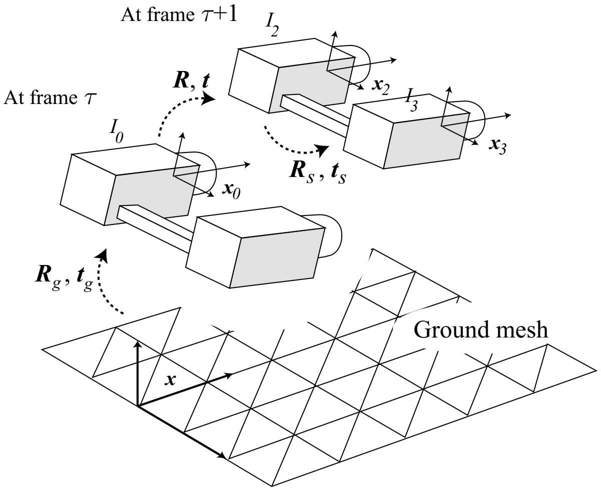


Figure 2. Moving stereo cameras

3. Motion Estimation and Map Generation

We extend the proposed stereo ground surface reconstruction method in Sec. 2 to ego-motion estimation and ground surface map generation. Fig. 2 shows the coordinate relationships of the ground and moving stereo cameras in consecutive frames. We estimate $\boldsymbol{R}, \boldsymbol{t}$ between $\boldsymbol{x}_0$ and $\boldsymbol{x}_2$ (*i.e.* the left camera at frame τ and at frame $\tau+1$), and then update $\boldsymbol{R}_g, \boldsymbol{t}_g$ between $\boldsymbol{x}$ and $\boldsymbol{x}_0$ (*i.e.* the ground and the left camera at frame τ). Let a camera motion vector we estimate be $\boldsymbol{\mu} \equiv (\boldsymbol{\omega}^T, \boldsymbol{t}^T)^T$, where $\boldsymbol{\omega}$ is a standard angle-axis vector of rotation.

We denote by I_2 and I_3 the images taken by the left and right cameras at time $\tau+1$. We also estimate the photometric parameters $\boldsymbol{\alpha}_2$ between image I_0 and I_2, and $\boldsymbol{\alpha}_2$ between image I_0 and I_3, in addition to $\boldsymbol{\alpha}_1$ between image I_0 and I_1.

3.1. Data term

We replace the data term in Eq. (2) with

$$C_D = \sum_{S'_n}\sum_{\boldsymbol{u}\in S'_n}\kappa_{\boldsymbol{u}}\sum_{j=1,2,3}D_j^2, \tag{13}$$

where

$$D_j = I_0[\mathbf{u}] - \mathcal{P}\left(I_j[\mathbf{w}_n^{(j)}(\boldsymbol{z},\boldsymbol{\mu})];\boldsymbol{\alpha}_j\right) \tag{14}$$

Herein we denote by $\boldsymbol{w}_n^{(j)}(\boldsymbol{z},\boldsymbol{\mu})$ homography functions of an image point $\boldsymbol{u}$. The above expression is somewhat informal since $\boldsymbol{w}_n^{(1)}$, identical to $\boldsymbol{w}_n(\boldsymbol{u};\boldsymbol{z})$ in Eq. (2), dose not depends on $\boldsymbol{\mu}$. The functions $\boldsymbol{w}_n^{(j)}(\boldsymbol{z}, \boldsymbol{\mu})$ are represented by homography matrices $\boldsymbol{H}_n^{(j)}$. In the case $j = 3$, we write

$$\boldsymbol{H}_n^{(3)} \quad = \quad \boldsymbol{R}_s\boldsymbol{R} + (\boldsymbol{R}_s\boldsymbol{t} + \boldsymbol{t}_s)\boldsymbol{m}_n^T(\boldsymbol{z}). \qquad (15)$$

Those in the cases of $j = 1, 2$ are derived from (15) by setting $\boldsymbol{R} = \boldsymbol{I}, \boldsymbol{t} = \boldsymbol{0}$ and $\boldsymbol{R}_s = \boldsymbol{I}, \boldsymbol{t}_s = \boldsymbol{0}$, respectively.

Although inverse compositional expressions and ESM-based direct ego-motion estimation for single plane tracking has been proposed (*e.g.* [5, 10]), these approach cannot be directly applied to the case with stereo epipolar constraints nor surface model parameter reconstruction. We rewrite (14) into inverse compositional forms by using the same manner to Eq. (5).

For the case $j = 3$, we write

$$\begin{aligned} D_3 = \Delta\mathcal{P}\left(I_0[\Delta\boldsymbol{w}_n^{(3)}(\boldsymbol{u}, \bar{\boldsymbol{z}}, \bar{\mu}; \Delta\boldsymbol{z}, \Delta\boldsymbol{\mu})]; \Delta\boldsymbol{\alpha}_3\right) - \\ \mathcal{P}\left(I_3[\boldsymbol{w}_n^{(3)}(\boldsymbol{u}; \bar{\boldsymbol{z}}, \bar{\mu})]; \bar{\boldsymbol{\alpha}}_3\right), \qquad (16)\end{aligned}$$

where $\Delta\boldsymbol{w}_n^{(3)}$ denots a local homography function. Then $\Delta\boldsymbol{w}_n^{(3)}$ is represented by a homography matrix $\Delta\boldsymbol{H}_n^{(3)}$ as

$$\Delta\boldsymbol{H}_n^{(3)} = (\boldsymbol{H}_n^{(3)})^{-1}\bar{\boldsymbol{H}}_n^{(3)} - \boldsymbol{I}, \qquad (17)$$

where

$$\begin{aligned} \boldsymbol{H}_n^{(3)} &= \boldsymbol{R}_s\bar{\boldsymbol{R}}\Delta\boldsymbol{R} + \\ &\quad (\boldsymbol{R}_s(\bar{\boldsymbol{R}}\Delta\boldsymbol{t} + \bar{\boldsymbol{t}}) + \boldsymbol{t}_s)(\bar{\boldsymbol{m}}_n + \Delta\boldsymbol{m}_n)^T, \qquad (18) \\ \bar{\boldsymbol{H}}_n^{(3)} &= \boldsymbol{R}_s\bar{\boldsymbol{R}} + (\boldsymbol{R}_s\bar{\boldsymbol{t}} + \boldsymbol{t}_s)\bar{\boldsymbol{m}}_n^T. \qquad (19)\end{aligned}$$

The dependency on $\Delta\boldsymbol{z}$ appears in $\Delta\boldsymbol{m}_n(\bar{\boldsymbol{z}}, \Delta\boldsymbol{z})$. Those in the cases of $j = 1, 2$ are also derived from (17).

Since the homography-based derivation indicates that we do not need to compute per-pixel Jacobian matrices in each iteration, the basic procedure in the optimization is the same as the stereo case described in 2.3.3.

3.2. Step-by-step estimation for robustness

In the sequential estimation of a ground surface and camera ego-motion using consecutive two frames, we can initialize the current surface at frame τ by using the surface and motion parameters estimated at the previous frame. However, a direct use of the cost (13) with the finest meshing level often engenders an undesirable result, especially when optical flows between two frames are totally large and/or scene brightness between two frames dramatically changes. The former is possibly engendered by a relatively large camera rotation, while the latter happens under clear weather with scattered clouds or by auto-brightness-control setting which we generally need on a standard camera for outdoor scenes. On the other hand, the stereo reconstruction method using hierarchical meshing presented in Sec. 2 is quite robust and successful. Considered with these advantage and drawbacks, we use a four-step estimation algorithm as follows.

We first estimate $\boldsymbol{z}$ and $\boldsymbol{\alpha}_1$ by the method described in Sec. 2. We initialize $\boldsymbol{z}$ at the finest meshing level by using the result estimated at the previous frame and then downsample the mesh into the roughest mesh (we initialize the height of each newcomer vertex by the same value at its nearest vertex which has a height). Starting with the roughest meshing level is somewhat redundant, but we take the advantage of its robustness for ground areas far from the cameras.

Then we estimate $\boldsymbol{\alpha}_2$ by the cost (14) of $j = 2$ and $\boldsymbol{\mu} = \boldsymbol{0}$, using $\boldsymbol{z}$ estimated at the previous step. Generally, a method for photometric parameter estimation based on [2] is successful even for a somewhat large brightness change. However, in an outdoor scene, an image brightness change is often too conspicuous to simultaneously estimate with motion parameters. For handling such a case we first adjust total image brightness of the two frames. Even though the camera motion vector is an incorrect one and the resultant $\boldsymbol{\alpha}_2$ would also be incorrect in this step, these parameters are recoverable after the decrease of a too large brightness difference.

Next, we estimate $\boldsymbol{\mu}$ and $\boldsymbol{\alpha}_2$ by the cost (14) of $j = 2$, using the same $\boldsymbol{z}$. We initialize $\boldsymbol{\mu} = \boldsymbol{0}$ and $\boldsymbol{\alpha}_2$ by the one from the previous step. We use a standared image pyramid approach [3] for handling a large image motion.

Finally, we estimate all parameters by the cost (1) whose data term is replaced with (13). We also add the update constraint term in the optimization process. We initialize $\boldsymbol{\alpha}_3 = \boldsymbol{\alpha}_2$, which is reasonable because image brightness of two stereo images is generally almost the same.

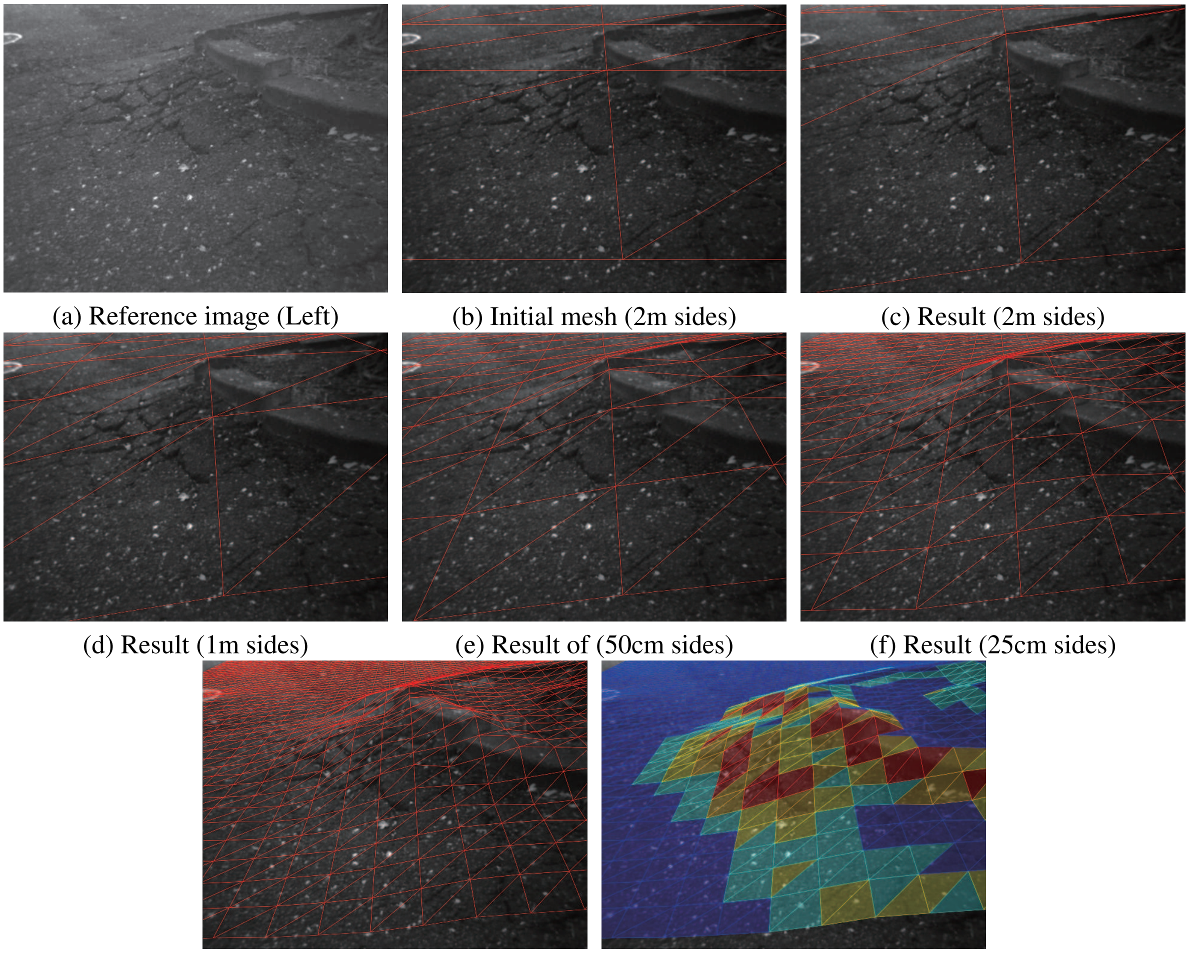
3.3. Ground Surface Map Generation

At each frame, we estimate a ground surface in a certain ground area (a region in x-y plane) in front of the camera. The area is simply defined by a rectangle, one of whose side is parallel to the perpendicular projection of the camera view direction on the x-y plane. For the first frame without any previous frame result, we initialize $\boldsymbol{z}$ by using the camera installation height (*i.e.* we set a *current* ground level in the ground coordinate system). The ground area generally includes invisible patches whose projection on the image I_0 lie outsize of the image. The vertex heights of such patches are estimated without the data term.

In each frame τ, we the compute weighted sums for all vertices estimated.

$$\hat{z}_v = \sum_\tau \omega_\tau z_{v\tau}, \text{ where } \omega_\tau = \frac{\frac{b_{v\tau}}{d_{v\tau}^2}}{\sum_{k=0}^\tau \frac{b_{vk}}{d_{vk}^2}} \qquad (20)$$

where $z_{v\tau}$ is the height of v-th vertex estimated at τ-th frame, $b_{v\tau}$ denote a binary flag representing whether the v-th vertex is visible at τ-th frame or not, and $d_{v\tau}$ is the distance between the v-th vertex position and the optical center



(a) Reference image (Left) (b) Initial mesh (2m sides) (c) Result (2m sides)

(d) Result (1m sides) (e) Result of (50cm sides) (f) Result (25cm sides)

(g) Result (12.5cm sides) (h) Patch clustering (12.5cm)

Figure 3. Result of the stereo ground surface reconstruction method (Sec. 2) using hierarchical meshing. The image (h) shows the angle between the plane normal and the z-axis of each patch in the final hierarchical meshing results (g) (12.5cm). Blue: smaller than 10. Cyan: 10$\sim$15, Yellow: 15$\sim$20, Orange: 20$\sim$25, Red: larger than 25 (in degrees).

of the left camera at τ-th frame. The computed height of each vertex is set to a ground surface map.

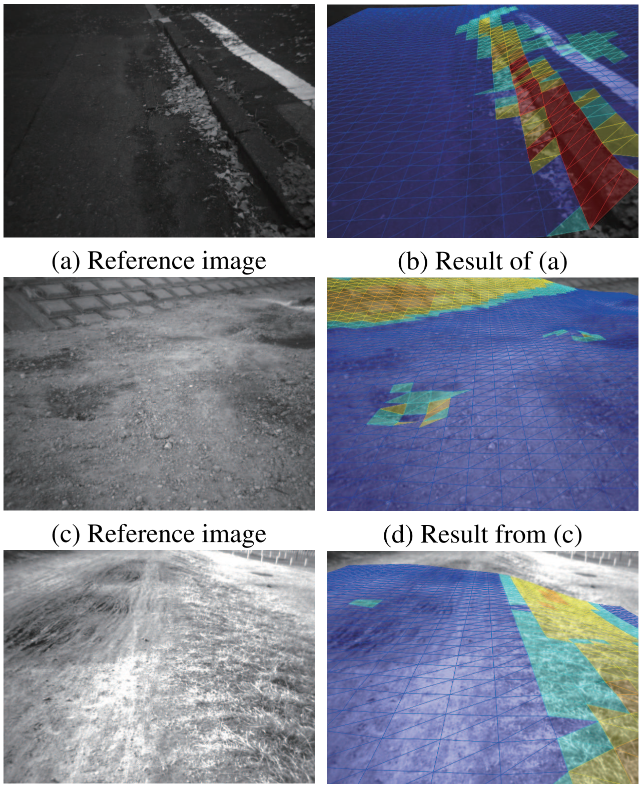
4. Experimental results

We first show ground surface reconstruction results for real environments by using the method described in Sec 2. Then we show ground surface map reconstruction results by using the method described in Sec 3. The algorithms were implemented in C++-language with single thread and run on a Windows7 PC (Xeon E3-1225 3.1GHz, 16GB). All images with the size of 640×480 pixels were captured by Point Gray Research Bumblebee2 with the baseline length of about 12cm, looking slightly down the ground at the height of about 1.0m. We experimentally set $\lambda_S = 10^4, \lambda_U = 5 * 10^3$ for all experiments.

4.1. Stereo surface reconstruction

Fig. 3 (a) shows an input reference (left) image which observes an asphalt road side where its ground level is raised by a long time growth of a tree in the right side of the scene. Fig. 3 (b) $\sim$ (g) shows results of our hierarchical meshing approach for the scene of (a). We set a mesh in the range of $\{-2 \leq x \leq 2\} \times \{1 \leq y \leq 9\}$ (in meters) in front of the reference camera, and started the estimation algorithm with a mesh grid with sides 2 meter long. At each level the target surface was well approximated. The final mesh (g) (with 12.5cm sides) recovered the raised ground level in the center area while keeping other flat areas very well. Fig. 3 (h) shows an patch clustering result from the result (g). Each color represents the angle between the plane normal of each patch and z-axis. The blue-colored patches indicate safely traversable area for mobile robots.

Fig. 4 shows three stereo reconstruction results for other



(a) Reference image (b) Result of (a)

(c) Reference image (d) Result from (c)

(e) Reference image (f) Result from (e)

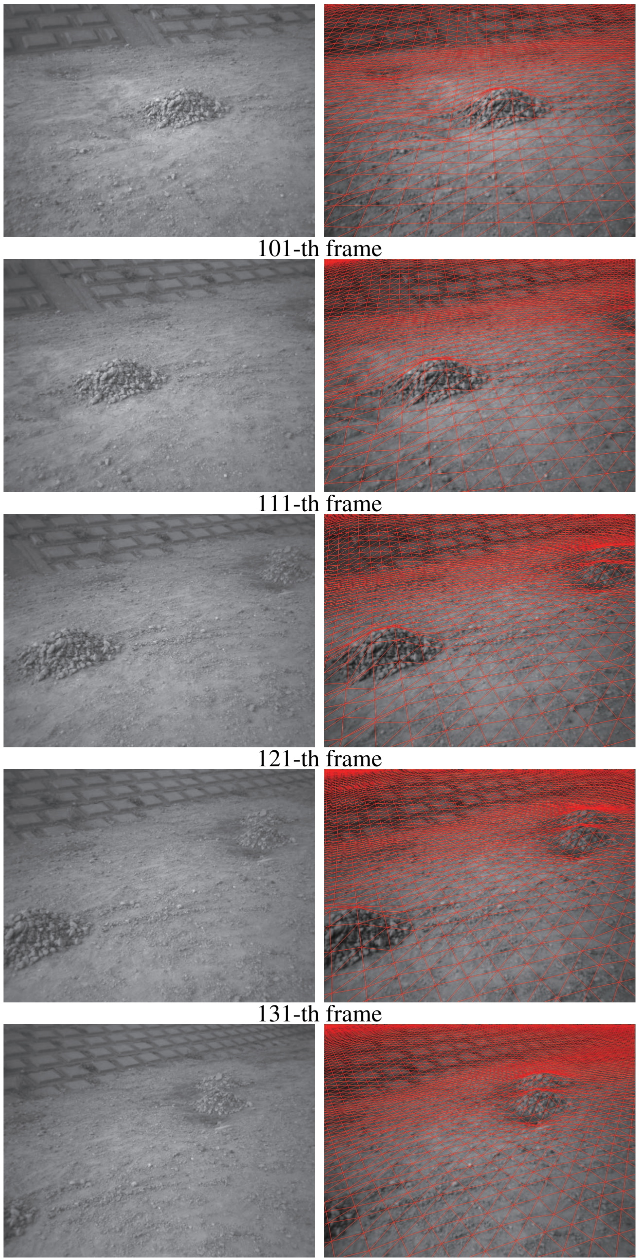
Figure 4. Results of the stereo ground surface reconstruction method (Sec. 2) for other three scenes. Left: reference image. Right: estimated surface and clustered colored patch.

scenes. Fig. 4 (a) shows an asphalt road with a sidewalk bump in the right of the scene. The reconstruction result (b) shows a preferably recovered the bump position. Fig. 4 (c) shows an off-road scene with very small hole and hillock on the ground surface, which were well recovered by the proposed method as shown in (d). Fig. 4 (e) and (f) also show an off off-road and a recovered large slope in the right part of the scene. Note that these colored regular-grid representations directly obtained from the stereo images are very helpful for traversable area detection and path-planning of robot systems.

The total computational time was about 1.2 second over the five hierarchical meshing levels (the final mesh had 3185 vertices and 6144 patches).

4.2. Motion estimation and Surface map generation

Fig. 5 shows a sequence of surface and motion estimation results for an off-road scene. The left image sequence is shown in the right column, and the ground surface estimate at each frame is overlapped in the right column. By using the camera ego-motion parameters estimated at each frame, these surfaces are well aligned on all images. We also show the ground surface map recovered from 300 frames in Fig. 6. We can see small holes and hillocks on the ground surface.



101-th frame

111-th frame

121-th frame

131-th frame

141-th frame

Figure 5. Surface and motion estimation result. Left: original images. Right: overlapped with estimated surface mesh.

5. Conclusions

We have proposed a method for directly reconstruct a ground surface with a regular square grid from stereo images. Then the method has been extended to motion estimation and ground surface map generation using stereo

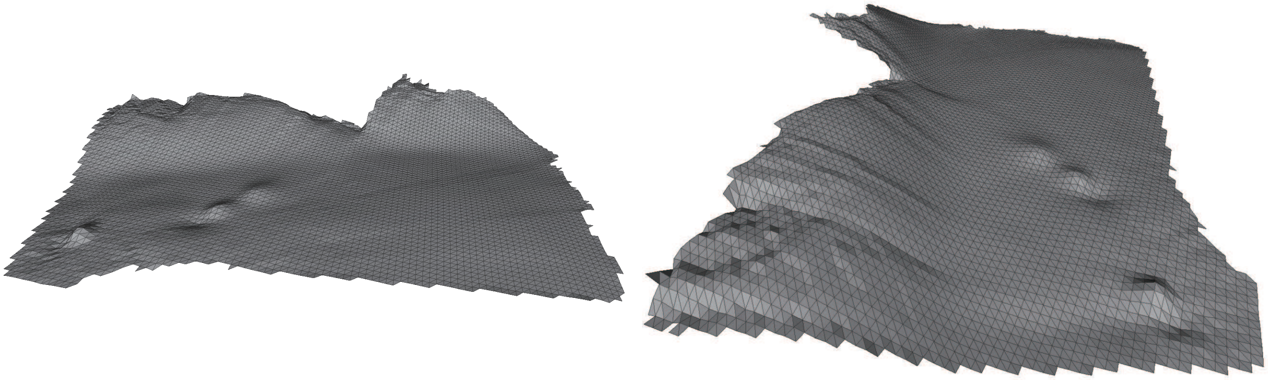


Figure 6. Ground surface map from Fig. 5. Two views from different positions

image sequences. We iteratively minimized a cost function composed of a data term formulated by the inverse compositional trick, a smoothness term, and an update constraint term, by Gauss-Newton optimization and a hierarchical meshing approach. The experimental results have shown that ground surfaces could be preferably recovered from stereo images even in the parts far from the cameras.

The current computational time for surface reconstruction using stereo images is promising for real-time applications since it is possible to highly parallelize the per-patch computation in our proposed method. Such an acceleration and fusion with feature-based methods will be studied during future research work.

Acknowledgement

This work was partially supported by the Grants-in-Aid for Scientific Research (No. 25240025) from the Japan Society for the Promotion of Science.

References

[1] S. Baker and I. Matthews. Lucas-kanade 20 years on: A unifying framework. *International Journal of Comoputer Vision*, 56(3):221–255, 2004. 2, 3

[2] A. Bartoli. Groupwise geometric and photometric direct image registration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(12):2098–2108, 2008. 2, 3, 5

[3] J. R. Bergen, P. Anandan, K. J. Hanna, and R. Hingorani. Hierarchical model-based motion estimation. In *European Conference on Computer Vision*, pages 237–252, 1992. 5

[4] M. C, G. Sibley, M. Chummins, P. Newman, and I. Reid. A constant-time efficient stereo slam system. In *British Machine Vision Conference*, 2009. 1

[5] D. Cobzas, M. Jagersand, and P. Sturm. 3D SSD tracking with estimated 3D planes. *Image and Vision Computing*, 27(1-2):69–79, 2009. 5

[6] A. Geiger, M. Roser, and R. Urtasun. Efficient large-scale stereo matching. In *Asian Conference on Computer Vision*, volume Part I, pages 25–38, 2010. 1

[7] B. Kitt, A. Geiger, and H. Lategahn. Visual odometry based on stereo image sequences with ransac-based outlier rejection scheme. In *IEEE Intelligent Vehicles Symposium*, pages 486–492, 2010. 1

[8] K. Konolige and M. Agrawal. Frameslam: From bundle adjustment to real-time visual mapping. *IEEE Transactions on Robotics*, 24(5):1066–1077, 2008. 1

[9] M. P. M. Vergauwen and L. V. Gool. A stereo-vision system for support of planetary surface exploration. *Machine Vision and Applications*, 14(1):5–14, 2003. 1

[10] C. Mei, S. Benhimane, E. Malis, and P. Rives. Efficient homography-based tracking and 3d reconstruction for single viewpoint sensors. *IEEE Transactions on Robotics*, 24(6):1352–1364, 2008. 5

[11] X. Mei, X. Sun, M. Zhou, S. Jiao, H. Wang, and X. Zhang. On building an accurate stereo matching system on graphics hardware. In *IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pages 467–474, 2011. 1

[12] S. Sugimoto and M. Okutomi. A direct and efficient method for piecewise-planar surface reconstruction from stereo images. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1–8, June 2007. 1, 2, 3, 4

[13] S. Sugimoto and M. Okutomi. Direct ground surface reconstruction from stereo images. *IPSJ Transactions on Computer Vision and Applications*, 5:60–64, 2013. 2

[14] R. Szeliski and J. Coughlan. Spline-based image registration. *International Journal of Computer Vision*, 22(3):199–218, 1997. 1

[15] L. Wang, R. Yang, M. Gong, and M. Liao. Real-time stereo using approximated joint bilateral filtering and dynamic programming. *Journal of Real-Time Image Proccessing*, pages 1–15, 2012. 1

[16] T. Williamson and C. Thorpe. A specialized multibaseline stereo technique for obstacle detection. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 238–244, 1998. 1

[17] Z. Zhang. A stereovision system for a planetary rover: calibration, correlation, registration, and fusion. *Machine Vision and Applications*, 10(1):27–34, 1997. 1