Efficient 7D Aerial Pose Estimation

Bertil Grelsson[1,2]
[1]Saab Dynamics
Saab, Linköping, Sweden
bertil.grelsson@liu.se

Michael Felsberg[2]
[2]Computer Vision Laboratory
Linköping University, Sweden
michael.felsberg@liu.se

Folke Isaksson
Vricon Systems
Saab, Linköping, Sweden
folke.isaksson@saabgroup.com

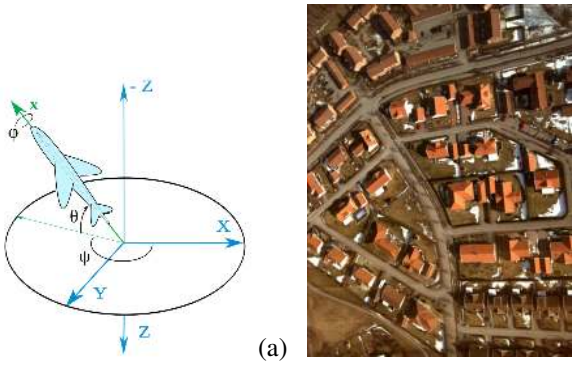
Abstract

A method for online global pose estimation of aerial images by alignment with a georeferenced 3D model is presented. Motion stereo is used to reconstruct a dense local height patch from an image pair. The global pose is inferred from the 3D transform between the local height patch and the model. For efficiency, the sought 3D similarity transform is found by least-squares minimizations of three 2D subproblems. The method does not require any landmarks or reference points in the 3D model, but an approximate initialization of the global pose, in our case provided by onboard navigation sensors, is assumed. Real aerial images from helicopter and aircraft flights are used to evaluate the method. The results show that the accuracy of the position and orientation estimates is significantly improved compared to the initialization and our method is more robust than competing methods on similar datasets. The proposed matching error computed between the transformed patch and the map clearly indicates whether a reliable pose estimate has been obtained.

1. Introduction

Automatically reconstructed large scale 3D models or digital surface models (DSMs) have become commonly available. The 3D reconstructions may be generated from images captured from satellites [**?**], unmanned aerial vehicles (UAVs) [**?**] or from a large number of public photographs [**?**]. Numerous applications making use of the 3D models are conceivable, many of them involving geolocation of an airborne camera. Micro aerial vehicles (MAVs) have become very popular as a remote sensing platform. The MAVs often have GPS and navigation sensors, but to achieve accurate geolocation over time, image based refinement is required.

Our specific problem is to determine the absolute global pose, fig.**??**(a), and the scale (7DoF) of aerial images where a calibrated camera and an onboard inertial measurement unit (IMU) are available. Since we are looking for a method to be processed onboard a small UAV, and which supports online guidance of the vehicle, a hard constraint is a computationally inexpensive and memory efficient real-time solution. Further, we avoid using GPS as input since this information is not always reliable in high-rise environments and can make the method less robust in case of GPS outage or jamming. Lastly, we want the method to be season invariant, *i.e*. the method should allow images captured in one season to be aligned with a 3D model acquired in another season when *e.g*. vegetation may look different. We present a new method to estimate the global pose of aerial images, where an IMU estimate of the global pose is used as an initialization. Due to drift in the IMU, the accuracy of the pose estimate will degrade with time unless supported with other sensor data. A dense local height patch of the ground, computed with motion stereo (MS), is aligned with a georeferenced 3D model of the corresponding area. Aligning height information is assumed to be more robust to season variations than a single-view based approach. The global pose is inferred from the 3D similarity transform between the local height patch and the 3D model. The main novelty of the method is a framework that enables the 3D similarity transform to be reliably and robustly estimated by solving three 2D subproblems. The reason that this decomposition



(a) (b)

Figure 1. (a) Definition of world coordinate frame and vehicle pose. X = north, Y = east, Z = down, ψ = yaw, θ = pitch, ϕ = roll. (b) Aerial image from helicopter flight.

works is that the pitch and roll angle initialization errors are relatively small, a few degrees at maximum. The pose initialization errors are dominated by the translation, yaw angle and scale errors, which are all considered in the first 2D subproblem. The method has been evaluated using real aerial images from helicopter and aircraft flights over urban areas, fig.**??**(b). A matching error indicating the reliability of the pose estimate is proposed and analyzed.

2. Related work

Since we are only interested in methods that register images with a 3D model captured at a different time, we do not consider methods like KinectFusion [**?**] or SLAM [**?**].

Pose estimation of an airborne camera by registration of aerial images with a DSM is a well studied area. One group of methods extract features in single images and register the features with the 3D model, another group use multi-view images and perform 3D-3D registration.

Ding *et al*. [**?**] use vertical vanishing points and navigation sensors to obtain a coarse registration of aerial images with a 3D LiDAR model. The registration is then refined by extracting 2D corners in the image and matching these with 3D corners in the model. The registration rate is 91% in a downtown area but falls to 50% in a residential area.

Mastin *et al*. [**?**] use mutual information (MI) to register aerial images with a 3D LiDAR model. They assume navigation sensors to give a coarse camera pose. They project the 3D model onto the image plane and maximize the MI between optical and LiDAR features. The registration rate is as high as 98.5% in an urban area. Processing time is in the order of 10 s when using GPU for 3D rendering.

Zhao *et al*. [**?**] register a dense 3D point cloud from video motion stereo with 3D range LiDAR data using an iterative closest point (ICP) algorithm to recover camera orientation. The method proved successful but ICP is computationally expensive and is not suitable for real-time solutions. Wendel *et al*. [**?**] used a MAV to collect images and their 3D reconstruction of buildings is aligned with a DSM using a coarse-to-fine correlation scheme over all unknown transformation parameters. Their method yields accurate alignments but is not suited for real-time solutions.

Gruen and Akca [**?**] derive a Gauss-Markov model for least-squares 3D surface matching. Their method gives accurate results for good initializations but it is too computationally demanding for a real-time solution. Lerner *et al*. [**?**] use a DSM to formulate a constraint between feature pairs in successive images. They approximate the terrain with the tangent plane of the DSM around the assumed feature point position and solve for the camera pose and motion. A computational advantage is that no 3D reconstruction is needed. On a coarse grid lab model of a mountainous area their method works fine as their model assumption applies, which however is not the case in urban areas.

3. Pose estimation method

Our method needs to be season invariant, and we assume that aligning a local height map with the 3D model will be more robust than a single-view based method trying to match an image with a projection from the 3D model. However, our method requires altitude differences in the scene. Our approach to estimate the camera pose consists of two main steps. First we compute a dense local height patch of the ground from an image pair. The second step is to align the local height patch with the 3D model. Due to initialization errors and as the scale is only roughly known, the local height patch is distorted by a 3D similarity transform compared to the 3D model. Our method estimates the transformation parameters to obtain a best match between the local height patch and the 3D model. From this alignment process, we can infer a more accurate estimate of the camera's global pose.

To generate a dense local height patch we use MS on two images from the airborne camera, where the stereo baseline is in the order of 1/10 of the height above the ground. We make an assumption on the global pose for image 1. The intrinsic camera parameters are known from calibration. We assume that the relative rotation between the images is accurately given by the IMU. The translation vector direction is estimated from the images. We compute the local height patch with a stereo method which is based on [**?**] but has been further developed and has proven to give state-of-the-art results when generating 3D models from aerial imagery [**?**]. The stereo output is a dense depth map which is orthorectified and converted to a georeferenced local height patch given the global pose assumption for image 1.

3.1. Translation vector direction estimate

To estimate the translation vector direction from an image pair is a well known problem and in our case the relative rotation between the images is known with good accuracy from IMU data. We estimate the translation vector in three steps: (1) Determine feature points. (2) Match feature points. (3) Determine the translation vector using the epipolar constraint.

Feature point detector. As we want to match airborne images taken with a frame rate around 1 fps, we are not dependent on scale and rotation invariance in the feature point matching process. The rotation angle from one image to the next is at most a few degrees. We use the FAST-12 detector [**?**] to find feature points. As FAST tends to generate many feature points along edges, we suppress these points by using the idea from [**?**] and employ a Harris corner measure to only retain points with a large Harris score.

Matching feature points. Since we have an approximate estimate of the vehicle velocity, we have a rough idea how the center pixel in image 1 has moved into image 2. To refine this first rough guess on the overall optical flow

vector, we run a Lucas-Kanade tracker (KLT) [**?**] at coarse scale. In the aerial image sequences used, the lighting conditions are very similar from one image to the next and the exposure parameters are set constant. Hence, KLT can be used for matching. If the brightness constancy is expected to be violated, normalized correlation methods may be more appropriate.

For each feature point p in image 1 we search for potential corresponding feature points q in image 2 within a radius r such that the pixel distance d is less than a threshold set to 30 pixels. For each potential corresponding point q_i within this disc, we run a back-and-forth KLT over an 11x11 patch between the points p and q_i. If the back-and-forth displacements deviate less than 0.25 pixels, we accept them as corresponding points.

As the FAST detector only gives integer pixel values, we use the displacement $\mathbf{d}$ from the KLT to obtain subpixel accuracy. The feature points are converted to normalised image coordinates $\mathbf{x} = [x \;\; y \;\; 1]^T$ and $\mathbf{x}' = [x' \;\; y' \;\; 1]^T$.

Epipolar constraint. The essential matrix $\mathbf{E}$ is defined as in [**?**]

$$\mathbf{E} = [\mathbf{t}]_{\mathrm{x}}\mathbf{R} \tag{1}$$

where $\mathbf{R}$ and $\mathbf{t}$ are the relative rotation and the translation for the camera between images 1 and 2, and $[\cdot]_{\mathrm{x}}$ is the cross-product operator. For corresponding points the epipolar constraint

$$\mathbf{x}^T\mathbf{E}\mathbf{x}' = 0 \;\;, \tag{2}$$

must be satisfied. If we expand this equation and remember that the rotation matrix is known, we obtain a linear equation system for $\mathbf{t}$, also found in [**?**].

$$\begin{bmatrix} (r_{12}x + r_{22}y + r_{32}) - y'(r_{13}x + r_{23}y + r_{33}) \\ x'(r_{13}x + r_{23}y + r_{33}) - (r_{11}x + r_{21}y + r_{31}) \\ y'(r_{11}x + r_{21}y + r_{31}) - x'(r_{12}x + r_{22}y + r_{32}) \end{bmatrix}^T \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} = 0 \tag{3}$$

We use a RANSAC scheme to suppress outliers among the feature point matches. In each iteration we randomly pick two points from the matching point set. We solve for $\mathbf{t}$ and given the resulting $\mathbf{E}$ we define the consensus set as all points within 0.25 pixels from the epipolar lines. The value on $\mathbf{t}$ giving the largest consensus set is taken as the translation vector direction. Mathematically the solution is ambiguous to the sign of the translation vector, but the sign is evident from the forward direction of the vehicle.

3.2. Alignment of height patch with 3D model

To align the local height patch with the 3D model, we use an approach similar to the tracking part in [**?**], but in our case the images to be aligned represent height information and not intensity values. We split the 3D alignment process into three steps; first alignment in the ground plane (XY), then in the XZ-plane and finally in the YZ-plane. According to [**?**, **?**], a high dimensional least-squares problem can be decomposed into several subspace problems or even single coordinate descent without hampering convergence. In general, this requires iterating between the coordinates or selecting the subspace of maximal descent in an outer loop and line-search in an inner loop. In our case the inner loop is iterated from coarse to fine scale. The outer loop is simplified by minimizing the subspace with dominating errors (x, y, yaw and scale) first and then minimizing the two subspaces with smaller errors (z, pitch and roll). Outer minimization is only iterated for the first subspace (XY), since further iterations on the other two subspaces did not improve results further.

The reduction to 2D is also very memory efficient. The full 3D volume to be aligned is often around 300 Mvoxels whereas the 2D areas only require 1 Mpixels.

Preliminaries. The equation systems to be solved in the three planes will all take the general form

$$\int_W \mathbf{U}\,\omega(\mathbf{x})\,d\mathbf{x} \;\; \mathbf{v} = \int_W \mathbf{w}\,\omega(\mathbf{x})\,d\mathbf{x} \tag{4}$$

where $\mathbf{U}$ is a model matrix, $\mathbf{v}$ is the sought parameter vector and $\mathbf{w}$ is a residual. They will be derived for each plane.

For a $p \times q$ matrix $\mathbf{A}$ and an $m \times n$ matrix $\mathbf{B}$, the Kronecker product $\mathbf{A}\otimes\mathbf{B}$ is a $pm \times qn$ block matrix

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} a_{11}\mathbf{B} & \cdots & a_{1q}\mathbf{B} \\ \vdots & & \vdots \\ a_{p1}\mathbf{B} & \cdots & a_{pq}\mathbf{B} \end{bmatrix} \tag{5}$$

Alignment in ground plane (XY-plane). Based on the assumptions on XY position and heading of the vehicle, we cut out a map from the 3D model with the same size as the local height patch. We minimize the error function ϵ_{XY}, the squared difference in height between the local height patch $h_p(\mathbf{x})$ and the map $h_m(\mathbf{x})$,

$$\epsilon_{XY} = \int_W \left[h_p((\mathbf{I}+\mathbf{D})\mathbf{x} + \mathbf{d}) - h_m(\mathbf{x})\right]^2 \omega(\mathbf{x})\,d\mathbf{x} \tag{6}$$

where $\mathbf{A} = \mathbf{I} + \mathbf{D}$ denotes a linear transformation, $\mathbf{x}$ is a point in the image, $\mathbf{d} = [d_x \;\; d_y]^T$ is a displacement and ω is a window function. We use a separable Hanning window in the x and y directions. Most important for the window function is to suppress boundary interpolation effects. The integration domain W is the whole patch area.

The error function in (**??**) is minimized by linearising the integrand and finding a least-squares solution. We make a Taylor expansion of the linear transformation and truncate to the linear term

$$h_p(\mathbf{A}\mathbf{x} + \mathbf{d}) \approx h_p(\mathbf{x}) + \mathbf{g}^T\mathbf{u} \tag{7}$$

$$\mathbf{g} = \begin{bmatrix} \frac{\partial h_p}{\partial x} & \frac{\partial h_p}{\partial y} \end{bmatrix}^T = \begin{bmatrix} g_x & g_y \end{bmatrix}^T, \;\; \mathbf{u} = \mathbf{D}\mathbf{x} + \mathbf{d} \tag{8}$$

To minimize the residual ϵ_{XY}, we differentiate it with respect to all unknowns in the deformation matrix $\mathbf{D}$ and the displacement vector $\mathbf{d}$ and set the result to zero. The general entities in (**??**) for the XY-plane alignment read

$$\mathbf{U_{XY}} = \left(\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \begin{bmatrix} x & y & 1 \end{bmatrix} \right) \otimes \left(\begin{bmatrix} g_x \\ g_y \end{bmatrix} \begin{bmatrix} g_x & g_y \end{bmatrix} \right) \quad (9)$$

$$\mathbf{v_{XY}} = \begin{bmatrix} D_{xx} & D_{yx} & D_{xy} & D_{yy} & d_x & d_y \end{bmatrix}^T \quad (10)$$

$$\mathbf{w_{XY}} = \begin{bmatrix} h_m - h_p \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \otimes \begin{bmatrix} g_x \\ g_y \end{bmatrix} \quad (11)$$

The equation system is solved iteratively at coarse-to-fine scales. For each scale we stop the iterations when the decrease in ϵ_{XY} is less than 1% or the max of 10 iterations is reached. After each scale we update the XY position with d_x and d_y, and the heading ψ with the angle between the y-axis and the $\mathbf{D}$-transformed y-axis. The deformation matrix $\mathbf{D}$ allows for a more general linear transformation instead of a 2D similarity transformation to cope with the errors in the pitch and roll angles.

Prior term in cost function. To constrain the unknown parameters not to drift away in the search algorithm, we introduce a prior term in the cost function, (i, j = x, y),

$$\epsilon_{XY,prior} = \sum_{i,j} \frac{(D_{ij} - D_{ij0})^2}{\sigma^2_{D_{ij}}} + \sum_i \frac{(d_i - d_{i0})^2}{\sigma^2_{d_i}} \quad (12)$$

For each parameter, the prior term is the square of the deviation from the initial value normalized with the variance. The total error function allows the parameters to be steered based on the certainty of that parameter estimate. We minimize $\epsilon_{XY,tot}$ given by

$$\epsilon_{XY,tot} = (1 - \lambda)\epsilon_{XY} + \lambda\epsilon_{XY,prior} \quad (13)$$

Note that maximizing $\exp(-\epsilon_{XY,tot})$ can be considered as a maximum a posteriori probability (MAP) approach. By using the prior term, we could save 2-3% of the runs that otherwise drifted away from the true pose. However, the parameter λ must be set low ($\sim$0.01) not to prevent the search from reaching the true pose.

The output from the alignment in the ground plane is a refined estimate of the XY position and the heading. We also get a scale estimate but we found that the scale could be estimated more accurately in the XZ-search.

Alignment in XZ-plane. Consider a cross-section M_{y_i} (= Z - h_{m,y_i}) of the map for a fixed value y_i along the Y-grid, where Z is the camera altitude. We assume that the global pitch angle error is $\Delta\theta$ and the relative scale error (stereo baseline error) is s. For small errors, the approximate linear transform between the map M_{y_i} and the height patch P_{y_i} (= Z_{est} - h_{p,y_i}) in the XZ-plane is obtained by Taylor expansion as

$$M_{y_i} \begin{pmatrix} x \\ z \end{pmatrix} \approx P_{y_i} \left(\begin{bmatrix} 1+s & \Delta\theta \\ -\Delta\theta & 1+s \end{bmatrix} \begin{bmatrix} x \\ z \end{bmatrix} + \begin{bmatrix} d_x \\ d_z \end{bmatrix} \right) = P_{y_i} \begin{pmatrix} x_p \\ z_p \end{pmatrix} \quad (14)$$

The equation also considers the error in absolute altitude Z and the implicit drift in X due to the erroneous assumption on the pitch angle by introducing the unknown translations d_x and d_z. Next we sample each cross-section M_{y_i} and P_{y_i} bilinearly over a grid in the XZ-plane. In order to compensate for roll errors, we sum over a stripe with the central NY_{width} (=100) cross-sections and denote the sums $M = \Sigma M_{y_i}$ and $P = \Sigma P_{y_i}$. [1] We minimize the error function

$$\epsilon_{XZ} = \int_W \left[P \begin{pmatrix} x_p \\ z_p \end{pmatrix} - M \begin{pmatrix} x \\ z \end{pmatrix} \right]^2 \omega(\mathbf{x})\, d\mathbf{x} \quad (15)$$

where the integration domain W is the total XZ-grid. We expand P and truncate to the linear term. We differentiate ϵ_{XZ} with respect to the unknowns and set the derivatives to zero. The general entities in (**??**) for the XZ-plane read

$$\mathbf{U_{XZ}} = \begin{bmatrix} xP_x + zP_z \\ zP_x - xP_z \\ P_x \\ P_z \end{bmatrix} \begin{bmatrix} xP_x + zP_z \\ zP_x - xP_z \\ P_x \\ P_z \end{bmatrix}^T \quad (16)$$

$$\mathbf{v_{XZ}} = \begin{bmatrix} s & \Delta\theta & d_x & d_z \end{bmatrix}^T \quad (17)$$

$$\mathbf{w_{XZ}} = \begin{pmatrix} M - P \end{pmatrix} \begin{bmatrix} xP_x + zP_z \\ zP_x - xP_z \\ P_x \\ P_z \end{bmatrix} \quad (18)$$

We solve the equation system iteratively and use the estimated parameter vector in (**??**) to transform each cross-section P_{y_i} of the height patch and denote this 3D matrix $P_{mat}(x, y, z)$.

Alignment in YZ-plane. The alignment in the YZ-plane is very similar to the alignment in the XZ-plane. What remains to be estimated in the YZ-plane is the roll angle and the translation drift caused by a roll angle error $\Delta\phi$. Consider a cross-section M_{x_i} of the map for a fixed value x_i along the X-grid. This cross-section is perceived as P_{x_i} if the assumption on roll angle is erroneous. Assuming small errors, an approximate transform between the map M_{x_i} and the height patch P_{x_i} is obtained by Taylor expansion as

$$M_{x_i} \begin{pmatrix} y \\ z \end{pmatrix} \approx P_{x_i} \left(\begin{bmatrix} 1 & -\Delta\phi \\ \Delta\phi & 1 \end{bmatrix} \begin{bmatrix} y \\ z \end{bmatrix} + \begin{bmatrix} d_y \\ d_z \end{bmatrix} \right) = P_{x_i} \begin{pmatrix} y_p \\ z_p \end{pmatrix} \quad (19)$$

[1] By doing so we assume that the non-overlapping areas are not correlated.

The equation also considers the implicit drift in Y due to the erroneous roll angle. As before we sample each cross-section bilinearly over the YZ-grid. We extract the central NX_{width} (=100) cross-sections from the map $P_{mat}(x, y, z)$ and define the sums $M = \Sigma M_{x_i}$ and $P = \Sigma P_{x_i}$. We minimize the error function

$$\epsilon_{YZ} = \int_W \left[P \begin{pmatrix} y_p \\ z_p \end{pmatrix} - M \begin{pmatrix} y \\ z \end{pmatrix} \right]^2 \omega(\mathbf{x})\, d\mathbf{x} \qquad (20)$$

where the integration domain W is the whole YZ-grid. Assuming small angles, making a Taylor expansion to the linear term and differentiating with respect to the unknown parameters yields the following entities in (**??**) for the YZ-plane:

$$\mathbf{U_{YZ}} = \begin{bmatrix} yP_z - zP_y \\ P_y \\ P_z \end{bmatrix} \begin{bmatrix} yP_z - zP_y \\ P_y \\ P_z \end{bmatrix}^T \qquad (21)$$

$$\mathbf{v_{YZ}} = \begin{bmatrix} \Delta\phi & d_y & d_z \end{bmatrix}^T \qquad (22)$$

$$\mathbf{w_{YZ}} = \left(M - P\right) \begin{bmatrix} yP_z - zP_y \\ P_y \\ P_z \end{bmatrix} \qquad (23)$$

We use the estimated parameter vector to transform each cross-section P_{x_i} of the height patch. We simply take the maximum value along the Z-direction for each point on the XY-grid to obtain the 3D transformed height patch $h_{p,est}$.

Alignment in XY-plane. Finally we run the transformed height patch $h_{p,est}$ through XY alignment with the map at fine scale, yielding $h_{p,xyz}$, to refine the estimates on the X and Y positions and the heading ψ.

Scales and iterations. The scales and number of max iterations used in the alignment for each plane are listed in table **??**. A student's t-test at 95% significance level was used to deduce that further alignment over the other two planes (XZ and YZ) did not improve the pose estimates.

Table 1. Scales and max iterations in each plane.

Plane	XY	XZ	YZ	XY
Scales	6→1	6→1	4→1	3→1
Max iterations	10	10	8	6

Matching error. To indicate how well the transformed height patch matches the map, we integrate over the whole patch area to compute the matching error

$$\epsilon_{XYZ} = \int_W \left[h_{p,xyz}(\mathbf{x}) - h_m(\mathbf{x}) \right]^2 \omega(\mathbf{x})\, d\mathbf{x} \qquad (24)$$

Note that the matching error scales with the absolute height of the structure.

4. Evaluation

The global pose estimation method has been evaluated using images from helicopter and aircraft flights. The main purpose of both flights was to scan an urban area and to build a textured 3D model. The ground truth for the camera pose used in the evaluation is the output from the bundle adjustment step when building the 3D models.

A sequence of 51 helicopter images (50 consecutive image pairs) when flying over a residential area has been used in the evaluation. The images were taken in March when there was still some snow on the ground and there were no leaves on the trees. The aircraft images also consist of in total 50 image pairs. They are taken from three image sequences, one from a downtown area and two from residential areas. The images were captured in late summer when the trees were full of leaves. Flight and camera parameters are listed in table **??**. The 3D model used for alignment in the evaluation was captured in summer time and had a resolution of 25 cm on the ground.

4.1. Translation vector direction estimate

Assuming that the relative rotation between images is obtained from the IMU, the translation vector direction was estimated from corresponding image points as detailed in section **??**. For 75% of the image pairs, the error was less than 3 mrad and the worst case was around 8 mrad (0.5°). In the downtown area all errors were less than 2 mrad. The error in the direction estimate is mainly in the vertical component of the translation vector which is reasonable as the motion between images is mainly in the ground plane.

4.2. Pose estimate

To evaluate the pose estimation method, we added errors to the ground truth pose as an initialization and measured how well these errors could be estimated by our method. By adding errors to the ground truth we simulate that we have lost track of the absolute pose and due to drift in the onboard IMU we only have an approximate pose estimate, the error being dependent on the time since the last good pose estimate and the type of IMU. As an example from the evaluation, we use an image pair from the helicopter flight. The local height patch computed with MS and the corresponding area from the 3D model are shown in fig. **??**. The

Table 2. Parameters for aircraft and helicopter images.

Parameter	Aircraft images	Helicopter images
Altitude	600 m	200 m
Field of View	38x26°	45x64°
Image size, pixels	5616x3744	3248x4872
Optical axis	Vertically down	20° left
Resolution on ground	12 cm	6 cm
Size, height patch	∼1250x800	∼650x800

Figure 2. Local height patch from an image pair (left). Height map from model (middle). Convergence zone for starting position of point in (left) is marked in map. Local height patch after transformation with estimated pose (right).

query patch is brighter (larger height values) to the left due to an erroneous initial roll estimate. It is rotated clockwise due to an error on the yaw estimate and a too long stereo base assumption makes the houses too large. After pose estimation, the local height patch is transformed according to the estimated parameters and it is now well aligned with the 3D model.

Robustness. An essential property of the pose estimation method is that it is robust and converges for a wide range of initializations. We compare our method with Mastin *et al*. [**?**] since we have found no other method with as high registration rates as theirs. We used the same range of initialization errors as they did, shown in table **??**. We picked independent errors from rectangular distributions for the different pose parameters. Note that our vehicle yaw (ψ) corresponds to their camera roll (γ). Their FOV error 0.5° corresponds to a scale error around 0.3%. We used $\pm 2\%$ to add some margin for uncertainty in the vehicle velocity affecting the stereo baseline and thus the scale.

For each image pair, we randomly selected five initializations and ran our pose estimation method. We thus have data for 250 initializations for helicopter and aircraft image pairs respectively. First, we determine the rate of correct registrations, *i.e*. if the pose estimate is close to the ground truth. We do this by analyzing the pose estimation errors and, as in [**?**], by visual inspection, *i.e*. if the 3D model and the transformed local height patch after the pose estimation are well aligned. It is very evident if the search method diverges and is trapped in a local minimum.

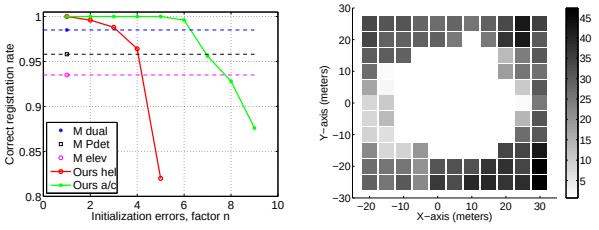
Table 3. Nominal errors used for initializations in evaluation.

Parameter	X	Y	Z	scale
Errors	$\pm$10m	$\pm$10m	$\pm$10m	$\pm 2\%$
Parameter	ψ	θ	ϕ	
Errors	$\pm 2.5°$	$\pm 0.25°$	$\pm 0.25°$	

Since all 500 image pairs were well registered, we increased the range of all orientation errors (yaw, pitch, roll) simultaneously by a factor n, (n = 2..9) compared to table **??**. The position errors and the scale error had the same range as before. For each n we evaluated the correct registration rate for 250 image pair initializations each for helicopter and aircraft images. The results are shown in fig. **??**(a).

For aircraft images, the correct registration rate is higher than 99% for $n \leq 6$, which means a 6-fold improvement compared to [**?**]. Note that the images for both methods have roughly the same FOV and are captured at similar altitudes. For helicopter images, we have a 3-fold improvement compared to [**?**], with a correct registration rate higher than 98.5% for $n \leq 3$. The helicopter images are taken at considerably lower altitude and the footprint on the ground for the local height patch is roughly 1/2 compared to the aircraft images.

Next, we evaluate the robustness of the pose estimation method for initialization errors in the XY position. The ini-



(a) (b)

Figure 3. (a) Correct registration rate vs initialization errors. Results for n=1 for three methods from Mastin *et al*. and for our method with helicopter and aircraft images. (b) XY position estimate error vs XY starting position around ground truth. The central white zone is the convergence zone. The bar indicates XY position error in meters.

tial orientation errors are picked from ranges with a factor of n=2 compared to table **??**. We then varied the starting position in XY within $\pm$25 meters around the ground truth over a 5 m grid. For each XY starting position we made 9 runs, randomly selecting the other parameters, and recorded the median XY position estimate error. Fig. **??**(b) shows that the pose estimation method converges towards the correct solution within an area that is roughly 30x30 meters wide (white central area). For starting positions further out, the method is trapped in a local minimum and results in a large XY position error. [2] The convergence zone has been marked in fig. **??** which indicates that the size of the convergence zone is scene dependent. The size agrees well with the distance between the houses in the height map.

There is a strong correlation between the matching error (**??**) and the XY position estimate error. The correlation coefficient computed over the data points in fig. **??**(b) is ρ=0.87, indicating that the matching error can be used as an indicator whether a good pose estimate has been obtained.

Accuracy. It is essential to have a robust pose estimation method but for navigation purposes it is of utmost importance also to have accurate unbiased estimates with low variance. As a measure of the accuracy, we compute the mean and standard deviation for all pose estimates where the registrations were classified as correct. We show results for three cases; when starting in the ground truth pose, for initialization errors as per table **??** and for n=6, the largest factor when the correct registration rate was still above 98.5%. The results when starting at the ground truth give an indication of the pose estimation errors induced by the stereo method to compute the local height patch. Results are shown for aircraft images from the downtown area, fig. **??**. Similar results were obtained for aircraft and helicopter images from the residential areas. Results are presented as the mean error for each parameter and the error bars indicate $\pm 1\sigma$ values.

When starting at the ground truth, the pose estimation orientation errors are a few mrad, fig. **??**(a). Note that a 2.5 mrad pitch error corresponds to a 1.5 m error in x (vehicle forward direction) at 600 m altitude. Similarly, a 1.5 mrad roll error corresponds to a 0.9 m error in y. The small orientation errors are thus compensated for by translational XY errors to give a good registration of the local height patch with the 3D model. The remaining X and Y errors are within the XY resolution of the 3D model used. The yaw estimate error is induced by the house boundaries being less distinct in the local height patch compared with the 3D model. As can be seen in fig. **??**, the houses in the local height patch are also slightly over-sized due to stereo shadowing effects at remote house boundaries. This will give a small bias in the scale estimate and will also affect the Z estimate. A 0.19% scale error agrees well with a 1.2 m error in Z at 600 m. Adding initialization errors as per table **??** and using the translation vector estimates, the yaw and pitch errors increase slightly, fig. **??**(b). As the error on the translation vector is mainly in the XZ plane, it is natural that the errors increase in X, Z and pitch. When increasing the initial orientation errors to a factor n=6, there is no significant change in the pose estimate error, fig. **??**(c). This means *e.g.* that a yaw initialization error with σ = 150 mrad has been reduced to an error with σ = 5 mrad by our method.

Sensor system. We consider our visual method and the IMU to constitute an onboard sensor system for pose estimation. The filtered pose and speed estimates will be used to initialize the visual method with the global pose and stereo baseline length (scale). The visual method needs to support the IMU with global pose estimates frequently enough to prevent IMU drift to give initializations outside the convergence zone of the visual method. The support frequency required depends on the type of IMU and the scene which affects the size of the convergence zone for the visual method.

A limited amount of outliers in the stereo method will degrade the pose estimate marginally in our dense matching method. A large amount of stereo outliers is likely to result in a poor pose estimate, but this will be indicated by a large matching error. The sensor fusion process taking the individual pose estimates and their corresponding matching errors as inputs, to qualitatively evaluate how stereo outliers affect the filtered pose relative to the convergence zone, is still to be developed.

Since our visual pose estimation method is based on alignment with a global reference, pose estimation errors will not accumulate along the flight path as in SLAM.

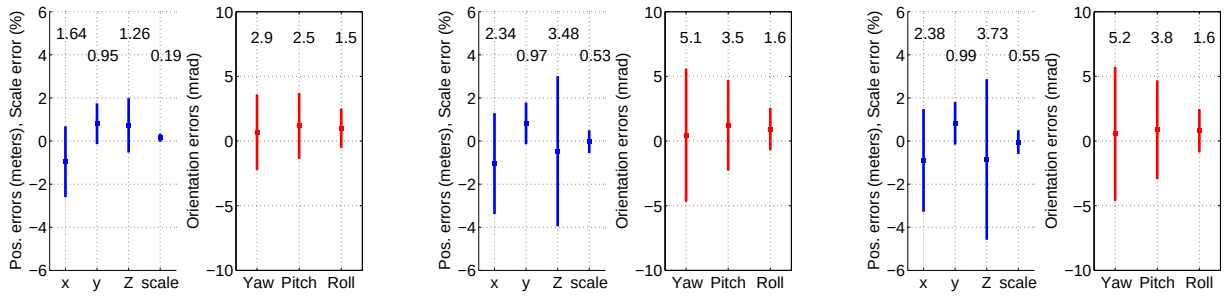
Processing time. The processing time for generation of the local height patch for 16 Mpixel images with a C implementation of the stereo algorithm is around 1.5 s. The pose estimation method is a non-optimized Matlab implementation and takes as an average 13 s for the helicopter images and 25 s for the aircraft images. All runs were made on a standard PC, an Intel Xeon CPU x5690 @ 3.47 GHz.

Our processing times compare well with Mastin *et al.* who report registration times around 13 s for their dual method when implemented on a graphics card.

5. Concluding Remarks and Future Work

A method for online global pose estimation of aerial images by alignment with a georeferenced 3D model is presented. Motion stereo is used to reconstruct a dense local height patch from an image pair. The global pose is inferred from the 3D transform between the local height patch and the model. For efficiency, the sought 3D similarity transform is found by least-squares minimizations of three 2D subproblems. The method does not require any reference

[2] The original 3D transformation problem would also be trapped in a local minimum in these cases.



(a) (b) (c)

Figure 4. Estimated pose errors averaged over image sequence for aircraft images taken at 600 m altitude. Bars indicate $\pm 1\sigma$. Initialization errors: (a) no errors, (b) n=1, (c) n=6.

points in the 3D model, but an approximate initialization of the global pose, in our case provided by an onboard IMU, is assumed. Real aerial images from helicopter and aircraft flights are used to evaluate the method. The results show that the accuracy of the position and orientation estimates is significantly improved compared to the initialization and our method is more robust than competing methods on similar datasets. The proposed matching error computed between the transformed patch and the map clearly indicates whether a reliable pose estimate has been obtained.

Experience from runs in different urban areas motivates future work on the matching error. Especially if there are repetitive height structures in the environment, local minima may give low matching errors as well. The absolute values for the matching error for a good estimate may be substantially higher in an area where there are mainly taller buildings. The matching error must thus be related to the local height structure in the map. Depending on the season, vegetation may look very different in a height map. Pose estimates may improve if vegetation areas are segmented in the map and given low weight in the alignment process.

Acknowledgements. This work was funded by the Swedish Governmental Agency for Innovation Systems, VINNOVA, under contract NFFP5 2010-01249, and supported by the Swedish Foundation for Strategic Research through grant RIT10-0047 (CUAS).