Optimal View Path Planning for Visual SLAM

Sebastian Haner and Anders Heyden

Centre for Mathematical Sciences
Lund University
{haner,heyden}@maths.lth.se
http://www.maths.lth.se

Abstract. In experimental design and 3D reconstruction it is desirable to minimize the number of observations required to reach a prescribed estimation accuracy. Many approaches in the literature attempt to find the next best view from which to measure, and iterate this procedure. This paper discusses a continuous optimization method for finding a whole set of future imaging locations which minimize the reconstruction error of observed geometry along with the distance traveled by the camera between these locations. A computationally efficient iterative algorithm targeted toward application within real-time SLAM systems is presented and tested on simulated data.

Keywords: Next best view planning, path optimization, SLAM

1 Introduction

Visual simultaneous localization and mapping (SLAM) is the task of determining the position and orientation of a camera while concurrently building a map of the environment, using the camera images and possibly other sensors as input. It is a chicken-and-egg type problem; given the map, localization is relatively easy and given the camera positions, map triangulation is straightforward. Accomplishing both at once is at the heart of the SLAM problem, which has received a lot of attention in both the robotics and vision research communities. Much effort is spent improving the robustness and accuracy of algorithms, particularly with respect to error accumulation, drift and loop closing (see e.g. [1,2]). A less studied problem is how to make efficient use of the information collected in active SLAM systems, i.e. systems where the motion of the sensor can be controlled. This article considers the problem of maximizing the useful information gained from a fixed number of images by active planning of the vision sensor movement. Specifically, we consider the task of finding a camera trajectory between two predetermined locations such that the reconstruction accuracy of observed geometry is maximized while the path length is minimized. The envisioned application is robot path planning, where the accuracy usually is a secondary objective, so the focus is on providing the best reconstruction given time or distance constraints.

In this work we only consider the geometric aspects of the problem and do not account for availability of texture or object occlusion, which are of course issues in a real system relying on feature tracking. We further assume the following:

- An initial maximum likelihood estimate of the structure is available, based on observations up to that point.
- All cameras along the trajectory are oriented towards a particular point of interest, e.g. the centroid of the features to be estimated.
- The camera can be positioned with such relative accuracy that its pose and location is fully known at each observation.

These assumptions may be relaxed, as discussed in section 6.2. Finally, the robot path is represented by a sequence of camera locations, and the number of cameras on the path must be chosen in advance.

As an experimental design problem, so-called 'camera network design' has been studied extensively in the photogrammetry literature. The emphasis is on obtaining the most accurate reconstruction given a limited number of cameras, and time can be spent finding an optimal configuration. For example, in [14] a genetic optimization algorithm is used to search the high-dimensional parameter space of camera placements. Similar stochastic algorithms are usually employed since the problem is intrinsically multi-modal i.e. the objective function has many local minima, cf. [3]. In the context of 3D reconstruction in controlled environments, the task at hand is usually referred to as 'next best view planning', suggesting that given an approximate reconstruction we seek a single next view that will reduce the error the most. This is the case in [4] where the authors reconstruct objects using a camera mounted on a robotic arm. The object geometry is estimated using a Kalman filter, and the next imaging location is determined by searching a discrete parameter space and evaluating the expected information gain in the filter at each position. A different approach is taken in [5] where the next imaging location is decided based only on the single currently least well-determined feature, allowing a simple closed form solution. In the above problem formulations there are usually few or no constraints imposed on possible sensor configurations, computational complexity is less of an issue and the 'next best view' approaches do not consider more than one future observation. This work will show that given constraints on the camera positions, good solutions for many future observations can be found relatively quickly. For a recent general survey of the sensor planning field see the book by Chen et al. [6].

The work most similar in spirit to ours is [7] where the path of a robot moving in the plane is planned based on the expected reconstruction accuracy of an observed object. An approximation of the geometry is given and the expected information gain from observing the object from a particular vantage point is determined on a discrete grid of camera locations. Each grid cell is assigned a cost proportional to the inverse of the information gain, and a minimum cost path is found between the starting point and the global minimum grid cell. The algorithm does not take into account the new information gained after an actual observation is made, however, and becomes computationally expensive if we allow the camera to move in three dimensions. The minimum cost path formulation also restricts the choice of cost function. This work proposes an efficient continuous optimization approach to the problem of finding a short path with large information gain.

2 Problem Formulation

The planner takes as input an initial estimate of the structure, the current location of the sensor and the desired destination. The output is a path, represented by a discrete set of sensor locations, connecting these points. The number of locations on the path can be set explicitly or deduced from e.g. the robot's speed and sample rate and the distance to be travelled. For the experiments in this paper the sensor is assumed to be a single fully calibrated camera, although extension to stereo and multi-camera systems is straightforward. The standard pinhole camera model is used, so that the relation $\hat{x} = f(P, X)$ between a world point X and its projection $\hat{x}$ in homogeneous coordinates is given by

$$\lambda f(P, X) = KM \begin{pmatrix} X \\ 1 \end{pmatrix} = \begin{pmatrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} R \mid -Rt \end{pmatrix} \begin{pmatrix} X \\ 1 \end{pmatrix} \tag{1}$$

where R and t are the camera rotation and translation and K represents the known intrinsic calibration parameters. However, any differentiable projection function $f(P, X)$ may be substituted, e.g. to include radial distortion terms.

In the interest of reducing the parameter space dimension, each camera is parametrized only by its position and is automatically oriented toward a point of interest, typically chosen as the centroid of the structure under consideration. Features are deemed visible if they fall within the camera's field of view; possible occlusion by other objects is not considered. The measurement uncertainty of features is also considered fixed.

We define the optimization problem as follows:

Problem 1. Minimize the reconstruction uncertainty of observed geometry and the distance traveled by the sensor between imaging locations.

These are conflicting objectives, which are combined in a cost function defined below.

3 Cost Function

Lacking ground truth data or other *a priori* information, the quality of a reconstruction can only be judged by the statistical uncertainty of the estimate. Condensing a probability distribution into a scalar quality measure is not entirely straight-forward, however, and choices must be made depending on the intended application. Also, in most situations only estimates of the probability distribution are available, e.g. the mean and covariance. In the experimental design literature, many summary statistics have been proposed and are usually functions of the eigenvalues of the covariance matrix, e.g. the trace and determinant, cf. [8]. In the structure-from-motion problem, the eigenvalues have a direct geometric interpretation which we consider below.

If we assume the position and orientation of the camera is fully known when an observation is made, the structure estimates corresponding to individual features are independent of each other, and the covariance matrix is block diagonal with 3-by-3 blocks (assuming point features). The eigenvalues of each block correspond to the semi-axes of the ellipsoid representing the variance of the feature location. We would like these ellipsoids to be as small as possible, but in what sense? If we minimize the volume, i.e. the determinant, we admit solutions where a point may be very well-determined in two directions but with a large uncertainty in the third (typically the depth). Minimizing the determinant of the entire covariance matrix (the so-called D-optimality criterion) could favor solutions where one point is very well determined while others are much less certain. For navigation and mapping purposes, we would like all, or at least the majority of features to be reconstructed to reasonable accuracy. Minimizing the largest eigenvalue (E-optimality) would achieve this, but results in a non-smooth objective function. We choose to minimize the sum of the eigenvalues (A-optimality), i.e. the trace of the covariance matrix, which provides a good trade-off with the added computational benefit of not having to calculate individual eigenvalues.

Before introducing the cost function, we discuss how to compute the trace given a set of measurements.

3.1 Calculating Covariance

In many recent SLAM systems (e.g. [9,10,11]) maximum likelihood estimates obtained via bundle adjustment are available. We assume the structure estimate is optimal in the ML sense with respect to the observations; then the information matrix is given to first order by $I = J^\top R^{-1} J$ where J is the Jacobian of the reprojection error evaluated at the minimum, and R the measurement noise covariance [12]. Also, the (pseudo-)inverse of I gives an approximation of the covariance matrix. Since information is additive, including new observations in the estimate amounts to summing the individual information matrices. In other words, to calculate the effect of new observations on the structure estimate, we compute the Jacobian of each observation and add the corresponding information matrices to the initial one. New observations may of course shift the ML estimate, invalidating the approximation, but this is avoided in a natural way as discussed in section 4.

Given a world point X and a camera P, let x be the measured image coordinate, and $f(P, X)$ the projection function mapping X to the expected image coordinate $\hat{x}$. Define the re-projection error as $E_X(P, X, x) = f(P, X) - x$ with Jacobian

$$J_X = \frac{dE_X}{dX} = \begin{pmatrix} \frac{\partial f_1}{\partial X_1} & \frac{\partial f_1}{\partial X_2} & \frac{\partial f_1}{\partial X_3} \\ \frac{\partial f_2}{\partial X_1} & \frac{\partial f_2}{\partial X_2} & \frac{\partial f_2}{\partial X_3} \end{pmatrix} . \tag{2}$$

If several points $X^{1,\dots,N}$ are observed simultaneously, let

$$E(P, X^{1:N}, x^{1:N}) = \begin{pmatrix} E_{X^1} \\ \vdots \\ E_{X^N} \end{pmatrix} \tag{3}$$

with block diagonal Jacobian

$$J = \begin{pmatrix} J_{X^1} & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & J_{X^N} \end{pmatrix} . \tag{4}$$

The information matrix for a single image is then given by

$$I(P, X^{1:N}) = \begin{pmatrix} J_{X^1}^\top R_1^{-1} J_{X^1} & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & J_{X^N}^\top R_N^{-1} J_{X^N} \end{pmatrix} \tag{5}$$

where usually the $R_i = \left(\begin{smallmatrix} \sigma^2 & 0 \\ 0 & \sigma^2 \end{smallmatrix}\right)$.

The final information matrix given the initial information I_0 and images from camera positions $P^{1,\dots,M}$ is now

$$I_M = I_0 + \sum_{j=1}^{M} I(P^j, X^{1:N}) . \tag{6}$$

Note that the computation is linear in the number of observed features and the number of images, and that the covariance of the estimate is the inverse, $\Sigma_{P^{1:M}, X^{1:N}} = I_M^{-1}$. For notational convenience, from hereon let P denote the set $P^{1:M}$ of camera poses along a path, and $X = X^{1:N}$ the estimated structure.

3.2 Cost Function

We propose the following cost function:

$$\begin{aligned} C(P, X) &= \frac{1}{N}\mathrm{tr}(\Sigma_{P,X}) + \frac{\alpha}{(M-1)^{1-q}} \sum_{j=1}^{M-1} \|P_{\mathrm{pos}}^{j+1} - P_{\mathrm{pos}}^{j}\|^q \\ &= U(P, X) + \alpha D(P) , \end{aligned} \tag{7}$$

i.e. the uncertainty measure plus a function of the camera path, weighted by a constant factor $\alpha > 0$, where $q \geq 1$. The normalization constants N^{-1} and $(M-1)^{q-1}$ are designed to make the cost approximately invariant with respect to the number of observed features and camera positions on the path. Note that by choosing $q > 1$, $D(P)$ will favor solutions with equidistant spacing between the camera positions, and introducing an offset d, $D(P) = \sum_{j=1}^{M-1} (\|P_{\mathrm{pos}}^{j+1} - P_{\mathrm{pos}}^{j}\| - d)^q$, we can impose the soft constraint that the path length be $d(M-1)$, if desired.

3.3 Cost function properties

The multi-modality of the objective functions normally used in next best view planning makes optimization difficult. The proposed cost function is no exception, but due to the somewhat local nature of the sought solution there are obvious bounds on the cost and geometry of the path.

Proposition 1. *$U(P^{1:M}, X)$ is a non-negative decreasing function of the number of observations M.*

Proof. The information matrix I is positive semidefinite. Including a new observation amounts to adding another positive semidefinite matrix $\varDelta I$ to I, and the result is again positive semidefinite. By the Courant-Fischer theorem, we know that the (sorted) eigenvalues satisfy $\lambda_i(I + \varDelta I) \geq \lambda_i(I)$ for all $i = 1, \ldots, n$ and equivalently $\lambda_i(\varSigma_{\text{updated}}) = \lambda_i\big((I + \varDelta I)^+\big) \leq \lambda_i(I^+) = \lambda_i(\varSigma_{\text{initial}})$. Evidently $\operatorname{tr}(\varSigma_{\text{updated}}) \leq \operatorname{tr}(\varSigma_{\text{initial}})$. □

Theorem 1. *The length of the path at the minimum P^* is bounded.*

Proof. Given any initial estimate $\hat{P}$ of the path, we have

$$\begin{aligned} \alpha D(P^*) &\leq U(\hat{P}, X) + \alpha D(\hat{P}) - U(P^*, X) \\ &\leq U(\hat{P}, X) + \alpha D(\hat{P}) \\ &\leq U_{\text{initial}} + \alpha D(\hat{P}) \end{aligned} \tag{8}$$

where $U_{\text{initial}} = \frac{1}{N}\operatorname{tr}(\varSigma_0)$ and $\varSigma_0$ the covariance of the current structure estimate. Since $\|P_{\text{pos}}^{j+1} - P_{\text{pos}}^{j}\| < \|P_{\text{pos}}^{j+1} - P_{\text{pos}}^{j}\|^q + 1$, the length of P^* is bounded from above by $(M-1)^{1-q}\big(\alpha^{-1}U_{\text{initial}} + D(\hat{P})\big) + M - 1$. □

We see that the path must be contained inside an ellipsoid with foci at the (fixed) first and last camera positions, and that the bound can be computed easily in advance. As expected, the optimal path approaches the line segment between the foci as α grows.

This result suggests that we may attempt to find and compare several local minima by optimizing with varying initial paths sampled from within the feasible ellipsoid.

4 Proposed Algorithm

As noted in the introduction, the next best view problem is known to suffer from multiple local minima, cf. [3]; this is true for all reasonable choices of U. Finding the global minimum is a difficult problem, and the prevailing approach in the literature seems to be more or less exhaustive search over a discretized parameter space, [4,7], or stochastic optimization methods, [13,14]. In the interest of speed, however, we adopt a gradient based optimization scheme, using the well-known

Levenberg-Marquardt (LM) method. LM minimizes the 2-norm of a residual vector r, which we construct as

$$r = \left(\frac{\mathrm{tr}(\Sigma_{P,X^1})}{N}, \ldots, \frac{\mathrm{tr}(\Sigma_{P,X^N})}{N}, \frac{\alpha \| P^2_{\mathrm{pos}} - P^1_{\mathrm{pos}} \|^q}{(M-1)^{1-q}}, \ldots, \frac{\alpha \| P^M_{\mathrm{pos}} - P^{M-1}_{\mathrm{pos}} \|^q}{(M-1)^{1-q}} \right)^{\frac{1}{2}}$$

(the exponent indicates element-wise square root) so that $\|r\|^2 = C(P, X)$. The parameter space is the $M - 2$ intermediate camera positions; the camera orientation is determined by its position and the interest point.

The final hurdle is how to evaluate the cost function *before* any observations are made. The best we can do is predict what the camera will see at a particular location given the current best estimate of the structure. Assuming that measurements are corrupted with zero-mean noise, the expected observation is simply the projection $\hat{x} = f(P_i, X)$. Such an observation has zero reprojection error, and so does not affect the ML estimate.

The optimization is applied within the following framework:

1. Given an initial estimate of the structure, calculate its centroid and let this be the camera's point of interest. Select a target location for the camera, i.e. select the end point of the path.
2. Generate an initial path by linear interpolation between the first and last camera locations. The number of discrete camera locations along the path could be selected to match the image sampling rate and speed of the robot, but this would normally result in far too many locations and a very high-dimensional search space. However, it stands to reason that more images taken from approximately the same vantage point do not contribute qualitatively to the reconstruction, so a relatively sparse distribution of camera locations is sufficient.
3. Find a minimum of the cost function wrt. P using the LM algorithm.
4. Move the camera to the next location along the path and make an actual observation. Update the structure estimate with this new information, and update the camera interest point location and path end point, if needed.

Repeat steps 3 and 4, each time with one less camera location along the path and using the previous path estimate as an initial guess.

5 Experiments

We first apply the above algorithm to the scenario of a robot trying to pass through a doorway. The doorway is represented by a rectangular array of point features which are optimally triangulated from the first two views, see figure 1(a). In all experiments we assume an image measurement noise σ equivalent to about one pixel. The target location is placed in front of the doorway, and the path is discretized with four waypoints in between. The optimization is run until convergence and the robot is moved to the next prescribed location along the path, where a new image is acquired and the structure estimate is updated using bundle adjustment.

The influence of the parameter α is illustrated in figure 2 and table 1. The robot passes by a point cloud, and to get a closer look it must make a detour. A large α penalizes long paths at the expense of reconstruction accuracy.

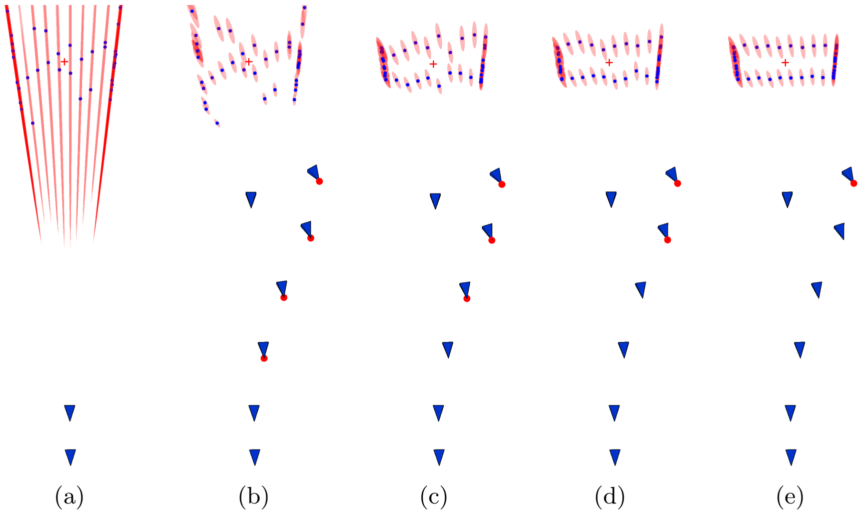

Fig. 1. Doorway scenario. The robot wishes to approach the passage while determining its geometry as accurately as possible. The first two cameras on the path represent the last two images the robot has acquired and provide the initial optimal triangulation of the geometry. Red dots indicate which cameras are free to move, the red cross is the point of interest. In this case subsequent observations do not visibly change the initially planned path. The uncertainty ellipsoids represent 5σ in (a) and 50σ in (b)-(e). Note that in the latter cases the *expected* uncertainties, given all observations along the path, are displayed. The values $q = 3$ and $\alpha = 4.5 \cdot 10^{-7}$ were used.

6 Discussion

6.1 Computational Complexity

As noted in section 3.1, the cost function can be evaluated in $\mathcal{O}(MN)$ time. The LM algorithm requires the computation of the Jacobian of the residual vector r each iteration. The analytic expression may be very complicated and expensive to evaluate, so a finite difference approximation is preferred. The cost function must be differentiated with respect to $3(M-2)$ parameters, requiring $3(M-2)+1$ function evaluations to compute the Jacobian. But the covariance matrix is a function of a sum of individual information matrices, where only one

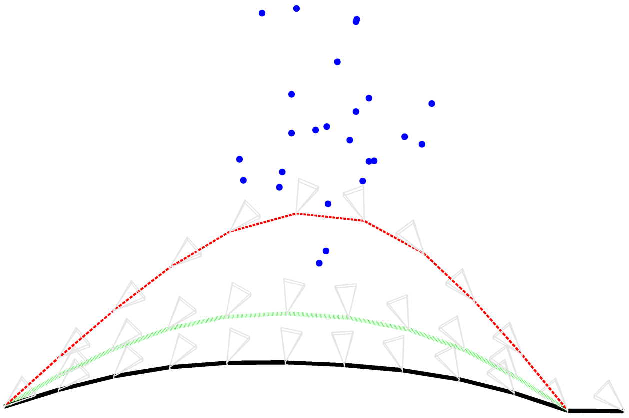

Fig. 2. Here the robot passes (from right to left) by a point cloud and makes a detour to get as close to the features as possible; this is natural, since the closer the feature, the higher its angular resolution. Three cases are plotted: $\alpha = 0.2 \cdot 10^{-7}$ (*red dashed*), $\alpha = 0.5 \cdot 10^{-7}$ (*green dotted*) and $\alpha = 10^{-6}$ (*black*).

	Optimized path		Straight path	
α	Rel. err.	Rec. err.	Rel. err	Rec. err
$1.0 \cdot 10^{-7}$	$1.64 \cdot 10^{-3}$	$8.32 \cdot 10^{-4}$	$2.02 \cdot 10^{-3}$	$1.03 \cdot 10^{-3}$
$0.5 \cdot 10^{-7}$	$1.25 \cdot 10^{-3}$	$7.15 \cdot 10^{-4}$	”	”
$0.2 \cdot 10^{-7}$	$5.36 \cdot 10^{-4}$	$4.53 \cdot 10^{-4}$	”	”

Table 1. Relative error $U(P,X)/U(P^{1:2},X)$ and absolute reconstruction error $\frac{1}{N}\sum_{i=1}^{N}\|X^i - X^i_{\text{true}}\|$, where $X^{1:N}_{\text{true}}$ is the ground truth structure being observed, computed for different values of α in the scenario of figure 2. The relative error represents the expected decrease in uncertainty from the initial estimate given by the first two images, the reconstruction error the actual error after all observations have been made. As α is decreased, the optimized path deviates more from the straight line between the first and last camera position, and the reconstruction error is decreased.

term changes as the camera parameters are perturbed one at a time. By careful bookkeeping of the information matrices only 4 instances need to be computed for each camera instead of all $3(M-2)+1$ of a naïve implementation. This lowers the complexity of computing the Jacobian from $\mathcal{O}(M^2N)$ to $\mathcal{O}(MN)$. Nevertheless, in real-time applications computing the path should take a few seconds at most, and recent SLAM systems track hundreds or thousands of features. It may therefore be necessary to restrict attention to a subset of reconstructed features, e.g. those with the largest uncertainty, when evaluating the cost.

Furthermore, due to the iterative nature of the optimization, the path computation may be aborted before convergence but still yield a good approximation, depending on available time and computational resources.

6.2 Extensions

The assumptions in section 1 can of course be relaxed. If an initial ML structure estimate is not available, we can either choose to ignore any prior information and initialize the algorithm using optimal triangulation from the most recent images, or simply substitute a non-ML estimate (e.g. from an EKF). If the estimate is good enough, the inverse of the covariance matrix will still be a good approximation to the Fisher information. Even if it's a poor approximation we would expect the optimized paths to yield better reconstruction accuracy than a straight or random one.

The requirement that the camera be oriented toward a particular point is only intended to reduce the dimension of the parameter space. Optimization over the orientations, or other rules for selecting orientation based on camera position and estimated structure could easily be incorporated.

It is also assumed that the camera position and orientation are known to high accuracy when acquiring images. Obviously, this is rarely true in a practical SLAM system, where there may be considerable uncertainty in the robot location. However, the location is usually well-determined relative to nearby, recently observed features, so for short-term local path planning this is a fair approximation. Nevertheless, incorporating the camera uncertainty in the covariance estimation would be straightforward, but would also introduce correlations between features. The information and covariance matrices would no longer be block diagonal, raising the computational load considerably, and the cost function would possibly have to be modified to include the camera location uncertainty. The practical gain of incorporating such information is less clear.

The nature of the optimization scheme makes it easy to incorporate different constraints. For example, obstacles in the robot's path can be modeled as a potential field added to the cost function.

7 Conclusion

This paper has presented a continuous optimization approach to certain instances of the next best view planning problem, aimed toward application in

SLAM systems. Unlike previous algorithms the next best view is chosen with consideration of several expected future observations. While the solutions are only locally optimal, experiments show that reconstruction accuracy is still much improved, at a computational cost linear in the number of cameras and features.

References

1. Tom Botterill, Steven Mills, and Richard Green. Bag-of-words-driven, single-camera simultaneous localization and mapping. *Journal of Field Robotics*, 2010.
2. Pedro Piniés, Lina María Paz, Dorian Gálvez-López, and Juan D. Tardós. Ci-graph simultaneous localization and mapping for three-dimensional reconstruction of large and complex environments using a multicamera system. *Journal of Field Robotics*, 27(5):561–586, 2010.
3. C.S. Fraser. Network design considerations for non-topographic photogrammetry. *Photo Eng. and Remote Sensing*, 50(8):1115–1126, 1984.
4. Stefan Wenhardt, Benjamin Deutsch, Joachim Hornegger, Heinrich Niemann, and Joachim Denzler. An information theoretic approach for next best view planning in 3-d reconstruction. In *Proc. International Conference on Pattern Recognition (ICPR'06)*, volume 1, pages 103–106. IEEE Computer Society, 2006.
5. Michael Trummer, Christoph Munkelt, and Joachim Denzler. Online next-best-view planning for accuracy optimization using an extended e-criterion. In *Proc. International Conference on Pattern Recognition (ICPR'10)*, volume 0, pages 1642–1645. IEEE Computer Society, 2010.
6. Shengyong Chen, Y. F. Li, Jianwei Zhang, and Wanliang Wang. *Active Sensor Planning for Multiview Vision Tasks.* Springer Publishing Company, Incorporated, 1st edition, 2008.
7. E. Dunn, J. van den Berg, and J.-M. Frahm. Developing visual sensing strategies through next best view planning. In *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, pages 4001 –4008, oct. 2009.
8. Douglas C. Montgomery. *Design and Analysis of Experiments.* John Wiley & Sons Canada, Ltd., 5th edition, 2000.
9. Georg Klein and David Murray. Parallel tracking and mapping for small AR workspaces. In *Proc. Sixth IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR'07)*, Nara, Japan, November 2007.
10. Hauke Strasdat, J. M. M. Montiel, and Andrew J. Davison. Scale drift-aware large scale monocular slam. In *Proc. Robotics; Science and Systems*, 2010.
11. E. Mouragnon, M. Lhuillier, M. Dhome, F. Dekeyser, and P. Sayd. Generic and real-time structure from motion using local bundle adjustment. *Image and Vision Computing*, 27(8):1178 – 1193, 2009.
12. Richard Hartley and Andrew Zisserman. *Multiple View Geometry.* Cambridge University Press, 2003.
13. S.Y. Chen and Y.F. Li. Automatic sensor placement for model-based robot vision. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 34(1):393 –408, feb. 2004.
14. Enrique Dunn, Gustavo Olague, and Evelyne Lutton. Parisian camera placement for vision metrology. *Pattern Recognition Letters*, 27(11):1209 – 1219, 2006. Evolutionary Computer Vision and Image Understanding.