Efficient Triangulation Based on 3D Euclidean Optimization

Klas Nordberg
Computer Vision Laboratory, Linköping University
klas@isy.liu.se

Abstract

This paper presents a method for triangulation of 3D points given their projections in two images. Recent results show that the triangulation mapping can be represented as a linear operator $\mathbf{K}$ *applied to the outer product of corresponding homogeneous image coordinates, leading to a triangulation of very low computational complexity.* $\mathbf{K}$ *can be determined from the camera matrices, together with a so-called blind plane, but we show here that it can be further refined by a process similar to Gold Standard methods for camera matrix estimation. In particular, it is demonstrated that* $\mathbf{K}$ *can be adjusted to minimize the Euclidean* L_1 *residual 3D error, bringing it down to the same level as the optimal triangulation by Hartley and Sturm. The resulting* $\mathbf{K}$ *optimally fits a set of 2D+2D+3D data where the error is measured in the 3D space. Assuming that this calibration set is representative for a particular application, where later only the 2D points are known, this* $\mathbf{K}$ *can be used for triangulation of 3D points in an optimal way, which in addition is very efficient since the optimization need only be made once for the point set.*

The refinement of $\mathbf{K}$ *is made by iteratively reducing errors in the 3D and 2D domains, respectively. Experiments on real data suggests that very few iterations are needed to accomplish useful results.*

1. Introduction

The following presentation is based on the multi-view pin-hole camera model where the relation between 2D image coordinates and 3D world coordinates can be described according to

$$\mathbf{y}_i \sim \mathbf{C}_i\, \mathbf{x} \tag{1}$$

where $\mathbf{x}$ and $\mathbf{y}_i$ are the homogeneous representation of corresponding 3D and 2D points, and $\mathbf{C}_i$ is the camera matrix of view i. Here, $\sim$ denote equality up to a scalar multiplication.

1.1 Euclidean optimization

Euclidean optimization implies that the error between a point and its reprojection is measured in the Euclidean space, rather than in the projection space which hosts the homogeneous coordinates of 2D or 3D points, and then minimized over some set of parameters. In a typical case, the parameters correspond to a camera matrix $\mathbf{C}$ and the error is defined in the Euclidean 2D image space. The Euclidean optimization problem is then to find $\mathbf{C}$ which minimizes the Euclidean error over some set of corresponding 3D and 2D points. In the case of camera view i, and for a set of corresponding 3D points $\{\mathbf{x}^{(k)}\}$ and 2D points $\{\mathbf{y}_i^{(k)}\}$, both produced with some amount of measurement error, we want to find $\mathbf{C}_i$ which minimizes

$$\epsilon_{\mathrm{2D}} = \sum_k \rho(\mathbf{C}_i\, \mathbf{x}^{(k)}, \mathbf{y}_i^{(k)}) \tag{2}$$

where ρ is a Euclidean error function. In the multi-view case, this problem has to be solved for each of the camera views.

Usually, in this case, ρ is chosen as the squared Euclidean 2D distance between the corresponding image coordinates which implies that ϵ_{2D} is the L_2 norm of the residual errors. A solution to this minimization problem cannot be given in closed form, instead iterative methods have to be used, relying on having a reasonable initial solution. This can be found by solving Equation (1) for the entire set of points using the direct linear transformation method [2] with normalized homogeneous coordinates [3]. Using a suitable non-linear optimizer, for example based on the Levenberg-Marquardt algorithm, ϵ_{2D} can be minimized by adjusting the elements of $\mathbf{C}$ (The Gold Standard algorithm [2]). As a result we obtain a camera matrix $\mathbf{C}$ which can perform the 3D-to-2D mapping with a small error compared to the initial calibration set. Given that this set is representative for some application, this camera mapping produce smaller errors when mapping also other 3D points.

978-1-4244-2175-6/08/$25.00 ©2008 IEEE

1.2 Triangulation

Given a pair of image coordinates in two different camera views which correspond to the same 3D point, triangulation (or 3D reconstruction) is the process of determining the coordinates of the 3D point. Provided that the camera matrices related to each of the two views are known, this problem can be solved using various approaches. Some of the triangulation methods found in the literature include the *mid-point method*, the *homogeneous* or the *inhomogeneous methods*, and the *optimal method*. See [2, 4] for detailed descriptions of these and other methods.

1.3 This paper

Recently, it has been shown that the triangulation process can be described as a linear transformation applied on the outer products of the corresponding homogeneous image coordinates, represented by a matrix $\mathbf{K}$ [5]. This means that the 2D-to-3D mapping can be described analogous to Equation (1), and suggests that Euclidean optimization can be applied to $\mathbf{K}$, similar to how $\mathbf{C}$ is optimized in the Gold Standard Algorithm for camera calibration. This paper shows that such an approach is useful, provided that it is carefully implemented.

2. The triangulation operator

In [5] it is shown that the computation of $\mathbf{x}$ can be made in terms of a linear operator applied to the image coordinates. The main result is that if we consider Equation (1) for two different cameras $\mathbf{C}_1$ and $\mathbf{C}_2$, producing two image coordinates $\mathbf{y}_1$ and $\mathbf{y}_2$ from the 3D point $\mathbf{x}$, it is possible to define a mapping $\mathbf{K}$ such that

$$\mathbf{x} \sim \mathbf{K}\,(\mathbf{y}_1 \otimes \mathbf{y}_2) \qquad \left[x_\alpha \sim K_\alpha^{\beta\gamma}\, y_\beta\, y_\gamma\right] \qquad (3)$$

The 4×9 matrix (or $4 \times 3 \times 3$ tensor) $\mathbf{K}$ is not unique, it is parameterized by a plane $\mathbf{p}$ in the 3D space which must include the camera focal points. Furthermore, if $\mathbf{x}$ lies in the plane $\mathbf{p}$, it follows that $\mathbf{K}\,(\mathbf{y}_1 \otimes \mathbf{y}_2) = \mathbf{0}$ which means that the triangulation operation cannot reconstruct points in this *blind plane*. In many applications this is not a problem since $\mathbf{p}$ can be chosen to lie completely outside the common field of view for the two cameras. Similar to the optimal triangulation proposed in [4], the triangulation mapping defined by $\mathbf{K}$ is invariant to projective transformation of the 3D space.

In [5] it is shown how $\mathbf{K}$ can be constructed from $\mathbf{C}_1$, $\mathbf{C}_2$ and $\mathbf{p}$. However, given the symmetry between Equation (1) and Equation (3) we will here consider an alternative approach. In the case that we have two sets of corresponding image coordinates from two camera views, together with the set of 3D points, we can apply Euclidean optimization in the 3D space.

3. Euclidean optimization in 3D

A practical case for which triangulation can be applied may be described as follows. We have a set of 3D coordinates $\{\mathbf{x}^{(k)}\}$ which is determined to some reasonable degree of accuracy. As will be apparent in the examples shown in Section 4, the accuracy may not be isotropic, e.g., if the points lie on a well-defined surface. Two images of this set are produced from two different view points and the corresponding image points are determined. Again, this can only be done with a certain accuracy. Assuming that lens distortion can be sufficiently compensated, this leaves us with two sets of image points, $\{\mathbf{y}_1^{(k)}\}$ and $\{\mathbf{y}_2^{(k)}\}$, which satisfy the pin-hole camera model, Equation (1), for some camera matrices $\mathbf{C}_1$ and $\mathbf{C}_2$ and for all $k = 1, \ldots, N$.

Given these three sets of calibration points, we can start by first determining $\mathbf{C}_1$ and $\mathbf{C}_2$, e.g., using the Gold Standard algorithm. With $\mathbf{C}_1$ and $\mathbf{C}_2$ at hand, it is also possible to compute both the fundamental matrix $\mathbf{F}$ [2], and the triangulation operation $\mathbf{K}$ [5]. In the latter case, we also need the plane $\mathbf{p}$ which must include the two camera focal points. These are given by $\mathbf{C}_1$ and $\mathbf{C}_2$ and $\mathbf{p}$ can then be determined as being approximately perpendicular to the direction which point to the center of the 3D points. In many applications this should produce a $\mathbf{p}$ which is completely outside the common field of view for the cameras, and therefore not cause any problem when $\mathbf{K}\,(\mathbf{y}_1 \otimes \mathbf{y}_2)$ is close to the zero vector.

The operator $\mathbf{K}$ produced in this way will be able to reconstruct 3D points from image points by means of Equation (3). The 3D reconstruction error, however, may not be optimal simply because we have derived $\mathbf{K}$ from $\mathbf{C}_1$ and $\mathbf{C}_2$ which are optimized in the image domain. The main result of this paper is that this can be dealt with by setting of a Euclidean optimization problem in the 3D space (instead of in the 2D space) and optimize it over the elements of $\mathbf{K}$ (instead of over the elements of $\mathbf{C}$). We form

$$\epsilon_{3D} = \sum_k \rho(\mathbf{K}(\mathbf{y}_1^{(k)} \otimes \mathbf{y}_2^{(k)}), \mathbf{x}^{(k)}) \qquad (4)$$

where ρ now is a Euclidean error function in the 3D space. Notice the symmetry between Equations (2) and (4) and between Equations (1) and (3).

In Euclidean optimization we are free to choose the ρ freely as long as it reflects an error based on Euclidean distances, rather than errors in a projective space. Here,

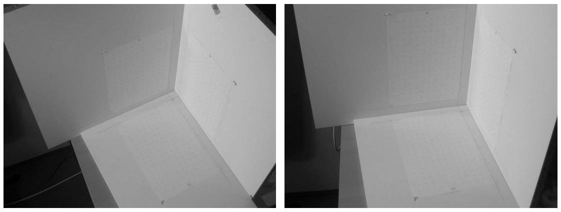

Figure 1. The two camera views.

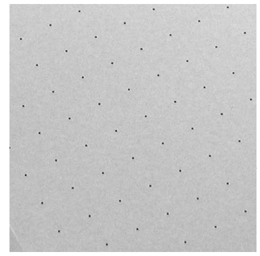

Figure 2. Closeup of part of left image.

we will set ρ as the distance between the corresponding 3D points, which implies that $\epsilon_{3\mathrm{D}}$ is the L_1-norm of the residual errors and that $\epsilon_{3\mathrm{D}}/N$ is the mean error distance between the reconstructed 3D points and the measured 3D point. The squared distance which is used for ρ in the image space is natural when we can assume the errors to be independent and of Gaussian distribution. These assumptions are not valid in general for errors in the 3D space; $\mathbf{K}$ will be based on all 3D and 2D coordinates in the calibration set and we know for sure that the 3D error distribution will be space varying [2].

The result of this process should be an adjusted operator derived from a calibration set of 2D+2D+3D points which very efficiently can triangulate 3D points with a very low L_1 error provided that they lie close to the calibration set. In the following section, this process is evaluated on a set of real data. It is also refined by including errors in the image domain in the optimization.

4. Experiments and refinements

A 3D corner of three perpendicular planes, each with 35 points in a regular grid, is viewed from two distinct points with a 1944×2592 digital camera. The positions of the 105 3D points are measured manually with an accuracy of 0.5 mm. The corresponding image points are determined by means of local minima detection, producing integer valued coordinates. These are affected by estimation noise, which is assumed to be in the order of 1-2 pixels, and geometric noise from the lens distortion. The latter is compensated for by assuming a radial distortion and optimize its parameter to makes lines in the point patters as straight as possible [1]. The result is a set of distortion compensated image points $\{\mathbf{y}_1^{(k)}\}$ and $\{\mathbf{y}_2^{(k)}\}$, corresponding to the set of 3D points $\{\mathbf{x}^{(k)}\}$. Figure 1 shows the two images, one of them close up in Figure 2. $\mathbf{C}_1$ and $\mathbf{C}_2$ are estimated with the Gold Standard algorithm, producing a mean and maximum 2D L_1-residual error at 0.50 and 1.5 pixels, respectively (although they are optimized relative the L_2 error!).

Given the camera matrices, $\mathbf{F}$ and $\mathbf{K}$ are determined according to Section 3, the latter is here denoted $\mathbf{K}_0$. We can also apply standard triangulation methods for reconstructing the 3D points, given their projections in each of the images, and measure the reconstruction L_1-error ϵ, Equation (4). This is done on an *evaluation set* of 3D points (together with their image projections) which lies in between the calibration points. The optimal method (along with the mid-point method) is the best method, with figures reported in Table 1.

Next, triangulation based on $\mathbf{K}_0$ is investigated. The result is presented in the second row of Table 1, which clearly is inferior to the optimal method. However, $\mathbf{K}_0$ is computed from $\mathbf{C}_1$ and $\mathbf{C}_2$ which are optimized in the image domain, not in the 3D space. Since we are interested in 3D errors, we minimize $\epsilon_{3\mathrm{D}}$ over the elements of $\mathbf{K}$, keeping one of them fix since $\mathbf{K}$ is an element of a projective space. Using $\mathbf{K}_0$ as the initial value, the minimization has to be done iteratively, resulting in $\mathbf{K}_1$. The corresponding errors are presented in the third row of Table 1, showing only a minor improvement.

The construction of $\mathbf{K}_1$ does not take into account that there are errors also in the image domain. To obtain a lower error we consider the possibility of reducing errors in both domains. The optimal method does this explicitly by first determining a new pair of image points which satisfy the epipolar constraint

$$\mathbf{y}_1^T\,\mathbf{F}\,\mathbf{y}_2 = \mathbf{F}\cdot(\mathbf{y}_1\otimes\mathbf{y}_2) = 0 \tag{5}$$

and triangulate based on these. Equation (5) can be interpreted as a condition on $\mathbf{Y} = (\mathbf{y}_1 \otimes \mathbf{y}_2)$; it must be perpendicular to $\mathbf{F}$. An alternative approach, therefore, is to adjust $\mathbf{Y}$ to make it perpendicular to $\mathbf{F}$ before applying $\mathbf{K}$. While the adjustment in the optimal method is related to a Maximum Likelihood estimation of Euclidean errors in the image domains, the adjustment of $\mathbf{Y}$ is a purely algebraic operation in a projective space. As has been shown in [3], however, a suitable normalization of homogeneous coordinates gives a good correspondence between Euclidean errors and operations in projective spaces. Consequently, the adjustment of $\mathbf{Y}$ is made using homogeneous coordinates which are normalized relative to each of the 2D calibration sets. In

Method	Mean L_1	Max L_1
Optimal (Hartley-Sturm)	0.37	1.37
$\mathbf{K}_0$, computed from $\mathbf{C}_1$, $\mathbf{C}_2$	0.73	2.06
$\mathbf{K}_1$, minimizing ρ	0.71	2.27
$\mathbf{K}_2$, enforcing the ep. constr.	0.42	1.00
$\mathbf{K}_3$, minimizing ρ again	0.33	1.06
$\mathbf{K}_4$, enforcing the ep. constr.	0.33	1.06

Table 1. 3D residual error.

summary, $\mathbf{Y}'$ is the adjusted version of $\mathbf{Y}$ given by

$$\mathbf{Y}' = \mathbf{T}^{-1}\,(\mathbf{I} - \hat{\mathbf{F}}\,\hat{\mathbf{F}}^T)\,\mathbf{T}\,\mathbf{Y} \tag{6}$$

where $\hat{\mathbf{F}}$ is the L_2-normalized version of $\mathbf{F}$ *transformed to normalized coordinates* and *reshaped to a 9-dim vector*, $\mathbf{I}$ is the 9×9 identity matrix, $\mathbf{T} = (\mathbf{T}_1 \otimes \mathbf{T}_2)$ (the Kronecker product of $\mathbf{T}_1$ and $\mathbf{T}_2$) and $\mathbf{T}_1$ and $\mathbf{T}_2$ are the normalizing transformations in view 1 and 2. The triangulation is then computed as $\mathbf{K}\,\mathbf{Y}'$, which means that we can construct an adjusted operator, e.g., based on $\mathbf{K}_1$, according to

$$\mathbf{K}_2 = \mathbf{K}_1\,\mathbf{T}^{-1}\,(\mathbf{I} - \hat{\mathbf{F}}\,\hat{\mathbf{F}}^T)\,\mathbf{T} \tag{7}$$

Table 1, row four, shows the 3D residual errors from $\mathbf{K}_2$; a clear reduction even though they are still not comparable to the best method. The errors can, however, be further reduced by again performing the Euclidean 3D optimization, now with $\mathbf{K}_2$ as initial value. The result is $\mathbf{K}_3$ and the corresponding errors are presented in Table 1, row five. These errors are comparable to the best of the standard methods (the optimal method).

To summarize, we have constructed a triangulation operator $\mathbf{K}$ (here corresponding to $\mathbf{K}_3$) from an initial estimate $\mathbf{K}_0$ by iteratively performing the operations of minimizing 3D L_1-errors followed by enforcing the epipolar constraint, Equation (7). The result is an operator with as low errors as any of the best standard techniques. In fact, we have performed this iteration only one and a half time, $\mathbf{K}_3$ were never compensated for the epipolar constraint. If this is done, the resulting $\mathbf{K}_4$ has errors presented in the sixth row of Table 1. As seen, no improvement is achieve from this step, so $\mathbf{K}_3$ appears as very close to a fix point solution to the iteration.

5. Summary and discussion

This paper presents a novel approach for defining a triangulation operator $\mathbf{K}$, which maps corresponding stereo image coordinates to a reconstructed 3D point as a linear transformation on the outer product of the homogeneous image coordinates. The method uses Euclidean optimization in the 3D domain; the resulting $\mathbf{K}$ minimizes the L_1-norm of the 3D residual error using an two-step iterative method. Given an initial $\mathbf{K}$, computed from the camera matrices and a suitable blind plane [5], the first step of each iteration adjusts $\mathbf{K}$ to minimize the 3D error. The second step adjusts $\mathbf{K}$ to enforce the epipolar constraint on the image coordinates, Equation (7), and these steps need to be repeated only a few times. Given a set of 2D+2D+3D calibration points, this iterative optimization is only required once and the resulting $\mathbf{K}$ can triangulate points close to the calibration set with small residual errors. Experiments on real data show that the $\mathbf{L}_1$-norm of 3D residual errors are as low as the best standard method (optimal triangulation). This fact combined with the computational efficiency of the operator based triangulation, makes it attractive for real-time applications or when large sets of points have to be triangulated at low computational cost.

All figures reported in Table 1 are based on deriving $\mathbf{C}_1$, $\mathbf{C}_2$, $\mathbf{F}$, and $\mathbf{K}$ from a smaller set of calibration points, while the reconstruction and error computations are made on a separate evaluation set. The first set, however, lies within the latter which means that the reported performance can be expected when the calibration set is representative for the point which are triangulated in a subsequent operation phase.

The work reported here has been made within the VISCOS project, funded by the Swedish Foundation for Strategic Research (SSF). It has been extended also to the three-view case in [6], which reports similar results: the 3D L_1 error can be brought to a similar level as the method which is optimal with respect to 2D L_2 errors but at significantly lower computational cost provided that the necessary calibration step can be performed.

References

[1] F. Devernay and O. Faugeras. Straight lines have to be straight: Automatic calibration and removal of distortion from scenes of structured environments. *Machine Vision and Applications*, 13:14–24, 2001.

[2] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision, 2nd ed.* Cambridge Univ. Press, 2003.

[3] R. I. Hartley. In defence of the 8-point algorithm. *IEEE Trans. on Pattern Recognition and Machine Intelligence*, 19(6):580–593, 1997.

[4] R. I. Hartley and P. Sturm. Triangulation. *Computer Vision and Image Understanding*, 68(2):146–157, 1997.

[5] K. Nordberg. A linear mapping for stereo triangulation. In *Proceedings of Scandinavian Conference on Image Analysis, 2007*, pages 838–847, Aalborg, Denmark. Springer LNCS Vol. 4522.

[6] K. Nordberg. Efficient three-view triangulation based on 3d optimization. In *Proceedings of British Machine Vision Conference*, Leeds, UK, 2008.