A New Autocalibration Algorithm: Experimental Evaluation

Andrea Fusiello*

Dipartimento di Informatica - Università degli Studi di Verona
Strada Le Grazie, 15 - 37134 Verona, Italy
fusiello@sci.univr.it

Abstract. A new autocalibration algorithm has been recently presented by Mendonça and Cipolla which is both simple and nearly globally convergent. Analysis of convergence is missing in the original article. This paper fills the gap, presenting an extensive experimental evaluation of the Mendonça and Cipolla algorithm, aimed at assessing both accuracy and sensitivity to initialization. Results show that its accuracy is fair, and – remarkably – it converges from almost everywhere. This is very significant, because most of the existing algorithms are either complicated or they need to be started very close to the solution.

Keywords: computer vision, self-calibration

The classical approach to *autocalibration* (or *self-calibration*), in the case of a single moving camera with constant but unknown intrinsic parameters and unknown motion, is based on the recovery of the intrinsic parameters by solving the Kruppa equations [1,2], which have been found to be very sensitive to noise [2]. Recently new methods based on the *stratification* approach have appeared, which upgrade a projective reconstruction to an Euclidean one without solving explicitly for the intrinsic parameters (see [3] for a review). An algorithm has been recently presented by Mendonça and Cipolla [4], which, like the Kruppa equations, is based on the direct recovery of intrinsic parameters, but it is simpler.

Apart from sensitivity to noise, the applicability of autocalibration techniques in the real world depends on the issue of initialization. Since a non-linear minimization is always required, convergence to the global minimum is guaranteed only if the algorithm is initialized in the proper basin of attraction. Unfortunately, this issue was not addressed by Mendonça and Cipolla.

This paper gives an account of the experimental evaluation of the Mendonça and Cipolla algorithm (in mine implementation), aimed at assessing its performances, especially the sensitivity to initialization. Results are quite interesting, as it turns out that the algorithm converges to the global minimum from *almost everywhere.*

[*] This article has been written while the author was a Visiting Research Fellow at the Department of Computing and Electrical Engineering - Heriot-Watt University, supported by EPSRC (Grant GR/M40844).

W. Skarbek (Ed.): CAIP 2001, LNCS 2124, pp. 717–724, 2001.

1 Notation and Basics

This section introduces the mathematical background on perspective projections necessary for our purposes.

A pinhole camera is modeled by its 3 × 4 *perspective projection matrix* (or simply *camera matrix*) $\tilde{\mathbf{P}}$, which can be decomposed into

$$\tilde{\mathbf{P}} = \mathbf{A}[\mathbf{R} \mid \mathbf{t}]. \tag{1}$$

The matrix $\mathbf{A}$ depends on the *intrinsic parameters*, and has the following form:

$$\mathbf{A} = \begin{bmatrix} \alpha_u & \gamma & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{2}$$

where α_u, α_v are the focal lengths in horizontal and vertical pixels, respectively, (u_0, v_0) are the coordinates of the *principal point*, given by the intersection of the optical axis with the retinal plane, and γ is the *skew* factor. The camera position and orientation (*extrinsic parameters*), are encoded by the 3 × 3 rotation matrix $\mathbf{R}$ and the translation $\mathbf{t}$.

Let $\tilde{\mathbf{w}} = [x, y, z, 1]^\top$ be the homogeneous coordinates of a 3D point in the world reference frame (fixed arbitrarily) and $\tilde{\mathbf{m}} = [u, v, 1]^\top$ the homogeneous coordinates of its pojection onto the image. The transformation from $\tilde{\mathbf{w}}$ to $\tilde{\mathbf{m}}$ is given by

$$\kappa \tilde{\mathbf{m}} = \tilde{\mathbf{P}} \tilde{\mathbf{w}}, \tag{3}$$

where κ is a scale factor.

Let us consider the case of two cameras. A three-dimensional point $\mathbf{w}$ is projected onto both image planes, to points $\tilde{\mathbf{m}} = \tilde{\mathbf{P}} \tilde{\mathbf{w}}$ and $\tilde{\mathbf{m}}' = \tilde{\mathbf{P}}' \tilde{\mathbf{w}}$, which constitute a *conjugate pair*. It can be shown [5] that the following equation holds:

$$\tilde{\mathbf{m}}'^\top \mathbf{F} \tilde{\mathbf{m}} = 0, \tag{4}$$

where $\mathbf{F}$ is the *fundamental matrix.* The rank of $\mathbf{F}$ is in general two and, being defined up to a scale factor, it depends upon seven parameters. In the most general case, all the geometrical information that can be computed from pairs of images are encoded by the fundamental matrix. Its computation requires a minimum of eight conjugate points to obtain a unique solution [5]. It can be shown [5] that

$$\mathbf{F} = \mathbf{A}'^{-\top} \mathbf{E} \mathbf{A}^{-1}. \tag{5}$$

where $\mathbf{E}$ is the *essential matrix*, which can be obtained from conjugate pairs when intrinsic parameters are known. The essential matrix encodes the rigid transformation between the two cameras, and, being defined up to a scale factor, it depends upon five independent parameters: three for the rotation and two for the translation up to a scale factor. Unlike the fundamental matrix, the only property of which is being of rank two, the essential matrix is characterized by the following Theorem (see [6] for a proof).

Theorem 11 *A real matrix* $\mathbf{E}$ 3×3 *can be factorized as product of a nonzero skew-symmetric matrix and a rotation matrix if and only if* $\mathbf{E}$ *has two identical singular values and a zero singular value.*

2 Autocalibration

In many practical cases, the intrinsic parameters are unknown and the only information that can be extracted from a sequence are point correspondences, which allow to compute a set of fundamental matrices. *Autocalibration* consist in computing the intrinsic parameters, or – in general – Euclidean information, starting from fundamental matrices (or, equivalently, from point correspondences). In this section we will see which constraints are available for the autocalibration.

2.1 Two-Views Constraints

As we saw in Section 1, the epipolar geometry of two views is described by the fundamental matrix, which depends on seven parameters. Since the five parameters of the essential matrix are needed to describe the rigid displacement, at most two independent constraints are available for the computation of the intrinsic parameters from the fundamental matrix.

These two constraints come from the characterization of the essential matrix given by Theorem 11. Indeed, the condition that the matrix $\mathbf{E}$ has a zero singular value and two non-zero equal singular values is equivalent to the following conditions, found by Huang and Faugeras [7]:

$$\det(\mathbf{E}) = 0 \quad \text{and} \quad \mathbf{trace}((\mathbf{E}\mathbf{E}^\top))^2 - 2\mathbf{trace}((\mathbf{E}\mathbf{E}^\top)^2) = 0. \tag{6}$$

The first condition is automatically satisfied, since $\det(\mathbf{F}) = 0$, but the second condition can be decomposed [5] in two independent polynomial relations that are equivalent to the two equations found by Trivedi [8].

This is an algebraic interpretation of the so-called *rigidity constraint*, namely the fact that for any fundamental matrix $\mathbf{F}$ there exist two intrinsic parameters matrix $\mathbf{A}$ and $\mathbf{A}'$ and a rigid motion represented by $\mathbf{t}$ and $\mathbf{R}$ such that $\mathbf{F} = \mathbf{A}'^{-\top}([\mathbf{t}]_\wedge \mathbf{R})\mathbf{A}^{-1}$. By exploiting this constraint, Hartley [6] devised an algorithm to factorize the fundamental matrix that yields the five motion parameters and the two different focal lengths. He also pointed out that no more information could be extracted from the fundamental matrix without making additional assumptions (e.g. constant intrinsic parameters).

2.2 N-Views Constraints

The case of three views is not a straightforward generalization of the two-views case. The epipolar geometry can be described using the *canonical decomposition* [9] or the *trifocal tensor*, both of which use the minimal number of parameters, that turns out to be 18. The rigid displacement is described by 11 parameters:

6 for 2 rotations, 4 for two directions of translation and 1 ratio of translation norms. Therefore, in this case there are seven constraints available on the intrinsic parameters. If they are constant, three views are sufficient to recover all the five intrinsic parameters.

In the general case of n views, Luong demonstrated that at least $11n - 15$ parameters are needed to describe the epipolar geometry, using his canonical decomposition. The rigid displacement is described by $6n-7$ parameters: $3(n-1)$ for rotations, $2(n-1)$ for translations, and $n-2$ ratios of translation norms. There are, thus, $5n - 8$ constraints available for computing the intrinsic parameters. Let us suppose that n_k parameters are known and n_c parameters are constant. Every view apart from the first one introduces $5 - n_k - n_c$ unknowns; the first view introduces $5 - n_k$ unknowns, therefore the unknown intrinsic parameters can be computed provided that

$$5n - 8 \geq (n-1)(5 - n_k - n_c) + 5 - n_k, \tag{7}$$

which is equivalent to the following equation reported in [10]:

$$nn_k + (n-1)n_c \geq 8. \tag{8}$$

As pointed out in [9], the $n(n-1)/2$ fundamental matrices are not independent, hence the $n(n-1)$ constraints like (Eq. 6) that can be derived from them are not independent. Nevertheless they can be used for computing the intrinsic parameters, since redundancy improves stability, as mentioned in [4].

2.3 The Mendonça and Cipolla Algorithm

Mendonça and Cipolla method for autocalibration is based on the exploitation Theorem 11. A cost function is designed, which takes the intrinsic parameters as arguments, and the fundamental matrices as parameters, and returns a positive value proportional to the difference between the two non-zero singular value of the essential matrix. Let $\mathbf{F}_{ij}$ be the fundamental matrix relating views i and j, and let $\mathbf{A}_i$ and $\mathbf{A}_j$ be the respective intrinsic parameters matrices. Let ${}^1\sigma_{ij} > {}^2\sigma_{ij}$ be the non zero singular values of $\mathbf{E}_{ij} = \mathbf{A}_i^\top \mathbf{F}_{ij} \mathbf{A}_j$. The cost function is

$$\mathrm{C}(\mathbf{A}_i \; i = 1 \ldots n) = \sum_{i=1}^{n} \sum_{j>n}^{n} w_{ij} \frac{{}^1\sigma_{ij} - {}^2\sigma_{ij}}{{}^2\sigma_{ij}}, \tag{9}$$

where w_{ij} are normalized weight factors.

3 Experiments

In these experiments, intrinsic parameters were kept constant, hence the following cost function was actually used:

$$\mathrm{C}(\mathbf{A}) = \sum_{i=1}^{n} \sum_{j>n}^{n} w_{ij} \frac{{}^1\sigma_{ij} - {}^2\sigma_{ij}}{{}^1\sigma_{ij} + {}^2\sigma_{ij}} \tag{10}$$

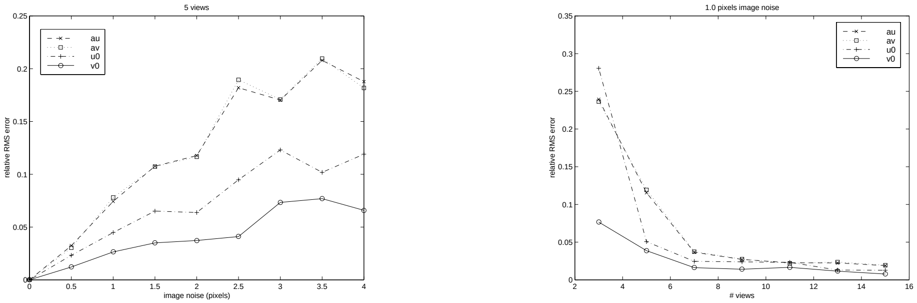


Fig. 1. Relative RMS error on intrinsic parameters versus image noise standard deviation (left) and number of views (right).

As customary it was assumed $\gamma = 0$. The weight w_{ij} was the residual of the estimation of $\mathbf{F}_{ij}$, as suggested by [4]. The minimum number of views required to achieve autocalibration in this case is three, according to (8). Fundamental matrices were computed using the linear 8-point algorithm with data normalization.

The algorithm was tested on synthetic data, which consisted of 50 points randomly scattered in a sphere of radius 1 unit, centered at the origin. Random views were generated by placing cameras at random positions, at a mean distance from the centre of 2.5 units with a standard deviation of 0.25 units. The orientations of the cameras were chosen randomly with the constraint that the optical axis should point towards the centre. The intrinsic parameters were given a known value: $\alpha_u = \alpha_v = 800, u_0 = v_0 = 256$. Image points were (roughly) contained in a 512x512 image.

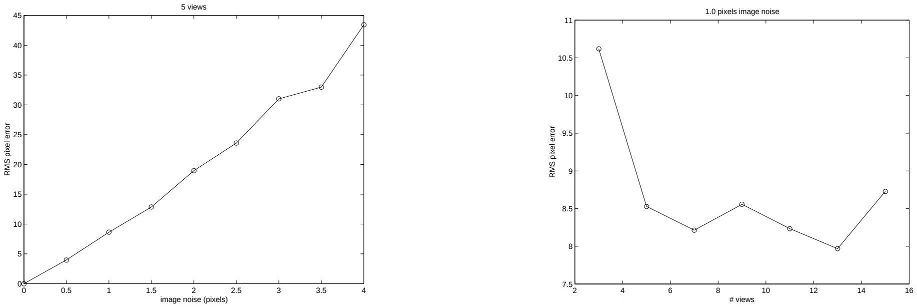


Fig. 2. Reconstruction residual RMS pixel error versus image noise standard deviation (left) and number of views (right).

I used the Nelder-Meads simplex method (implemented in the `fmins` function of MATLAB), to minimise the cost function. This methods does not use gradient information and is less efficient than Newton methods, particularly if the function is relatively smooth as in our case.

In order to determine the accuracy of the algorithm, Gaussian noise with variable standard deviation was added to image points. The algorithm was started from the true values of the intrinsic parameters and always converged to a nearby solution. Since the fundamental matrices are affected by the image noise, the minimum of the cost function does not coincide with the actual intrinsic parameters. The relative RMS error is reported in Figure 1. Each point is the average of 70 independent trials.

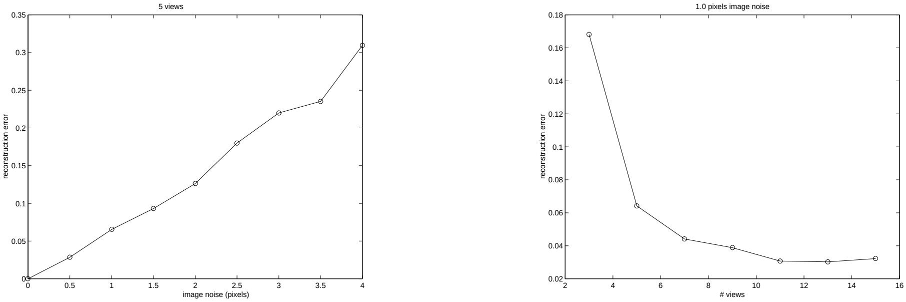


Fig. 3. Reconstruction RMS error versus image noise standard deviation (left) and number of views (right).

Using the intrinsic parameters computed by autocalibration, and the fundamental matrices, structure was recovered by first factorizing out the motion from the essential matrices, then recovering the projection matrices and finally computing 3-D structure from projection matrices and point correspondences by *triangulation*. More details and references can be found in [11].

The *pixel error* is the distance between the actual image coordinates and the ones derived from the reconstruction. The reconstruction error is the distance between the actual and the reconstructed point locations. Figures 2 and 3 report RMS errors, averaged over 70 independent trials.

In order to evaluate the sensitivity to the initialization, I ran an experiment in which the algorithm was initialized by perturbing the actual value of the intrinsic parameters with uniform noise with zero mean and increasing amplitude. For each standard deviation value I ran 70 independent trials and recorded how many times the algorithm converged to the correct solution, which was assumed to be the one to which it converged when initialized with the actual intrinsic parameters. Perturbation was obtained with the following formula (in MATLAB syntax):

```
a0 = a_true + pert * a_true.*(rand(1,4)-0.5)
```

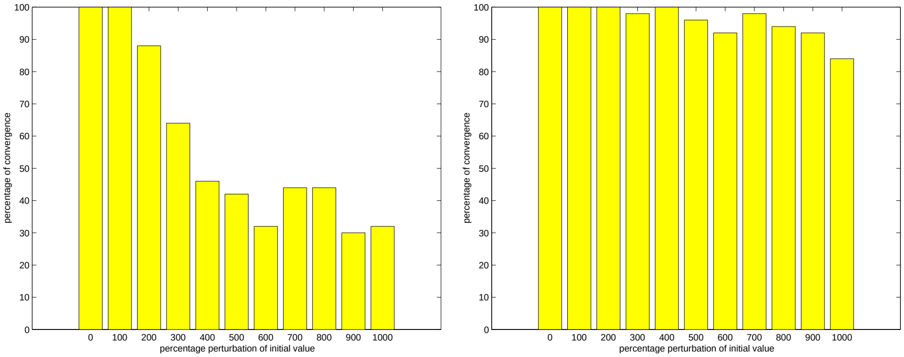
where `a0` is the initialization, `a_true` is a vector containing the true intrinsic parameters, and `pert` is a value ranging from 0 to 10 (corresponding to 1000%!).

Figure 4(a) shows the result with 5 views and 1.0 pixels image noise. The same experiment with 15 views yielded very similar result, not shown here. In another experiments I used only positive uniform noise:

```
a0 = a_true + pert * a_true.*(rand(1,4))
```

and the results are shown in Fig 4(b).

Finally, the algorithm was initialized with a random point in the 4D cube $[0, 2000] \times [0, 2000] \times [0, 2000] \times [0, 2000]$ and it converged in the 86% of cases, with 5 views and 1.0 pixels image noise.



(a) Zero-mean uniform noise

(b) Positive uniform noise

Fig. 4. Percentage of convergence vs initial value perturbation (percentage) for 5 views.

On the average, it takes 5 seconds of CPU time on a Sun Ultra 10 running MATLAB to compute intrinsic parameters with 5 views.

The MATLAB code used in the experiments is available on the web from `http://www.sci.univr.it/~fusiello/demo/mc`.

4 Discussion

Intrinsic parameters are recovered with fair, but not excellent, accuracy. The error consistently increases with image noise and decreases with the number of views. With 1.0 pixel noise no appreciable improvement is gained by using more than seven views, but this number is expected to increase with the noise. It is not advisable to use the minimum number of views (three).

As for the reconstruction, the residual pixel error depends only on the image noise and not sensibly on the number of views (excluding the three views case). The reconstruction error, consistently decreases with the number of views. With 5 views and image noise of 1.0 pixel, the accuracy is about 30%. This figure depends only partially on the computation of the intrinsic parameters. It also depends on the recovery of motion parameters and on the triangulation. In both cases linear algorithm were used. Improvements can be expected by using non-linear refinement.

The algorithm shows excellent convergence properties. Remarkably, even when true values are perturbed with a relative error of 200% convergence is achieved in the 90% of the cases (Figure 4(a)). Results suggest that failure occurs when the sign of the parameters is changed. Indeed, figures improve dramatically when perturbation is a positive uniform random variable: in this case the algorithm converges from almost everywhere (Figure 4(b)).

In summary, the algorithm is fast and converges in a wide basin, but accuracy is not its best feature. If accuracy is a concern, it is advisable to run a bundle adjustment, which is known to be the most accurate method, but very sensitive to initialization.

References

1. Maybank, S.J., Faugeras, O.: A theory of self-calibration of a moving camera. International Journal of Computer Vision **8**(2) (1992) 123–151
2. Luong, Q.T., Faugeras, O.: Self-calibration of a moving camera from point correspondences and fundamental matrices. International Journal of Computer Vision **22**(3) (1997) 261–289
3. Fusiello, A.: Uncalibrated Euclidean reconstruction: A review. Image and Vision Computing **18**(6-7) (May 2000) 555–563
4. Mendonça, P.R.S., Cipolla, R.: A simple technique for self-calibration. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (1999) 500–505
5. Luong, Q.T., Faugeras, O.D.: The fundamental matrix: Theory, algorithms, and stability analysis. International Journal of Computer Vision **17** (1996)
6. Hartley, R.I.: Estimation of relative camera position for uncalibrated cameras. Proceedings of the European Conference on Computer Vision. Santa Margherita L. (1992) 579–587
7. Huang, T.S., Faugeras, O.D.: Some properties of the E matrix in two-view motion estimation. IEEE Transactions on Pattern Analysis and Machine Intelligence **11**(12) (Dec 1989) 1310–1312
8. Trivedi, H.P.: Can multiple views make up for lack of camera registration? Image and Vision Computing, **6**(1) (1988) 29–32
9. Luong, Q.-T., Viéville, T.: Canonical representations for the geometries of multiple projective views. Computer Vision and Image Understanding **64**(2) (1996) 193–229
10. Pollefeys, M., Koch, R., Van Gool, L.: Self-calibration and metric reconstruction in spite of varying and unknown internal camera parameters. Proceedings of the IEEE International Conference on Computer Vision. Bombay (1998) 90–95
11. Fusiello, A.: The Mendonça and Cipolla self-calibration algorithm: Experimental evaluation. Research Memorandum RM/99/12. Department of Computing and Electrical Engineering, Heriot-Watt University, Edinburgh, UK (1999). Available at ftp://ftp.sci.univr.it/pub/Papers/Fusiello/RM-99-12.ps.gz