Camera Calibration using Parallelism Constraints

Ioannis Stamos

1.0 Introduction - Basic Idea

The purpose of this work is the construction of a method for camera calibration. Our ultimate goal is the reconstruction of the 3D structure of buildings. We want to exploit the constraints imposed by three orthogonal directions in the scene. Three sets of scene lines which are mutually orthogonal have projections on the image which define three vanishing points when the imaging process is ideal. Due to lens imperfections the projected edges are distorted.

We want to correct this distortion taking into account the fact that each one of the three projected sets of lines should define one vanishing point.

Using the three vanishing points we compute after the undistortion we can compute the principal point on the image plane and the effective focal length of the camera. That means that we can compute the center of projection (COP) with respect to the coordinate frame of the camera. The COP is the forth apex of the right tetrahedron whose other three vertices are the computed undistorted vanishing points.

2.0 The camera model

We can correct an observed distorted image pixel (x_d, y_d) using the following model ((x_u, y_u) is the pixel after the correction of the distortion):

$$x_u = x_d + \bar{x}_d(K_1 r_d^2 + K_2 r_d^6) + (P_1(r_d^2 + 2\bar{x}_d^2) + 2P_2\bar{x}_d\bar{y}_d)(1 + P_3 r_d^2)$$
$$y_u = y_d + \bar{y}_d(K_1 r_d^2 + K_2 r_d^6) + (P_1(r_d^2 + 2\bar{y}_d^2) + 2P_2\bar{x}_d\bar{y}_d)(1 + P_3 r_d^2)$$

- $\bar{x}_d = \dfrac{x_d - c_x}{R_{max}}, \bar{y}_d = \dfrac{y_d - c_y}{R_{max}}$: distorted coordinates wrt the coordinate system which has as center the principal point, scaled by the maximum film radius (not nessecary).
- $r_d = \sqrt{\bar{x}_d^2 + \bar{y}_d^2}$: distance of the distorted point from the principal point.
- K_1, K_2: radial distortion.
- P_1, P_2, P_3: principal point's shear.

3.0 Computation of the distortion coefficients from one set of parallel scene lines

We assume at this stage that we have an initial good approximation of the camera's center of projection (COP) wrt camera's coordinate system: $\tilde{\boldsymbol{c}} = \left[pp_x \ pp_y \ ff\right]^T$, where (pp_x, pp_y) is the camera's principal point and ff is the effective focal length.

We are considering k parallel scene lines of direction d_i. The projection of these lines on the image plane are the edges e_k. The plane defined by the edge e_k and the approximate COP $\tilde{\boldsymbol{c}}$ has a normal $\boldsymbol{n_k}$. Under ideal conditions (no distortion) the normals which correspond to all k lines lie on a single plane P. The normal of this plane is parallel to the direction d_i and therefore the vector intersects the image plane on the vanishing point VP^i of the direction d_i.

When camera distortion comes into play the edges e_k do not intersect at a single vanishing point and the normals $\boldsymbol{n_k}$ do not lie on a single plane.

The goal of the minimization procedure is to undistort the observed distorted image edges such that the resulted undistorted edges produce normals $\boldsymbol{n_k}$ which are coplanar.The model of section 2.0 is an approximation of the real physical process. Noise comes into play in the detection of edges (or in the selection of edges by a user). Lastly, we have an approximate COP at this point . Therefore we can not hope in normals which are coplanar. Our goal is to make them as coplanar as possible.

3.1 A measure of coplanarity

The distance of the endpoint of the vector $\boldsymbol{n_k} - \tilde{\boldsymbol{c}}$ from a plane which passes through the point $\tilde{\boldsymbol{c}}$ and has a normal $\hat{\boldsymbol{n}}$ is $(n^k{}_0\hat{n}_x + n^k{}_1\hat{n}_y + n^k{}_2\hat{n}_z) = D_k$ ($\boldsymbol{n_k} = [n_0^k, n_1^k, n_2^k]^T$, $\hat{\boldsymbol{n}} = [\hat{n}_x, \hat{n}_y, \hat{n}_z]^T$). We want to find the plane which has the minimum total distance from all endpoints, that is:

$min\big|_{\hat{n}_x, \hat{n}_y, \hat{n}_z} \sum (n^k{}_0\hat{n}_x + n^k{}_1\hat{n}_y + n^k{}_2\hat{n}_z)^2$ such that $\hat{n}_x^2 + \hat{n}_y^2 + \hat{n}_z^2 = 1$ (the equality constraint is important, otherwise the result of the minimization would be the 0 vector). This is a constraint minimization problem and can be solved using Lagrance multipliers. We have to minimize the function

$$h(\hat{n}_x, \hat{n}_y, \hat{n}_z, \mu) = \sum (n^k{}_0\hat{n}_x + n^k{}_1\hat{n}_y + n^k{}_2\hat{n}_z)^2 - \mu(\hat{n}_x^2 + \hat{n}_y^2 + \hat{n}_z^2 - 1)$$

which leads to the following

$$\overbrace{\left(\sum_k \begin{bmatrix} (n_0^k)^2 & n_0^k n_1^k & n_0^k n_2^k \\ n_0^k n_1^k & (n_1^k)^2 & n_1^k n_2^k \\ n_0^k n_2^k & n_1^k n_2^k & (n_2^k)^2 \end{bmatrix}\right)}\underbrace{}\begin{bmatrix} \hat{n}_x \\ \hat{n}_y \\ \hat{n}_z \end{bmatrix} = \mu \begin{bmatrix} \hat{n}_x \\ \hat{n}_y \\ \hat{n}_z \end{bmatrix}$$

$$\left(\sum_k \boldsymbol{n_k} \boldsymbol{n_k}^T\right)\hat{\boldsymbol{n}} = \mu\hat{\boldsymbol{n}}$$

$$A\hat{\boldsymbol{n}} = \mu\hat{\boldsymbol{n}}$$

That means that one of the three eigenvectors[1] of the symmetric matrix A is the normal of the plane with the minimum distance (in the least squares sense) from the endpoints of the vectors $\boldsymbol{n_k} - \tilde{\boldsymbol{c}}$.

The eigenvector $\hat{n}_{min}$ which corresponds to the smallest of the three eigenvalues μ_{min} is the normal of the best plane which fits our data. Also we have:

$$|\mu_{min}\hat{n}_{min}| = |\mu_{min}| = \left|\left(\sum_k \boldsymbol{n_k}\boldsymbol{n_k}^T\right)\hat{n}_{min}\right| = \left|\sum_k \boldsymbol{n_k}(\boldsymbol{n_k}^T\hat{n}_{min})\right|$$

But the dot product $\boldsymbol{n_k}^T\hat{n}_{min}$ is the distance D_k of the vector $\boldsymbol{n_k}$ from the best plane (since $\hat{n}_{min}$ is the normal of this plane). That is

$$|\mu_{min}| = \left|\sum_k D_k \boldsymbol{n_k}\right|$$

Thus, the smallest eigenvalue is used as a measure of the coplanarity of our vectors. Our goal is to minimize the smallest eigenvalue in order to make our data as planar as possible.

Another approach would be to minimize the sum $\sum_k D_k$. Both approaches lead to similar results.

1. Those eigenvectors are orthogonal to each other since A is symmetric.

3.2 The function we want to minimize

We want to minimize the 5-D function $f_k(K_1, K_2, P_1, P_1, P_3) = \mu_{min}$ when we are given the a set of k distorted image segments (x_d^k, y_d^k) and an approximate COP $\tilde{\boldsymbol{c}}$. For a particular value of the distortion coefficients $(K_1, K_2, P_1, P_1, P_3)$ we undistort the endpoints of the (x_d^k, y_d^k) segments. That results to the undistorted segments (x_u^k, y_u^k). Using the undistorted segments and the approximate COP we can calculate the normals $\boldsymbol{n_k}$ and then compute the smallest eigenvalue according to section 3.1. The minimization algorithm will computes the distortion parameters which provide the minimum smallest eigenvalue μ_{min}.

4.0 Computation of the distortion coefficients from multiple sets of parallel lines.

The only difference in this case is that we want to minimize the function

$$\sum_k f_k(K_1, K_2, P_1, P_1, P_3) = \sum_k \mu^k{}_{min}$$

5.0 Computation of the COP

We are using the following property of three vanishing points (VP_1, VP_2, VP_3) which correspond to three sets of parallel lines which are orthogonal (the sets) to each other:

The three vanishing points together with the COP form a right tetrahedron. That means that the principal point is the orthocenter of the triangle $\Delta(VP_1, VP_2, VP_3)$ *and the effective focal length is the height of this tetrahedron.*

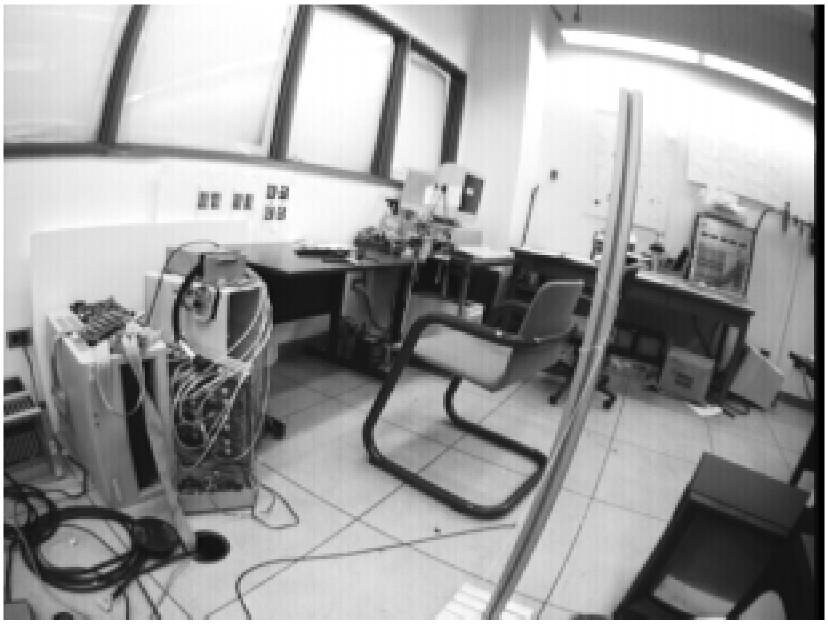
It is easy to prove this property. See 2nd reference.

We need three sets of parallel lines which are orthogonal to each other in this case. The vanishing point of each set equals to the intersection of the vector $\hat{n}_{min} - \tilde{\boldsymbol{c}}$ with the image plane (section 3.1). From the three vanishing points we compute a better estimation of the COP $\boldsymbol{c_{better}}$.

6.0 Integration of distortion and COP computation

We start with an initial COP and compute the distortion coefficients (section 3.0). Then we calculate a better approximation of the COP (section 5.0). We use the newest approximation of the COP and recompute the distortion coefficients. We recalculate an approximation of COP and so on. We stop when our observed error $\sum_k \mu^k{}_{min}$ does not change dramatically.

7.0 Results



Distorted (up) and undistorted image (down). The computed parameters are:

TABLE 1.

Radial Coeff.	3.305956E-06	2.223277E-11	
Tangential Coeff.	-1.006010E-04	3.241836E-04	1.247927E-05
Prinipal Point	462.702497	214.072428	in pixels
Focal Length	223.023460		in pixels

The initial distorted image has size 640 by 480.

8.0 Next steps

- Edge detection in the original image. Minimize noise by user's selection of edges.
- Overconstrain the problem using more that 3 orthogonal image directions.
- Organize the simulation results into charts and investigate the behavior of the algorithm.
- Estimate the rotation/translation parameters of each camera.
- Improve the user interface (right now implemented in TCL-TK).
- Improve the termination criterion for the minimization algorithm.

9.0 References

The basic reference is the PhD thesis of Shawn Becker. I used his ideas. I changed the observation function to be minimized (section 3.1, coplanarity). The matrix A in Becker's thesis is $\frac{1}{N}\sum_k \boldsymbol{n_k}\boldsymbol{n_k}^T - E[\boldsymbol{n_k}]E[\boldsymbol{n_k}^T]$, where $E[\boldsymbol{n_k}] = \frac{1}{N}\sum_k \boldsymbol{n_k}$ and N are the number of observed distorted image segments.

1. *Vision-assisted modeling for model-based video representations,* Shawn Becker, PhD thesis, MIT 1996
2. *Using Vanishing Points for Camera Calibration,* B. Caprile and V. Torre, IJCV 4, 127-140 (1990)
3. *Decentering Distortion of Lenses*, Duane C. Brown, Symposium on close-range photogrammetry, Urbana, Illinois, 1971.