Camera Self-Calibration

Navid Azizi

April 30, 2003

1 Introduction

Camera calibration is the problem of estimating a camera's intrinsic parameters, which among others include the focal length and image center of the camera. The estimation of the camera's intrinsic parameters is important as it is a prerequisite to a wide variety of vision tasks related to motion and stereo analysis. Historically, camera calibration was performed by knowledge of 3D world points and their corresponding image locations. For example, first a flat plate with a regular pattern, whose 3D coordinates are well known, is placed in the field of view of the camera. Then the image coordinates of a number of well recognizable and measurable features of the pattern are used to estimate the camera's parameters [1]. The camera calibration problem can be solved in this way with the knowledge of a minimum of 6 point correspondences [2].

The above method estimates the camera parameters very accurately [1], but it has a couple drawbacks. The first drawback is that a calibration grid may not be available, and secondly it is not possible to calibrate the camera when the camera is on the field and it's parameters may be changing [1]. The change in the camera parameters may be due to mechanical or thermal variations or may be due to focusing and zooming of the camera [1].

It has become apparent, however, that camera calibration can also be performed without the use of a calibration grid, but instead by only using a number of image correspondences [3]. Many methods have been proposed that perform this *self-calibration*. Most of the self-calibration methods relate the image correspondences to the absolute conic which stays fixed under all Euclidean transformations [4]. The methods, furthermore, seem to fall into one of three categories:

1. The Kruppa Equations proposed by Maybanck and Faugeras.
2. A linear constraint on the calibration matrix pioneered by Hartley.
3. An approach that finds the explicit location of the absolute quadric, shown by Triggs.

Within each of these three self-calibration technique categories there remains many different configurations of the problem. The number of parameters that are unknown or changing and the type of camera movement can lead to different algorithms or optimizations to solve the problem. For example, an algorithm may be specialized to estimate the changing focal length of a camera in a rotating scene when all other camera parameters are known. Also, there might be different ways to solve the constraints that are placed on the camera parameters; for example algebraic or numeric techniques may be used to solve the constraints.

This paper will provide an overview of the various techniques that have been presented in the literature to solve the camera self-calibration problem. Section 2 will give an overview of camera parameters and the absolute conic that will be needed as background. Section 3 will go over Kruppa's Equations. In section 4 various algorithms based on Hartley's method will be explored, and in section 5 methods based on the absolute quadric will be presented. Section 6 will present some practical constraints in the self-calibration problem such as the effect of radial distortion, and critical motion sequences. Finally, section 7 will conclude.

2 Background

Throughout this survey paper perspective projection will be used to model cameras. A projection that maps a 3D point, Q, to an image point, q, is a 3×4 matrix, P such that $q \sim PQ$ [5]. The projection matrix can be decomposed into two matrices; a calibration matrix, K and a matrix representing the position and orientation of the camera [6]:

$$P \sim KR(I| - t) \tag{1}$$

where R is a rotation matrix, and t is a vector representing the position of the camera [7]. Since K is multiplied by a rotation, R, in equation 1, K can be assumed to be upper triangular as follows [8]

$$K = \begin{bmatrix} f_x & s & o_x \\ 0 & f_y & o_y \\ 0 & 0 & 1 \end{bmatrix}$$

where f_x and f_y are the focal lengths with respect to the width and height of pixels respectively [7], o_x and o_y are the coordinates of the principal point and s is the skew. s can be normally assumed to be 0, except when the pixels are not exactly rectangular; then s is non-zero.

2.1 Quadrics and Conics

A set of points in $\mathbb{P}^n$ satisfying a quadratic equation in their homogeneous coordinates is said to be a quadric [6]; a symmetric $(n + 1) \times (n + 1)$ matrix can represent a

$n-$dimensional quadric [6]. In 3D space, a quadric is a 4×4 matrix $\mathbf{Q}$ where the relationship between the set of homogeneous points and the quadric can be represented by

$$M^T Q M = 0 \tag{2}$$

where M is a homogeneous representation of a 3D-point [7]. A quadric is proper if it's matrix has a non-zero determinant and a virtual quadric is a quadric with no real points [6].

The equivalent concept in 2D is the conic, and the projection of the quadric onto an image is a conic which can be expressed as [7]

$$C^* \sim P Q^* P^T \tag{3}$$

When an image is taken of a scene, the affine structure is lost in the image, and is only encoded by the plane at infinity, Π_∞, and the proper virtual conic on Π_∞; this conic on Π_∞ is called the absolute conic and is denoted by Ω^*_∞ [7]. Furthermore, Euclidean transformations leave the absolute conic unchanged, and thus the image of the absolute conic is only dependent on the intrinsic and not extrinsic parameters of the camera [7]. Thus, finding the intrinsic parameters of a camera is equivalent to finding the image of the absolute conic, ω [6]

$$\omega^* \sim P \Omega^*_\infty P^T \sim K K^T \tag{4}$$

Given enough different views of a scene under certain kinds of motion it becomes possible to pinpoint the absolute conic, and thus the camera's calibration.

3 Kruppa's Equations

Kruppa's equations are equations that relate the epipoles of a camera's displacement to the absolute conic, and thus to the calibration matrix K. In this section, a derivation of Kruppa's equations is shown, and then an algorithm to obtain the calibration matrices is presented.

3.1 Derivation

Let a camera take an image of a scene, undergo a displacement and take another image of the scene. Associated with the displacement are two epipoles, e and e' [1]. Since the image of the absolute conic, ω is unchanged due to the displacement, "it follows that the two tangents to ω from e correspond under the epipolar transformation to the two tangents to ω from e' [1]". The constraint of the two tangents from the epipolar lines produces two constraints linking the epipoles to the absolute conic.

Mathematically the above constraint can be written as

$$(e \times y)^T w^* (e \times y) = 0 \tag{5}$$

where y is the point chosen on the first image. [1]. If we write w^* as follows

$$w^* = \begin{bmatrix} -\delta_{23} & -\delta_3 & \delta_2 \\ \delta_3 & -\delta_{13} & \delta_1 \\ \delta_2 & \delta_1 & -\delta_{13} \end{bmatrix}$$

then equation 5 can be written as

$$A_{11}y_1^2 + 2A_{12}y_1y_2 + A_{22}y_2^2 = 0 \tag{6}$$

where

$$\begin{aligned} A_{11} &= -\delta_{13}e_3^2 - \delta_{12}e_2^2 - 2\delta_1e_2e_3 \\ A_{12} &= \delta_{12}e_1e_2 - \delta_3e_3^2 + \delta_2e_2e_3 + \delta_1e_1e_3 \\ A_{22} &= -\delta_{23}e_3^2 - \delta_{12}e_1^2 - 2\delta_2e_1e_3 \end{aligned} \tag{7}$$

Furthermore, a similar constraint and equations can be placed on e' [1]. One other operation that is needed, however, is a bilinear transformation, which can easily be found from the two epipoles [1]. If we let the bilinear transformation be

$$\tau' = \frac{a\tau + b}{c\tau + d} \tag{8}$$

then Kruppa's equations are obtained to be

$$\begin{aligned} A_{12}(A'_{22}a^2 + A'_{11}c^2 + 2A'_{12}ac) - (A'_{12}a + A'_{22}a + A'_{11}bc + A'_{12}ab)A_{11} &= 0 \\ A_{22}(A'_{22}a^2 + A'_{11}c^2 + 2A'_{12}ac) - (2A'_{12}b + A'_{22} + A'_{11}b^2)A_{11} &= 0 \end{aligned} \tag{9}$$

It was later found that these exact constraints can be obtained by relating the epipoles and the Fundamental Matrix, F.

3.2 Obtaining the Calibration with Kruppa's Equations

Since each displacement produces two constraints, and the camera calibration matrix has five unknowns, three camera displacements are needed to determine the camera calibration uniquely [9]. With three displacements, there are six equations for 5 unknowns, and thus the problem is over-determined.

[1] states that numerical methods usually fail to solve the problem, and thus one equation is thrown out, and five equations are used to solve for the five unknowns. The sixth equation is only used to discard impossible solutions. There are other methods, however, to solve the system of equations such as least-squares optimization [10] or even singular-value-decomposition can be used to simplify the problem [11].

As it will be seen in Section 6, Kruppa's equations suffer from a number of singularities, for which the methods described in Section 4 do not[1]. Thus Kruppa's equations are not the ideal way of solving the camera calibration problem. Before linear methods of solving the camera calibration problem are explored, however, a derivative of Kruppa's equations will be explained.

---

[1] Singularities are instances where a unique solution cannot be found.

3.3 Derivative of Kruppa's equation

In the above method, it was assumed that none of the camera's intrinsic parameters were known. It has been shown, however, that it is valid to assume that pixels are rectangular, and that the principal point is the center of the image [5]. Thus the only parameter that remains to be estimated is the focal length.

Referring to the Fundamental Matrix derivation of Kruppa's Equations (10), [5] uses the Singular-Value-Decomposition of F to obtain equation 11, where u_3 is the third eigenvector of U. Then U^T can be multiplied on the left and U can be multiplied on the right to obtain equation 12.

$$
\begin{aligned}
F diag(f^2, f^2, 1) F^T &\sim [e']_\times diag(f^2, f^2, 1)[e']_\times && (10)\\
U \Sigma V^T \; diag(f^2, f^2, 1) V \Sigma U^T &\sim [u_3']_\times diag(f^2, f^2, 1)[u_3']_\times && (11)\\
\Sigma V^T \; diag(f^2, f^2, 1) V \Sigma &\sim \begin{bmatrix} u_2^T \\ -u_1^T \\ 0^T \end{bmatrix} diag(f^2, f^2, 1)[u_2 \quad -u_1 \quad 0] && (12)
\end{aligned}
$$

The upper 2×2 part of equation 12 leads to a symmetric matrix and thus to three constraints of which two are linear and one is quadratic [5]. The three constraints are a version of the general Kruppa's equations, but they do not suffer from all the singularities of the general Kruppa's equations [5].

4 A Linear Constraint on the Absolute Conic

The calibration method described in Section 3 placed a non-linear constraint on the absolute conic. It will be seen in Section 6 that Kruppa's equations can be numerically inaccurate in many instances and suffer from many singularities. Due to these problems, Hartley came up with a linear constraint on the absolute conic, where the calibration problem was divided into two parts: an affine part and a Euclidean part. This section will explain Hartley's derivation, and then discuss other methods that have appeared in the literature that are based on Hartley's approach that solve the calibration problem under different configurations.

4.1 Hartley's Approach

Hartley performs the Euclidean reconstruction of the scene and camera calibration in a two-step approach. In the first stage, a projective reconstruction of the scene is computed and then upgraded to a quasi-affine reconstruction, and in the second stage the quasi-affine reconstruction is transformed to a Euclidean reconstruction, from which the camera calibration can be computed.

The above statement can be described by finding the 3D projective transformation

$$
H \sim \begin{bmatrix} K^{-1} & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} I & 0 \\ v^T & 1 \end{bmatrix} \qquad (13)
$$

where the right-hand matrix in equation 13 is the transformation from the projective reconstruction to the quasi-affine reconstruction, and the left-hand matrix is the transformation from a quasi-affine reconstruction to a Euclidean reconstruction [12]. Thus the camera calibration problem is transformed into estimating the eight unknowns of H (five unknowns in K and three unknowns in v).

Hartley's specific algorithm to perform the above calibration proceeds in five steps. The first step in Hartley's approach is to estimate a projective reconstruction of the scene [12]. The reconstruction is performed by using the Essential Matrix, E. When the camera calibration is known, it is possible to use E to determine the relative placements of cameras as well as the relative location of 3D points [13]. However, it is shown in [13] that even when the cameras are uncalibrated it is possible to reconstruct E up to a projective transformation [13].

The next step is to transform the projective reconstruction found in the first step into a *quasi-affine* reconstruction. This transformation is found by solving a set of cheiral inequalities to obtain a representation of the plane at infinity, v. The cheiral inequalities result from the fact that all points in the image must be in front of the camera, and this limits the placement of the plane at infinity [12]. v is computed with the aid of dynamic programming where the objective function that is maximized is the margin by which the cheiral inequalities are maximized [12]. Intuitively the dynamic programming is placing the plane at infinity as far away from the cameras as possible [12].

In step three, K is computed by using v,t_i and R_i where t_i and R_i are the translation and rotation of the ith camera in relation to the zeroth camera. An equation relating K to the other constraints is shown to be

$$(KK^T)B_i^{-T} = B_i(KK^T) \tag{14}$$

where B_i is a function of H, R_i, v and t_i [12]. If two B_i's are known from sufficiently many views then K can be solved by Choleski factorization [12].

If only the camera calibration is needed, then the two remaining steps in Hartley's algorithm do not need to be performed, however if the algorithm is completed then a more accurate K may be found. The fourth step is a Levenberg-Marquardt (LM) iteration to find the Euclidean reconstruction, and the final step is to perform another complete LM iteration to find the optimal values for K and the Euclidean reconstruction.

4.2 Linear Methods with weakly calibrated cameras

In the above linear methods, it was assumed that none of the camera's intrinsic parameters were known. As in section 3.3, there are a number of methods to solve for the focal length of the camera when all other parameters are known.

In [8] the fundamental matrix, F, is used to solve for the two focal lengths, f_1 and f_2, of two cameras viewing the same scene. The algorithm first estimates the Fundamental matrix through eight point correspondences, and then uses Singular Value Decomposition and properties of rotation matrices to solve for the focal lengths [8].

There is however a simpler algorithm presented in [14] that also uses Singular Value Decomposition to solve for the two unknown focal lengths in a system.

As in [8], the first step is to estimate the Fundamental matrix, F, through sufficiently many point correspondences [14] and then compute the Singular Value Decomposition and obtain the eigenvectors u_i. Using the characteristics of the Essential and Fundamental matrices and the translation of the camera shown in equations 15 through 17

$$E \sim K'^T F K \tag{15}$$

$$EE^T = I - tt^T \tag{16}$$

$$t = \frac{K'^{-1}u_3}{\|K'^{-1}u_3\|} \tag{17}$$

the following constraint is obtained [14]

$$\begin{aligned} K'^T F K K^T F^T K' &\sim EE^T \\ &= I - \frac{K'^{-1}u_3 u_3^T K'^{-T}}{u_3^T K'^{-T} A'^{-1} u_3} \end{aligned} \tag{18}$$

Equation 18 through some algebraic simplification can be reduced to

$$F(I + \mu i_3 i_3^T)F^T \sim (1 + \nu(u_3^T i_3)^2)(I + \nu i_3 i_3^T) - (u_3 + \nu(u_3^T i_3) i_3)(u_3 + \nu(u_3^T i_3) i_3)^T. \tag{19}$$

where $\mu = f^{-2} - 1$ and $\nu = f'^2 - 1$ and $i_3 = [0, 0, 1]^T$ [14].

Some further simplification and rearrangement can be performed leaving three linear equations with three unknowns from which f and f' can be read off.

There is, however, a problem with the above formulation. The focal lengths cannot be identified uniquely under two circumstances. The first scenario occurs if "the optical axes of the two cameras and the baseline between them are coplanar [14]". This scenario is important since many robotic vision systems are placed in this exact configuration. The second scenario, which is of little practical interest, occurs if "one optical axis, the baseline and the vector perpendicular to the baseline and the other optical axis are coplanar [14]".

4.3 Numerical Solution to Linear Constraints

For all the methods presented above, the constraints on the absolute conic or fundamental matrix were solved in an algebraic manner, where algebraic simplifications were used to find an analytical formula for the unknowns. Another possible method to solve the problem is to use numerical methods to solve for solutions that satisfy the constraints.

The numerical method presented in [15] uses the property of the Essential Matrix that it is rank two and has two equal singular values. Since $E = K_2^T F K_1$ and F can be measured from a pair of images, K_1 and K_2 can be numerically solved to minimize the difference between the two singular values of the Essential Matrix [15].

The steps in the algorithm are as follows: first a sequence of n images are taken and $n(n-1)/2$ Fundamental matrices are found. A calibration matrix, K, is then estimated and the Essential Matrix computed. The singular values of E, $\sigma_{1_{ij}}$ and $\sigma_{2_{ij}}$ are used in the following cost function which is to be minimized:

$$C(K) = \sum_{ij} \frac{w_{ij}}{\sum_{kl} w_{kl}} \frac{\sigma_{1_{ij}} - \sigma_{2_{ij}}}{\sigma_{2_{ij}}} \tag{20}$$

w_{ij} in the cost function is a weight indicating the degree of confidence in the estimation of the Fundamental Matrix for the image pair i, j [15].

Since this method uses many pairs of images, and many point correspondences within image pairs, [15] states that the above method has a great deal of redundancy and thus is numerically stable. [15] also states that because of the inclusion of the weight element in the cost function, the algorithm eliminates any bias towards any image.

The second numerical method presented in [16] uses algebraic methods to obtain linear constraints on the fundamental matrix of a stereo pair. It then estimates the Euclidean reconstruction of the scene (which also provides the camera calibration) through a least-squares minimization technique [16]. Again the authors state that numerical methods allow for the problem to be over-constrained and thus the stability of the algorithm is increased [16].

4.4 Summary

In this section Hartley's linear constraint on the Absolute Conic was explained, where the camera calibration problem is divided into two steps; affine and then Euclidean reconstruction. The section then summarized some other camera self-calibration techniques that are based on Hartley's approach when the camera is weakly calibrated. Finally, self-calibration techniques that used numerical methods to solve for the intrinsic parameters were explained.

5 Triggs' Constraint

Instead of separating the reconstruction problem into an affine and Euclidean part as Hartley did, Triggs explicitly locates the absolute quadric in an initial projective reconstruction [11].

If the scale factor in the relationship between the absolute conic, and it's image was known, then $\omega \sim P\Omega P^T$ would be linear and could be solved trivially [11]. To remove the scale factor, ratios of components can be used and then cross-multiplied to obtain

$$\omega^{AB}(P\Omega P^T)^{CD} - \omega^{CD}(P\Omega P^T)^{AB} = 0 \tag{21}$$

Triggs calls the above equation the **absolute quadric projection constraint** [11]. This constraint can be written as $\omega \wedge (P\Omega P^T) = 0$ where it leads to 15 bilinear equations of the 16 independent components of Ω and ω or it can be written as $\omega^{-1}P\Omega P^T = \frac{1}{3}\text{trace}(\omega^{-1}P\Omega P^T)I$ as nine bilinear equations in Ω and ω^{-1} [11]. Furthermore three images are required to obtain a unique solution, but for better stability it is recommended that more than three images are used [11].

There are two methods to solve for the unknowns. The first is a non-linear method that uses a numerical approach to directly solve the quadratic constraints in $\omega \wedge (P\Omega P^T) = 0$ [11]. Sequential Quadratic Programming is used [17] with the cost function being the sum of squared violations of the projection constraints

$$\sum_i \|\omega \wedge (P\Omega P^T)\|^2. \tag{22}$$

The problem with this numerical approach is that an initial guess for Ω and ω are needed. The authors argue that any initialization will suffice, and that $\omega_0 = I$ and $\Omega_0 = I$ perform quite well and that convergence occurs in four to ten iterations [11].

Another approach shown in [11] is a quasi-linear approach where the independent components of Ω and ω are written in a vector and a Singular-Value-Decomposition is used to factorize it into vectors for Ω and ω, which can then be rewritten as matrices. With this approach, however, four images are needed, and it is less stable than the non-linear version; the method also runs slower [11].

Finally after ω is found, the camera calibration can be found by the Choleski decomposition of $\omega = KK^T$.

6 Practical Considerations

In section 3 the notion of singularities was introduced. In this section the cases which lead to singularities for all camera calibration methods will be presented. Then a discussion of why Kruppa's equations have more singularities and are numerically less stable than the linear methods introduced by Hartley and Triggs will follow. Finally the effect of radial distortion on camera calibration will be discussed.

6.1 Critical Motion Sequences

Critical Motion Sequences (CMS) are a sequence of camera motions that do not allow for a unique Euclidean reconstruction of the scene [18]. CMS's arise when there exists at least one other proper virtual conic, the potential absolute conic, Φ, besides Ω, that projects unto the same image ϕ in all the frames of the sequence [18].

Since there are at least two images of an absolute conic that remain unchanged in all images, the proper absolute conic cannot be usually properly identified. Therefore, Euclidean reconstructin is not possible since a unique absolute conic cannot be found. However, self-calibration might still be possible under a CMS. A motion sequence that **is** critical for self-calibration has an image ϕ of Φ that is different than ω. The difference between a critical sequence for Euclidean reconstruction and camera calibration is that Φ and Ω might have the same image ω in the general CMS [18].

There are two sets of potential absolute conics: ones that lie on the plane at infinity Π_∞ and ones that do on lie on Π_∞. For a potential absolute conic to lie on Π_∞ all relative rotations between pairs of images must conserve the eigenspaces of Φ [18]. This constraint leads to the following CMS: "Motion sequences for which all relative rotation are either rotation by an arbitrary amount about some line l or by 180° about lines perpendicular to l [18]". The potential absolute conics that do not lie on Π_∞ are more abstract and will not be described here.

Of all the CMS for self-calibration there are a couple that are practically more important than others. They include orbital motion; rotation about a parallel axes and arbitrary motion; planar motion; and pure translations [18]. Under all these circumstances, no algorithm can uniquely determine the camera calibration (i.e. singularities exist).

6.2 Problems with Kruppa's Equations

While the camera motions presented in 6.1 are critical for any self-calibration technique, Kruppa's equations suffer from other critical configurations. Furthermore, Kruppa's equations can be nummerically unstable. This section will discuss these two problems with Kruppa's Equations.

As pointed above inherent CMSs which cause a failure in the camera self-calibration for any method are due to the existence of a proper virtual conic, different from the absolute conic which are indentical in all views of the image sequence. Kruppa's equations are in a class of methods for self-calibration that do not enforce the planarity of the absolute conic[2] which thus lead to another class of CMSs[9].

Kruppa's equations are degenerate when the camera motion is along a sphere [9]. If C is the center of the viewing sphere, there are an infinite set of spheres, Φ, that are also centered at C. Since the camera is always pointed towards C, the projection of the sphere Φ remains identical, and can be misinterpreted as the absolute conic [9].

In this situation all intrinsic parameters except for the focal length are estimated correctly. The focal length is misestimated by a factor of

$$\frac{r}{\sqrt{r^2 - d^2}} \tag{23}$$

where r is the radius of Φ and d is the radius of the camera motion [9]. Since the focal length is usually the intrinsic parameter that is unknown and changing, the degeneracy in Kruppa's equations is quite important.

In many instances Kruppa's equations are solved by numerical analysis. Even when the camera motion is not a CMS but is close to the above CMS, Kruppa's equations can give large errors in the focal length. Experiments were performed by Sturm in [9] where

---

[2] The quasi-linear approach by Triggs is another method.

the camera motion was along a sphere. However a number of inaccuracies were added to move the camera motion away from the CMS. The inaccuraccies included changing the camera's orientation slightly such tht it no longer focused on C and translating the camera off the viewing sphere [9].

It was noted that while the prinicipal point was estimated with an error of less than 5%, the focal length was overestimated by close to 20% even when the offset in orientation was 10% of the radius or when there was one view off the viewing sphere. To reach the 5% error threshold, half of the views had to be off the viewing sphere or there had to be an offset in the orienttion of 20% of the radius [9]. Thus even when the camera motions are not CMS's Kruppa's equations can be numerically unstable.

6.3 Effect of Radial Distortion

One issue that has been unresolved in all the methods presented thus far for camera self-calibration is that of radial distortion. Radial distortion is the "displacement of an image point from the position predicted by the ideal pinhole camera model [19]". Radial distortion can be positive, which leads to a pin-cushion effect on the image, or negative, which leads to a barrelling effect.

When radial distortion exists in an image, the essential assumption, in camera self-calibration methods for rotating cameras, that 3D lines that connect matched features intersect at the same fixed pont in space is no longer valid [19]. Under pin-cushion distorion, the line intersections, are instead confined to a small region between the camera centre and the image, and thus the focal length is underestimated. When barrelling distortion exists in an image, the line intersections are not confined to a specific region. Instead the lines mostly intersect behind the camera centre and may even not intersect [19]. Thus with barrelling distorion the focal length is overestimated or may completely fail [19].

The reason for the failure of the self-calibration algorithms to be able to deduce the focal length when barrelling distortion is present is that barrelling distortion changes the projection of the scene unto the image to be more like speherical projection instead of perspective projection [19]. Spherical projection contains no information about the focal length or principial point [19].

In [19] the authors performed an experiment where the radial distortion is varied from 1×10^{-7} to -1×10^{-7}. As the radial distortion becomes closer to 0, the estimate of the focal length smoothly increases towards the correct focal length [19]. The standard deviation of the esimtates for positive radial distortion are also almost zero. However, when the distortion becomes negative, the self-calibration algorithms start to rapidly overestimate the focal length with large standard deviation [19]. Near a distorion of -8×10^{-8} the results become unreliable as the standard deviation of the estimates becomes too large [19].

[19] proposes a two stage self-calibration technique to deal with radial distortion in camera self-calibration. Firt, inter-image homographies are calculated where the radial

distortion is minimzed, and then calibration is performed.

7 Conclusion

This survery paper summarized a number of different methods for a camera to self-calibrate. Different methods each have their advantages and can be used under different circumstances. Many other methods also exist such as self-calibration with image triplets [3] and self-calibration in spite of varying intrinsic parameter [20][4]. This survery paper also presented some of the difficulties in self-calibration such as Critical Motion Sequences and the effect or radial distortion.

References

[1] Olivier D. Faugeras, Quang-Tuan Luong, and Stephen J. Maybank. Camera self-calibration: Theory and experiments. In *European Conference on Computer Vision*, pages 321–334, 1992.

[2] W. James MacLean. ECE1772-S03 lecture notes 1. January 2003.

[3] Martin Armstrong, Andrew Zisserman, and Richard I. Hartley. Self-calibration from image triplets. In *ECCV*, pages 3–16, 1996.

[4] Marc Pollefeys, Reinhard Koch, and Luc J. Van Gool. Self-calibration and metric reconstruction in spite of varying and unknown internal camera parameters. *International Journal of Computer Vision*, 32(1):7–26, August 1999.

[5] Peter Sturm. On focal length calibration from two views. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, volume II, pages 145–150, Dec 2001.

[6] P. Sturm. Critical motion sequences for the self-calibration of cameras and stereo systems with variable focal length. In *Procedings of 10th British Machine Vision Conference*, 1999.

[7] Marc Pollefeys and Luc Van Gool. Some geometric insight in self-calibration and critical motion sequences. Technical Report KUL/ESAT/PSI/0001, K.U. Leuven, Belgium, 2000.

[8] Richard I. Hartley. Estimation of relative camera positions for uncalibrated cameras. In *Proceedings of the European Conference on Computer Vision*, pages 579–587, 1992.

[9] P. Sturm. A case against kruppa's equations for camera self-calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(10):1199–1204, 2000.

[10] C. Zeller and O. Faugeras. Camera self-calibration from vide sequences: the kruppa equations revisited. Technical Report 2793, INRIA, INRIA Sophia-Antipolis, France, 1996.

[11] Bill Triggs. Autocalibration and the absolute quadric. In *Proceedings of the Conference on Computer Vision and Pattern Recogntion*, pages 609–614, June 1997.

[12] Richard I. Hartley. An algorithm for self calibration from several views. In *Proceedings of the Conferences on Computer Vision and Pattern Recognition*, pages 908–912, 1994.

[13] R. Hartley, R. Gupta, and T. Chang. Stereo from uncalibrated cameras. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 761–764, 1992.

[14] G. Newsam, D. Huynh, M. Brooks, and H. Pan. Recovering unknown focal lengths in self-calibration: An essentially linear algorithm and degenerate configurations. In *ISPRS*, volume 18, pages 575–580, 1996.

[15] P. Mendonca and R. Cipolla. A simple technique for self-calibration. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 500–505, 1999.

[16] Frederic Devernay and Olivier Faugeras. From projective to euclidean reconstruction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 264–269, June 1996.

[17] P. Gill, W. Murray, and M. Wright. *Practical Optimization.* Academic Press, 1981.

[18] Peter Sturm. Critical motion sequences for monocular self-calibration and uncalibrated euclidean reconstruction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1100–1105, 1997.

[19] B. Tordoff and D. Murray. Violating rotating camera geometry: The effect of radial distortion on Self-Calibration. In *ICPR*, pages 423–427, 2000.

[20] L. de Agapito, E. Hayman, and I. Reid. Self-calibration of a rotating camera with varying intrinsic parameters. In *Proceedings of the 9th British Machine Vision Conference*, pages 105–114, 1998.