# Recovery of Epipolar Geometry as a Manifold Fitting Problem

Liran Goshen, Ilan Shimshoni
Faculty of Industrial Engineering
Technion
Haifa, Israel 32000
lirang@techunix.technion.ac.il
ilans@ie.technion.ac.il

P. Anandan
Vision Technology
Microsoft Research
One Microsoft Way, Redmond
WA 98052-6399 USA
anandan@microsoft.com

Daniel Keren
Department of
Computer Science
Haifa University
Haifa, Israel 31905
dkeren@cs.haifa.ac.il

## Abstract

*The introduction of the joint image manifold allows to treat the problem of recovering camera motion and epipolar geometry as the problem of fitting a manifold to the data measured in a stereo pair. The manifold has a singularity and boundary, therefore care must be taken when fitting it.*

*This paper reviews the notion of joint image manifold, and how previous motion recovery methods can be viewed in its context, and then offers a new fitting method, which improves upon previous results, especially when the extent of the data and/or the motion are small.*

## 1. Introduction and Previous Work

Given a stereo pair with point correspondences, one seeks to recover the epipolar geometry, which encompasses the camera motion and 3D structure. This is a fundamental problem in computer vision, and there exists a huge body of research tackling it; see [5] for a thorough treatment. Due to the small space allotted to submissions, we will assume familiarity with such basic concepts as the essential and fundamental matrices and the focus of expansion (FOE). We also apologize for not being able to provide a more complete survey of previous work.

In the pioneering work [9], a simple algebraic relation between the corresponding points and the epipolar geometry was derived, which allows to recover the essential matrix given eight matching points in a stereo pair. We refer to this as the *direct solution*. In [16], it was assumed that more matching pairs are given, and that there are errors in the coordinates. In this scenario, the problem cannot be solved exactly as in [9], therefore one seeks an approximate solution by minimizing the sum of squares of the aforementioned algebraic relation. We will henceforth refer to this method by the commonly used name *algebraic method*.

More recent work has roughly followed two other directions:

- The *geometric method.* The idea here is to find a "legal" geometric configuration (i.e. one satisfying the epipolar geometry constraints), such that the sum of squared distances of the matching pairs from it is minimal. This problem is numerically more challenging, but it yields better results [5].
- The *ML (Maximum Likelihood) method*, sometimes referred to as the *Bayesian approach*. Here, the idea is to recover the epipolar geometry $G$ which maximizes the probability $Pr(G/D)$, where $D$ is the measured data (in this case the matching pairs). Some work in this direction is presented in [14, 6, 7, 13, 3].

In the rest of this paper, we will first introduce the relation between the four approaches describe here (direct, algebraic, geometric, ML), and the problem of manifold fitting. Then we shall pursue the ML method in some depth, and point out some scenarios in which it is advantageous compared to other methods. Lastly, some experimental work is presented.

## 2. The Joint Image Space and Manifold Fitting

The important notion of *joint image space* (JIS hereafter) [2, 15] allows an attractive interpretation of the epipolar geometry problem as a problem of fitting an algebraic variety. The JIS for a given epipolar configuration consists of the set of matching pairs which adhere to the epipolar geometry. One may work in projective or Euclidean space; we will use the latter, in which the JIS is a three dimensional manifold which happens to be an algebraic variety of order two [2].

The key observation in this paper is that since the JIS manifold is an algebraic variety, the JIS (and epipolar geometry) recovery problem reduces to the problem of fitting an algebraic variety, i.e. an implicit polynomial, to the data. While this idea is not new [2, 15], this work suggests to use a fitting method which in some cases performs better than previous work.

Proceedings of the Ninth IEEE International Conference on Computer Vision (ICCV 2003) 2-Volume Set
0-7695-1950-4/03 $17.00 © 2003 IEEE

## 2.1. Fitting Algebraic Varieties

Given a set of points $\{q_i\}$ in Euclidean space, one may seek a polynomial $p$ such that its zero set (i.e. the points in which it obtains a value of zero), approximates $\{q_i\}$ [8, 12]. Obviously, this is useful when one seeks a polynomial relation which has to be satisfied by some measured data – but this is exactly the situation we face when trying to recover the epipolar geometry! An explanation follows, as well as an interpretation of the four aforementioned methods as fitting techniques.

In [9], the following equation was derived: $(x_1, y_1, 1)E(x_2, y_2, 1)^t = 0$, where $E$ is the essential matrix and $\{(x_1, y_1), (x_2, y_2)\}$ a pair of matching points. This is a linear constraint on $E$'s elements, and if we look at the manifold (which is really the JIS) in 4D space defined by $(x_1, y_1, 1)E(x_2, y_2, 1)^t = 0$, the problem reduces to fitting such a manifold (defined by $E$) to the data. How should this be done? Let us proceed to review some methods and compare them to work done in the realm of epipolar geometry recovery.

- Direct solution. If it is assumed that no error is present in the data, it is possible to recover $E$ by directly solving the equations $(x_1, y_1, 1)E(x_2, y_2, 1)^t = 0$. Clearly, if eight pairs are available, there results a system of eight linear equations in eight variables [9]. Alas, usually the data is susceptible to measurement errors.

- Algebraic method – minimize the sum of squares of the constraints [16]. This is a common method for treating noisy data and the case in which there are more degrees of freedom in the data than in the model. In the context of algebraic variety fitting, this is equivalent to minimizing $\sum\limits_i p^2(q_i)$; but this is a notoriously weak method for fitting varieties [8, 12, 1].

- Geometric method. This is easily seen to be equivalent to fitting a variety by minimizing the sum of squared distances of the data from it. While computationally challenging, it yields better results [12, 1, 5].

- ML method. This was introduced in [17]. The idea is to recover the variety $V$, given the data $D = \{q_i\}$, by maximizing the probability $Pr(V/D)$ (here, $q_i$ will be a point in $\mathcal{R}^4$ obtained by concatenating two matching points in a stereo pair, and $V$ a variety defining the epipolar geometry, as will be explained shortly). Using Bayes' formula, this is usually taken to be proportional to $Pr(D/V) = \prod\limits_i Pr(q_i/V)$ (assuming no a-priori preference of one variety over the other, and considering that $D$ is fixed). But, as opposed to the geometric method – which assumes that $Pr(q_i/V)$ is proportional to $\exp\,(-\frac{d^2(q_i,V)}{2\sigma^2})$, where $\sigma$ is the noise variance and $d(q_i, V)$ the distance from $q_i$ to $V$ – the ML method seeks a more accurate estimate which uses the full probability distribution over $V$, which up to a normalizing factor equals

$$Pr(q_i/V) = \int\limits_V \exp(-\frac{d^2(q_i,v)}{2\sigma^2})dv \tag{1}$$

where the integration is with respect to the usual Lebesgue measure, which assigns identical measures to regions with identical area. In [17] it is shown that while the ML criterion is more complicated than the algebraic and geometric ones, it yields better results, especially when

- The variety is small with respect to the noise.

- The variety is strongly curved.

- The variety has a boundary.

In all these cases, $\exp(-\frac{d^2(q_i,V)}{2\sigma^2})$ is a poor approximation to Eq. 1, especially if there's data close to the singularity or the boundary. This possible pitfall was noted in [14], however it was assumed that the JIS is "locally linear", and it was proved (as in [17]) that in this case $\exp(-\frac{d^2(q_i,V)}{2\sigma^2})$ equals Eq. 1. However, as we shall demonstrate, the JIS is not locally linear, and therefore the ML method is expected to perform better, especially in scenarios in which there is data close to to the singularity or boundary of the JIS.

It should be noted that the geometric method can also be viewed as a maximum likelihood estimate – not of the manifold alone, but of the manifold and the "true" (denoised) sources of the measurement points simultaneously. The ML method used here alleviates this problem by integrating out the "true" points, yielding the correct probability of the manifold only.

# 3. The Cone

We now take a closer look at the cone which constitutes the JIS [2]. Using the well-known notion of the fundamental matrix $F$, the epipolar constraint can be written as $(x_1, y_1, 1)F(x_2, y_2, 1)^t = 0$ for matching points $(x_1, y_1), (x_2, y_2)$. It is also well-known that $F$ is of rank 2 (see [5] for discussion and references). Now follow a few lemmas:

**Lemma 1** *Under the transformation* $(x_1, y_1, x_2, y_2) \rightarrow (x_1 - a_1, y_1 - b_1, x_2 - a_2, y_2 - b_2)$, *where* $(a_1, b_1, 1)F = 0, F^t(a_2, b_2, 1)^t = 0^t$, *the fundamental matrix assumes the form*

$$F = \begin{pmatrix} F_{11} & F_{12} & 0 \\ F_{21} & F_{22} & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

*Note that this transformation is achieved simply by moving the origin of the left and right images to the FOE. The proof is immediate.*

**Lemma 2** *In the notation of Lemma 1, the constraint* $(x_1, y_1)F(x_2, y_2)^t$ *can be expressed as* $(x_1, y_1, x_2, y_2)F_4(x_1, y_1, x_2, y_2)^t = 0$*, where ([2])*

$$F_4 = \begin{pmatrix} 0 & 0 & F_{11} & F_{12} \\ 0 & 0 & F_{21} & F_{22} \\ F_{11} & F_{21} & 0 & 0 \\ F_{12} & F_{22} & 0 & 0 \end{pmatrix}$$

*The proof is immediate.*

**Lemma 3** *There is a rotation of coordinates such that if* $F_4$ *is in the form of Lemma 2, then* $(x_1, y_1, x_2, y_2)F_4(x_1, y_1, x_2, y_2)^t$ *is equal to*

$$(x_1^2 - y_1^2) + \lambda^2(x_2^2 - y_2^2) \tag{2}$$

*The proof follows simply by diagonalizing the* $4 \times 4$ *matrix of Lemma 2. It turns out that it has two pairs of eigenvalues with opposite signs,* $\pm\lambda_1, \pm\lambda_2$ *given by the following expressions:*

$$e_1 \equiv 2F_{11}^2 + 2F_{12}^2 + 2F_{21}^2 + 2F_{22}^2, e_2 \equiv F_{11}^4 + F_{12}^4 + F_{21}^4 + F_{22}^4 + 2F_{22}^2F_{21}^2 + 2F_{22}^2F_{12}^2 - 2F_{22}^2F_{12}^2 - 2F_{12}^2F_{21}^2 + 2F_{11}^2F_{21}^2 + 2F_{11}^2F_{12}^2 + 8F_{11}F_{12}F_{21}F_{22},$$

$$\lambda_1 = \frac{1}{2}\sqrt{e_1 + 2\sqrt{e_2}}, \lambda_2 = \frac{1}{2}\sqrt{e_1 - 2\sqrt{e_2}}$$

*We note, however, that the rotation required for diagonalizing* $F$ *is not separable in the images – i.e. it cannot be represented as a combination of separate rotations in* $(x_1, y_1)$ *and* $(x_2, y_2)$*, but it "mixes" all the four coordinates* $(x_1, y_1, x_2, y_2)$*. However, as far as the fitting is concerned, this makes no difference.*

Eq. 2 evidently describes a cone in $\mathcal{R}^4$. We next address the problem of fitting such a cone. Since the object being fit is not a function but a manifold, this is a problem often referred to as "errors in variables" [11]. We have adopted the solution offered in [17].

## 3.1. Integrating Over the Cone

Following the discussion on the ML method, in order to compute the probability of a candidate epipolar geometry given measurement data, we must first make the coordinate transformations indicated in Lemmas 1,2 (translation) and Lemma 3 (rotation), and then compute $\int\limits_C \exp\left(-\frac{d^2(q_i, v)}{2\sigma^2}\right)dv$, where $C$ is a cone as in Eq. 2 and $q_i$ a point in $\mathcal{R}^4$. In order to somewhat simplify this integral, define the following change of variables: $x_1 = \lambda r \cos(\theta_1), x_2 = r\sin(\theta_1), y_1 = \lambda r \cos(\theta_2), y_2 = r\sin(\theta_2)$. Note that this change of variables allows to define the manifold as "free" (i.e. without constraints), while reducing the number of variables to three. Under this change, the integral becomes

$$\int_0^{2\pi}\int_0^{2\pi}\int_0^{\infty} |J| \exp(-\frac{(\lambda r\cos(\theta_1) - X_1)^2 + (r\sin(\theta_1) - X_2)^2}{2\sigma^2} - \frac{(\lambda r\cos(\theta_2) - Y_1)^2 + (r\sin(\theta_2) - Y_2)^2}{2\sigma^2})dr d\theta_1 d\theta_2 \tag{3}$$

where $|J| = r^2\lambda\sqrt{2 + (\lambda^2 - 1)(\sin^2(\theta_1) + \sin^2(\theta_2))}$ is the Jacobian corresponding to the variable change, and $X_1, Y_1, X_2, Y_2$ are the (transformed) measured data. The internal integral (over $r$) can be explicitly computed, since it is $r^2$ times a Gaussian in $r$, and it therefore remains to numerically evaluate a double integral over the square $[0, 2\pi] \times [0, 2\pi]$. We have applied the Hermite and Legendre integration techniques to solve both this integral and similar ones over the "half-cones" discussed in Section 3.2.

## 3.2. Is it Necessary?

Clearly, the ML method is more computationally expensive than other methods, due to the numerical integration. When is it important to suffer this overhead? As noted in [17, 14], if the data is in a locally linear region of the cone, not much is gained by integrating. However, in the following two cases, the local linearity assumption is strongly violated.

- The (transformed) data points are close to the cone's apex, $(0, 0, 0, 0)$. Clearly, local linearity is violated. This can happen, for example, when the object is small, and the camera is moving towards it (as in tracking).

- For the sake of simplicity, assume for now that there is only camera translation present, and that it is forward or backward relative to the center of the scene, which we assume to be at the origin of the coordinate system. It is clear that if the matching point pairs are denoted $(p_1^{(i)}, p_2^{(i)})$, then either all the "true" $p_1^{(i)}$'s are closer to the origin than all the corresponding $p_2^{(i)}$'s, or vice-versa – the "order constraint".

  What does this mean, in terms of the manifold? If we disregard the order constraint, then the only restriction on the matching pairs (in the very simple scenario described above), is that each $p_2^{(i)}$ is the product of $p_1^{(i)}$ by a certain scalar. So, the corresponding cone is equal to

$$C = \{(x_1, y_1, \delta x_1, \delta y_1) | x_1, y_1 \in \mathcal{R}, \delta \in \mathcal{R}^+\}$$

However, the order constraint implies that the legal configurations of the "true" points (that is, the denoised measurement points) are in the union of the "half-cones" $C_1, C_2$, where

$$C_1 = \{(x_1, y_1, \delta x_1, \delta y_1) | x_1, y_1 \in \mathcal{R}, 0 \leq \delta \leq 1\}$$

and

$$C_2 = \{(x_1, y_1, \delta x_1, \delta y_1) | x_1, y_1 \in \mathcal{R}, 1 \leq \delta \leq \infty\}$$

note that $C_1, C_2$ are *manifolds with boundary*; the boundary of both is $\{(x_1, y_1, x_1, y_1) | x_1, y_1 \in \mathcal{R}\}$. When can there be data points close to the boundary? If the disparity between the matching points is large relative to the noise, then the noised "true points" will be close to each other (and hence to the boundary) only with low probability. However, if the motion is small (as can be the case in a video sequence), data will lie by the boundary.

What does this mean, intuitively? Suppose that the camera motion is forward, hence $p_2^{(i)}$ is farther from the origin than $p_1^{(i)}$. If we simply integrate over the entire cone, we are allowing *illegal* configurations in which $p_1^{(i)}$ is farther from the origin than $p_2^{(i)}$. If the disparity is small, these illegal configurations are assigned relatively high probabilities, as even a small noise can switch the order of the corresponding points.

In light of this, we have to integrate over $C_1$ and $C_2$, and multiply the resulting probabilities. Due to the space limitation, we do not discuss the numerical integration in this case. It should be clear that the problem of violating the order constraint for small motions occurs in all scenarios, not only the simple one discussed here.

### 3.3. A Simple Example

In order to demonstrate the importance of the ML method in the presence of measurement data near a singularity of the variety, we study a very simple case – a cone in $\mathcal{R}^2$, which consists of two straight lines: $y = ax, y = -ax$ for some $a$. Let us compare the implementation of the algebraic, geometric, and ML methods, and their performance.

Denote the lines $y = ax$ and $y = -ax$ by $L_1$ and $L_2$ respectively. Then clearly the cone $C$ is the union of $L_1$ and $L_2$. Its implicit equation is $(y - ax)(y + ax) = 0$. Following the previous study of the three methods, and noting that $\int\limits_{L_i} \exp(-\frac{d^2(q,L_i)}{2\sigma^2})dl = \exp\left(-d^2(q, L_i)\right)$ (this is easy to verify either from symmetry consideration or direct integration), one can conclude that the following cost functions have to be optimized for the various methods, in order to recover the optimal slope $a$ when given measured data $\{q_i = (x_i, y_i)\}$ (let us assume that the noise variance satisfies $2\sigma^2 = 1$):

- Algebraic method: minimize
$$\sum_i (y_i - ax_i)^2 (y_i + ax_i)^2$$
- Geometric method: minimize
$$\sum_i \min\{d^2(q_i, L_1), d^2(q_i, L_2)\}$$
- ML method: maximize
$$\sum_i \log(\exp(-d^2(q_i, L_1)) + \exp(-d^2(q_i, L_2)))$$

Note that for points far away from the origin (which is the cone's apex) the geometric method and ML criteria are nearly equivalent, since such points, being generated by adding noise to a cone point, will be much closer to $L_1$ than to $L_2$ or vice-versa (unless the slope is very large); in that case, one of the expressions $\exp(-d^2(q_i, L_1)), \exp(-d^2(q_i, L_2))$ is much smaller with respect to the other, hence the cost function will be well represented by $\min\{d^2(q_i, L_1), d^2(q_i, L_2)\}$. However, for points near the apex, that is not the case. The power of the ML method is that it does not "force" us to decide from what part of the cone – $L_1$ or $L_2$ – the point came from; both options are considered. Even in this simple case, the ML method yields better results than the geometric method, and both are far superior to the algebraic method – see Fig. 1. Note that there is no analog for the manifold's boundary in this simple case.

## 4. Experimental Results: Simulations

We performed several experiments to compare the ML method to the geometric method. The pure translation case as well as translation and rotation case (with known camera calibration) were studied. The mathematical details for pure translation and calibrated camera are in the same spirit as those for the fundamental matrix and are omitted. The average results are presented in all cases – there were cases in which the geometric method outperformed the ML method, but the ML method won on the average.

The distribution of the noise is $N(0, \sigma)$. In each experiment, 3D points were chosen at random in the common field of view. In each set of experiments, the epipolar geometry (FOE or essential matrix) was estimated 100 times with different instances of points and noise.

For each experiment sample the estimation error of the epipole and the rotation angle were calculated. The estimation error of the epipole was calculated as follows: Let $v$ be

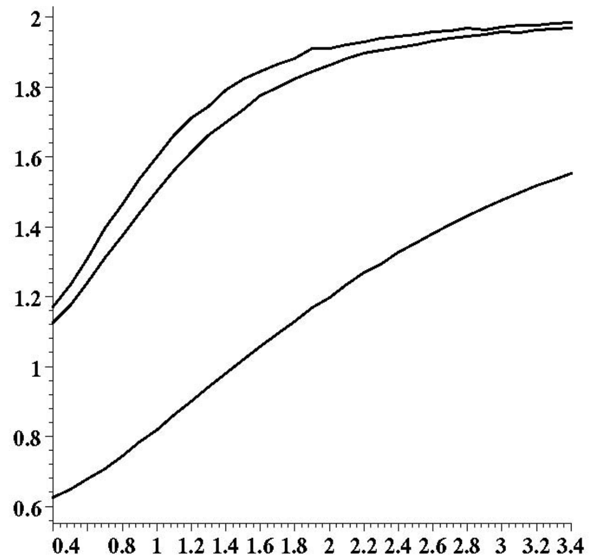

Figure 1: Simulation results of fitting a 1D cone to data generated by adding Gaussian noise of unit variance to a cone of slope 2. The $x$-axis represents the data's extent (meaning that it ranged between $-x$ and $x$), and the $y$-axis the average estimate of the slope – upper graph shows the ML results, middle graph the geometric method results, and lower graph the algebraic method results. When the extent of the data reaches 3.4 (which means that there are more data points away from the apex), both the ML and geometric methods converge to the correct slope, but for smaller extent of the data ML consistently performs better than geometric. The algebraic method performs very poorly. In all experiments, the same number of data points were used and the distribution was the same for the $y = x$ and $y = -x$ branches.

the "real" epipole point in the simulation and let $\hat{v}$ be the estimated epipole. Let $\vec{v} = (v_x, v_y, f)$ and $\vec{\hat{v}} = (\hat{v_x}, \hat{v_y}, f)$, where $f$ is the focal length. The estimation error is the angle $v_{err}$ between these two vectors. i.e., $v_{err} = \arccos(\frac{\hat{v} \cdot \vec{\hat{v}}}{|\hat{v}||\vec{\hat{v}}|})$. The estimation error of the rotation angle is the rotation angle corresponding to the rotation matrix $R_{err} = R^{-1}\hat{R}$, where $R$ is the "real" rotation matrix in the simulation and $\hat{R}$ is the estimated rotation matrix. The average errors are shown.

The numerical parameters we used in the simulations were as follows: The image size was 600 x 800 pixels, and the internal calibration matrix was

$$K = \begin{pmatrix} 1000 & 0 & 0 \\ 0 & 1000 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

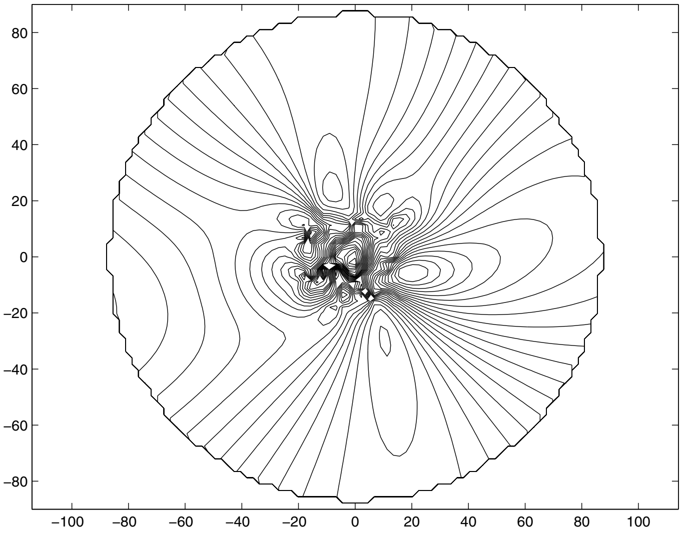

Figure 2: Likelihood function for the geometric method, 100 point correspondence. The correct FOE is at $(5, 5)$.

## 4.1. Pure Translation

First, the likelihood functions of the ML and geometric methods are compared. In this simulation there are 100 point correspondences, and a very small translation, resulting in a disparity of 1.8 pixels on the average. The simulated motion was nearly forward, with the (normalized) motion vector being equal to $(0.086, 0.086, 0.992)$. The noise was Gaussian with standard deviation 1. The likelihood functions are shown in Figs. 2-3 as contour maps. In Figs. 4-5 the parameters are similar to those in Figs. 2-3, but the motion has a stronger sideways component, and the correct FOE is at $(60, 5)$. As in the other simulation, the ML likelihood function is more stable and the location of its global minimum is closer to the correct location. Next, the estimates of the geometric and ML methods for the parameters corresponding to Figs. 2-3 are compared; the results are shown in Fig. 6, with the average error plotted as a function of the disparity between corresponding points. The ML method performs better, especially for small translations. The error is given in terms of the angle between the correct and estimated normalized translation vectors.

## 4.2. Calibrated Translation and Rotation

In this experiment the ML and geometric method were compared for translation and rotation (essential matrix recovery). There were 10 point correspondences. The translation was such that the mean disparity between the corresponding points, due to the translation alone, was 8 pixels. The rotation angle was 3 degrees. The results are presented in Figs. 7,8. As for the translation only case, the ML method outperforms the geometric method, especially when the noise

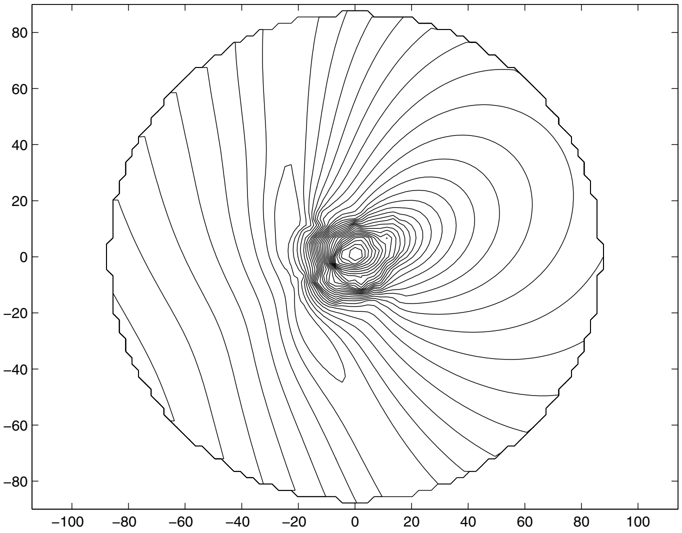

Figure 3: ML likelihood function for the scenario corresponding to Fig. 2. The function is much more stable and obtains its global minimum close to the correct location, $(5, 5)$.

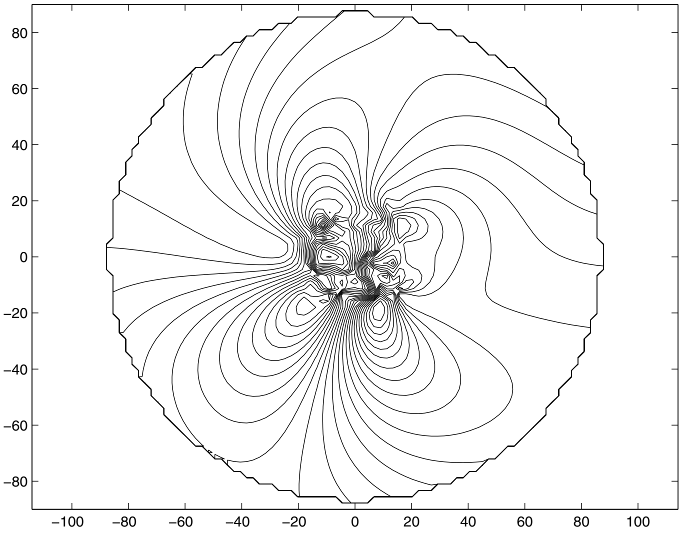

Figure 4: Likelihood function for the geometric method, with parameters similar to those in Fig. 2, except that the motion has a stronger sideways component and the correct FOE is at $(60, 5)$.

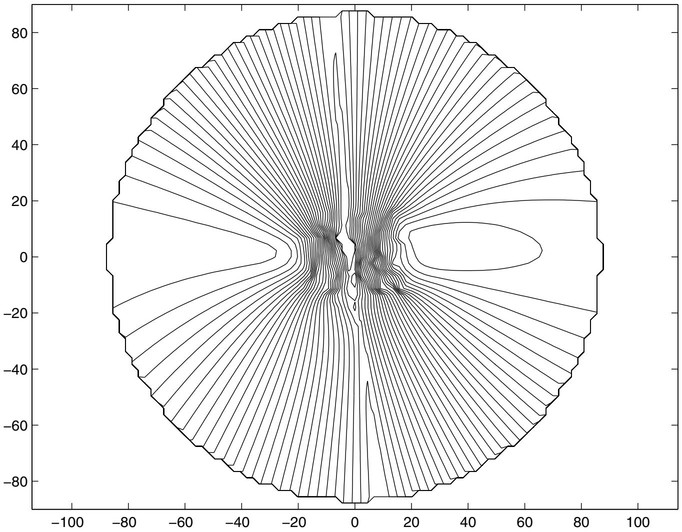

Figure 5: ML likelihood function for the scenario corresponding to Fig. 4. The function is much more stable and obtains its global minimum close to the correct location, $(60, 5)$.

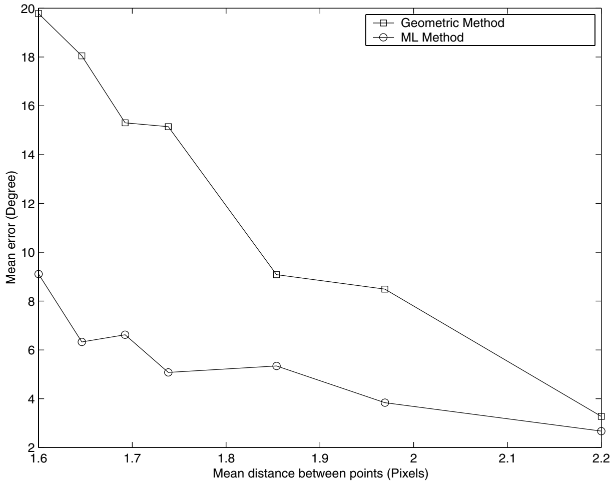

Figure 6: Performance of the ML and geometric methods for pure translation.

increases.

The ML method becomes more significant in the essential matrix case. It gives superior results also in configurations in which the noise is relatively small and translation is relatively large. In such configurations in the pure translation case the geometric and ML methods yield very similar results.

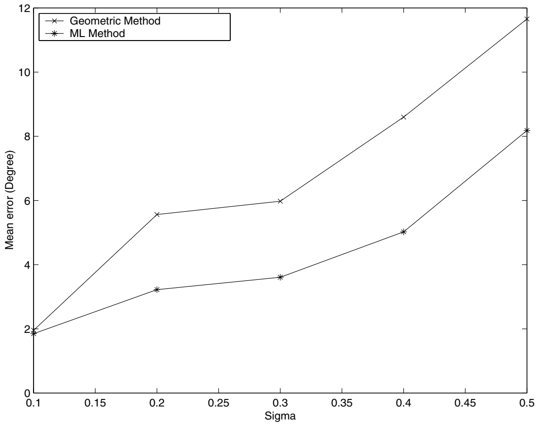

Figure 7: Results for the FOE in the calibrated rotation and translation case. Sigma is the standard deviation of the noise.

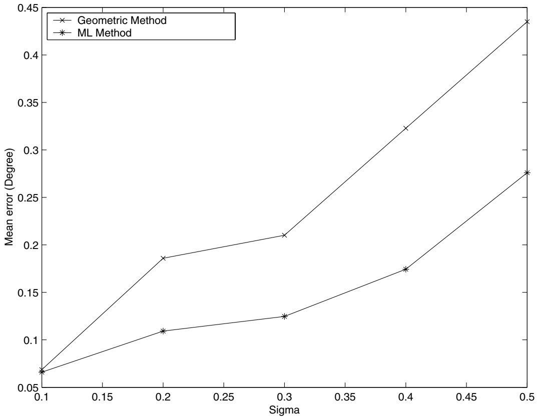

Figure 8: Results for the rotation angle in the calibrated rotation and translation case. Sigma is the standard deviation of the noise.

# 5. Experimental Results: Real Images

Due to space limitations, we show only one pair of real images. The image pair consists of two images of an office scene shown in Figs. 9,10. The camera motion was very small (a few centimeters), and the rotation angle 1 degree. The corners were recovered using the Harris corner detector [4].

When taking the entire field of view, and using 120 point correspondences, an accurate estimation to the camera motion was found using the geometric method, using the Nelder-Mead optimization method [10]; this was regarded as the ground truth. Then, the performance of the ML and geometric methods were tested on small random subsets of the matched pairs. Figs. 11,12 show the superior results achieved by the ML method for sets of corresponding pairs of sizes between 10 to 30.

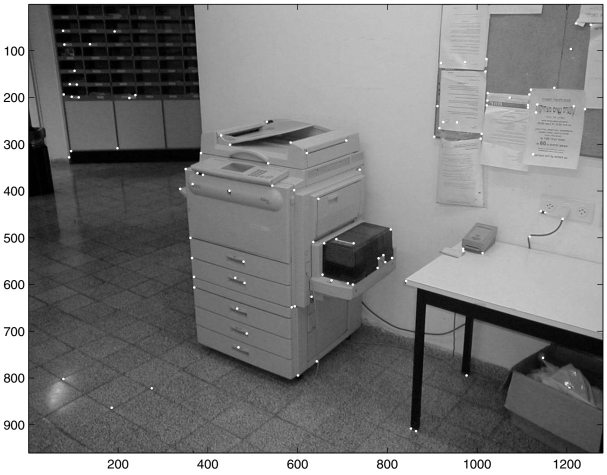

Figure 9: First picture of office scene, with matching points marked.

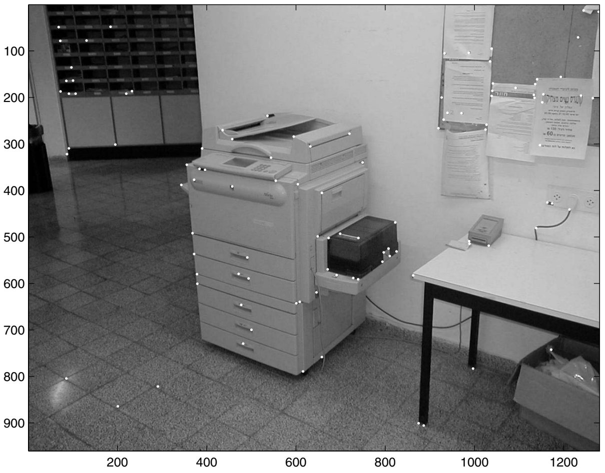

Figure 10: Second picture.

# 6. Summary and Conclusions

If the camera motion is small, and/or the objects are small relative to their distance from the camera, the ML method

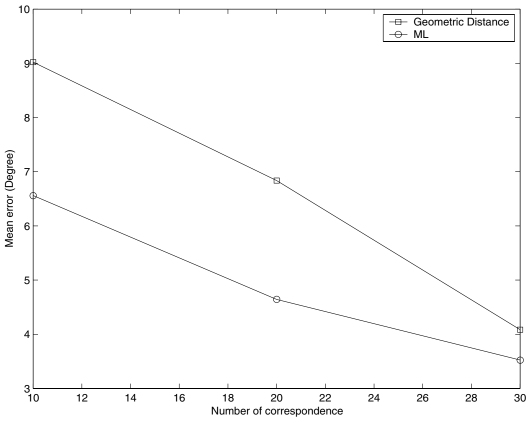

Figure 11: Error in FOE for ML and geometric methods, vs. the number of point correspondences.

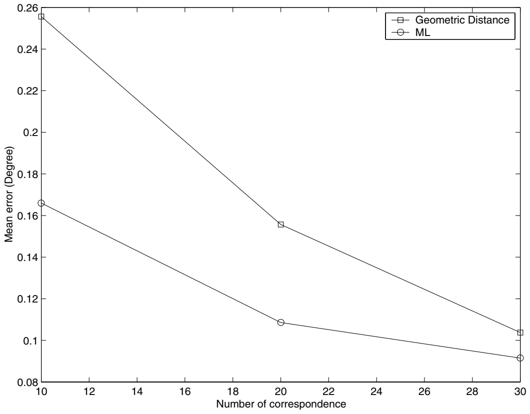

Figure 12: Error in rotation for ML and geometric methods, vs. the number of point correspondences.

has the potential to significantly improve on the geometric method. This is because the variety which represents the epipolar geometry has a singularity, and it is a manifold with boundary; hence the local linearity assumption, under which the geometric method is a reasonable approximation, may well be violated – since the points may in these cases be close to the singularity and to the manifold's boundary. The ML method can handle these situations better than the geometric method.

Planned future work includes further developing the numerical integration and optimization techniques, as well as extending the ideas presented here to the trilinear tensor.

## 7 Acknowledgments

We gratefully acknowledge the comments of the three anonymous ICCV referees.

## References

[1] S.J. Ahn, W. Rauh, and H.J. Warnecke. Least-squares orthogonal distances fitting of circle, sphere, ellipse, hyperbola, and parabola. *PR*, 34(12):2283–2303, December 2001.

[2] P. Anandan and S. Avidan. Integrating local affine into global projective images in the joint image space. In *European Conference on Computer Vision*, 2000.

[3] D.A. Forsyth, S. Ioffe, and J. Haddon. Bayesian structure from motion. In *ICCV99*, pages 660–665, 1999.

[4] C. Harris and M.J. Stephens. A combined corner and edge detector. In *Alvey88*, pages 147–152, 1988.

[5] R.I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge, 2000.

[6] K. Kanatani. Geometric computation for machine vision. In *Oxford University Press*, 1993.

[7] K. Kanatani. Statistical-analysis of geometric computation. *CVGIP*, 59(3):286–306, May 1994.

[8] D. Keren, D.B. Cooper, and J. Subrahmonia. Describing complicated objects by implicit polynomials. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 16(1):38–53, January 1994.

[9] H.C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135, September 1981.

[10] W.H. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling. *Numerical Recipes*. Cambridge University Press, 1986.

[11] G.A.F Seber and C.J Wild. *Nonlinear Regression*. John Wiley and Sons, 1989.

[12] G. Taubin, F. Cukierman, S. Sullivan, J. Ponce, and D.J. Kriegman. Parameterized families of polynomials for bounded algebraic curve and surface fitting. *PAMI*, 16(3 March 1994):287–303, March 1994.

[13] P. H. S. Torr and A. Zisserman. Concerning Bayesian motion segmentation, model averaging, matching and the trifocal tensor. In H. Burkharddt and B. Neumann, editors, *ECCV98 Vol 1*, pages 511–528. Springer, 1998.

[14] P.H.S. Torr. Bayesian model estimation and selection for epipolar geometry and generic manifold fitting. *IJCV*, 50(1):35–61, October 2002.

[15] B. Triggs. Matching constraints and the joint image. In *ICCV95*, pages 338–343, 1995.

[16] J. Weng, T.S Huang, and N.Ahuja. Motion and structure from two perspective views: Algorithms, error an alysis, and error estimation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 11:451–476, 1989.

[17] M. Werman and D. Keren. A Bayesian method for fitting parametric and nonparametric models to noisy data. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 23(5):528–534, May 2001.