# On Estimating the Uncertainty in the Location of Image Points in 3D Recognition from Match Sets of Different Sizes

Ilan Shimshoni
Industrial Engineering and Management Department
Technion, Haifa 32000, Israel

**Abstract**

Efficient and robust model-based recognition systems need to be able to estimate reliably and quickly the possible locations of other model features in the image when a match of several model points to image points is given. Errors in the sensed data lead to uncertainty in computed pose of the object which in turn lead to uncertainty in those positions. We present an efficient and accurate method for estimating these uncertainty regions. Our basic method deals with an initial match of three points. With a small additional computational cost it can be used to compute the uncertainty regions of the projection of many model points using the same match triplet. The basic method is then extended employing statistical methods to estimate the uncertainty region when given initial matches of any size. This is the major practical contribution of the paper because when the number of points in the match increases the size of the uncertainty region decreases dramatically which helps to discriminate much better between correct and incorrect matches in model-based recognition algorithms.

# 1 Introduction

Model-based object recognition systems usually involve estimating the pose of the object using a small set of matched features. The hypothesis is then verified by using the pose to find additional matches. These systems perform the operations of pose estimation and estimation of the location of other features many times during the recognition process. Therefore efficient methods for computing them must be developed for these systems themselves to be efficient. For example, in the alignment recognition method presented by Huttenlocher and Ullman [7], the pose of the object is estimated from a match of triples of model and image points and used to predict the approximate image location of other points. If there are $m$ model points and $n$ image points, $O(m^3n^3)$ pose estimations will be computed and for each pose $m$ estimations of the position in the image of other points will be computed. Therefore it is critical for the performance of this and similar algorithms to find efficient methods for pose estimation and computation of the uncertainty region in the image of other model points.

Another important factor which is critical for the performance of recognition algorithms is the quality of the estimate of the uncertainty region. The more exact the estimate, the lower the chances for the projection of a model point not to be within the uncertainty region (false negative). A good estimate of the position of projection of the model point will result in a smaller uncertainty region. This lowers the chances for a point in the image to be within the uncertainty region of an incorrect match (false positive), which might cause an incorrect match to be considered correct, requiring a costly verification process. Therefore a method which produces good estimates with small exact uncertainty regions will improve the efficiency of recognition algorithms considerably.

Even with an improved method quite a few incorrect matches will be considered correct due to the relatively large uncertainty regions. To prune out most of these false positives, image points which lie within the uncertainty region of the projection of other model points can be added to the match and new uncertainty regions of other points which are based on the new larger match can be computed. These uncertainty regions

are considerably smaller (by a factor of 10 on average) reducing the chances for a false positive. The verification stage will only be left with a handful of cases to check.

In this paper we will consider the "weak-perspective" imaging model. We use this model because it is simpler than the full perspective model and because it is a good estimate of the perspective model when the object is relatively far from the camera relative to its size.

Several methods for estimating the pose of the object from a match of three model points to image points have been presented [1, 2, 4, 7, 10]. Methods for estimating the uncertainty regions have also been presented. Several methods estimate the pose uncertainty region [4, 5, 10] which can then be used to estimate the uncertainty regions in the image, others estimate directly the uncertainty regions in the image [2, 3, 8]. Our method belongs to the latter.

In [3], the three uncertainty circles around the three image points in the match are uniformly sampled and for each combination of sampled points the position of the fourth point is calculated. The results of these experiments show that the true uncertainty regions tend to be circular. They also show that a very coarse sampling of only 8 points of each circle (i.e. 512 pose computations) is needed to achieve a very good estimate of the radius of the circle.

In [2], the uncertainty region is estimated by a linear approximation. When the given match has more than three points, linear constraints on the uncertainty region are derived and the uncertainty region is estimated using linear-programming.

In [5], the pose uncertainty region is estimated for a point triplet. For each additional point the uncertainty region is reduced using a Kalman filter. From the resulting pose uncertainty region it is possible to compute the uncertainty region of a projection of a point.

Our method for estimating the uncertainty regions is based on the pose estimation method presented by Alter [1] which is very efficient and has an elegant geometric interpretation. The main advantages of our method is that without any additional cost it is able to produce smaller uncertainty regions which reduce the probability for false

matches, and has a higher probability for the image point to be within the uncertainty region. The method is then extended to deal with matches of more than three points efficiently using a statistical framework. Using larger matches reduce considerably the size of the uncertainty region and show considerable improvement on previously published results. This extended method can be used to prune out most of the false positives at a small computational cost.

The rest of the paper is organized as follows. We review Alter's pose estimation method in Section 2. We present our method for estimation of the uncertainty regions using a three point match in Section 3, and extend our method to larger sets of matches in Section 4. Finally a number of issues raised by our algorithm and future research directions are discussed in Section 5.

# 2 Geometric Pose Estimation

In this section we will present a brief overview of Alter's geometric pose estimation method. For details please refer to [1]. The geometric setting underlying the weak-perspective three point pose estimation problem is shown in Figure 1. The picture shows the three model points being projected orthographically to the plane that contains $\mathbf{m}_0$ and is parallel to the image plane, and then shows them being scaled down by scale factor $s$ into the image. Let the distances between the model points be $R_{01}$, $R_{02}$ and $R_{12}$, and the corresponding distances between the image points be $d_{01}$, $d_{02}$ and $d_{12}$. Also let

$$
\begin{aligned}
a &= (R_{01} + R_{02} + R_{12})(-R_{01} + R_{02} + R_{12})(R_{01} - R_{02} + R_{12})(R_{01} + R_{02} - R_{12}) \\
b &= d_{01}^2(-R_{01}^2 + R_{02}^2 + R_{12}^2) + d_{02}^2(R_{01}^2 - R_{02}^2 + R_{12}^2) + d_{12}^2(R_{01}^2 + R_{02}^2 - R_{12}^2) \\
c &= (d_{01} + d_{02} + d_{12})(-d_{01} + d_{02} + d_{12})(d_{01} - d_{02} + d_{12})(d_{01} + d_{02} - d_{12}) \\
\sigma &= \begin{cases} 1 & \text{if } d_{01}^2 + d_{02}^2 + d_{12}^2 \leq s^2(R_{01}^2 + R_{02}^2 + R_{12}^2), \\ -1 & \text{otherwise.} \end{cases}
\end{aligned}
$$

Then the unknown parameters in Figure 1 are

$$
\begin{aligned}
s &= \sqrt{\frac{b + \sqrt{b^2 - ac}}{a}} \\
(h_1, h_2) &= \pm \left( \sqrt{(sR_{01})^2 - d_{01}^2}, \sigma \sqrt{(sR_{02})^2 - d_{02}^2} \right) \\
(H_1, H_2) &= \frac{1}{s}(h_1, h_2).
\end{aligned} \tag{1}
$$

For each image of three points there are two poses which yield that image. (1) yields two pairs of values for $h_1$ and $h_2$ which correspond to those two poses.

Given image points $\mathbf{i}_0 = (x_0, y_0), \mathbf{i}_1 = (x_1, y_1)$, and $\mathbf{i}_2 = (x_2, y_2)$, the 3D locations of the model points in camera-centered coordinates are:

$$\mathbf{m}_0 = \frac{1}{s}(x_0, y_0, w), \tag{2}$$

$$\mathbf{m}_1 = \frac{1}{s}(x_1, y_1, h_1 + w), \tag{3}$$

$$\mathbf{m}_2 = \frac{1}{s}(x_2, y_2, h_2 + w). \tag{4}$$

# 3 Uncertainty Region Estimation

We now turn to the first topic of this paper, estimating the region in the image in which the projection of a fourth model point $\mathbf{m}_3$ will be located. We model errors by assuming that the detected feature point is within $\epsilon$ pixels from the location of the true point. The main problem is to try to estimate the effects of the uncertainty in the image locations of $\mathbf{i}_0, \mathbf{i}_1$, and $\mathbf{i}_2$ on the estimated location of the fourth point $\mathbf{i}_3$. Using a first order approximation, the maximum displacement occurs when the errors of the three points lie on the circle of radius $\epsilon$. Higher order terms may cause the position of the maximum displacement to move slightly from the borders of the $\epsilon$-disks, but we will ignore their effect in our approximation since as we tested experimentally their effect on the size of the uncertainty region is minimal. As mentioned above the true uncertainty regions tend to be circular therefore we will try to find a combination of errors in the projections of

the three model points which will yield the largest displacement in the $x$ coordinate of $\mathbf{m}_3$ and use that as our estimate for the radius of the uncertainty region.

In [1] it is shown that the coordinates of $\mathbf{m}_3$ can be expressed as a function of the coordinates of $\mathbf{m}_0$, $\mathbf{m}_1$, and $\mathbf{m}_2$ by solving the following vector equation for the "extended affine coordinates", $(\alpha, \beta, \gamma)$, of $\mathbf{m}_3$.

$$\mathbf{m}_3 = \alpha(\mathbf{m}_1 - \mathbf{m}_0) + \beta(\mathbf{m}_2 - \mathbf{m}_0) + \gamma(\mathbf{m}_1 - \mathbf{m}_0) \times (\mathbf{m}_2 - \mathbf{m}_0) + \mathbf{m}_0. \tag{5}$$

Substituting (2-4) into (5) yields that the image location $\mathbf{i}_3$ of $\mathbf{m}_3$ is

$$\begin{aligned} \mathbf{i}_3 = (&\alpha(x_1 - x_0) + \beta(x_2 - x_0) + \gamma((y_1 - y_0)H_2 - (y_2 - y_0)H_1) + x_0, \\ &\alpha(y_1 - y_0) + \beta(y_2 - y_0) + \gamma(-(x_1 - x_0)H_2 + (x_2 - x_0)H_1) + y_0). \end{aligned} \tag{6}$$

We will use this expression for the position of the image location of $\mathbf{m}_3$ to estimate the uncertainty region. Let $\boldsymbol{\epsilon}_0, \boldsymbol{\epsilon}_1, \boldsymbol{\epsilon}_2$ be the error vectors for points $\mathbf{i}_0, \mathbf{i}_1, \mathbf{i}_2$ respectively. We will compute the first order approximation of the size of the uncertainty region. Such an estimate requires replacing each image coordinate by the measured value plus the error and substitute any value which depends on image coordinates ($H_1$ and $H_2$) by their value and plus their derivatives with respect to the image coordinates multiplied by the error in the corresponding coordinate. The derivation will be done in two steps. In the first step we will disregard the changes to $H_1$ and $H_2$ due to perturbations in the positions of the image points and add this effect in the second stage. Substituting the perturbed points into (6) and subtracting the unperturbed image position of $\mathbf{m}_3$ yields:

$$\begin{aligned} &(\boldsymbol{\epsilon}_0 \cdot (1 - \alpha - \beta, \gamma(H_1 - H_2)) + \boldsymbol{\epsilon}_1 \cdot (\alpha, \gamma H_2) + \boldsymbol{\epsilon}_2 \cdot (\beta, -\gamma H_1), \\ &\boldsymbol{\epsilon}_0 \cdot (-\gamma(H_1 - H_2), 1 - \alpha - \beta) + \boldsymbol{\epsilon}_1 \cdot (-\gamma H_2, \alpha) + \boldsymbol{\epsilon}_2 \cdot (\gamma H_1, \beta)), \end{aligned} \tag{7}$$

for the uncertainty vector in the image.

Let

$$\begin{aligned}
\Sigma_0 &= (1-\alpha-\beta, \gamma(H_1 - H_2)), \\
\Sigma_1 &= (\alpha, \gamma H_2), \\
\Sigma_2 &= (\beta, -\gamma H_1), \\
\Sigma_0^\perp &= (-\gamma(H_1 - H_2), 1-\alpha-\beta), \\
\Sigma_1^\perp &= (-\gamma H_2, \alpha), \\
\Sigma_2^\perp &= (\gamma H_1, \beta).
\end{aligned}$$

Using this notation (7) is reduced to

$$(\boldsymbol{\epsilon}_0 \cdot \Sigma_0 + \boldsymbol{\epsilon}_1 \cdot \Sigma_1 + \boldsymbol{\epsilon}_2 \cdot \Sigma_2, \boldsymbol{\epsilon}_0 \cdot \Sigma_0^\perp + \boldsymbol{\epsilon}_1 \cdot \Sigma_1^\perp + \boldsymbol{\epsilon}_2 \cdot \Sigma_2^\perp). \tag{8}$$

Under the constraint $||\boldsymbol{\epsilon}_0|| = ||\boldsymbol{\epsilon}_1|| = ||\boldsymbol{\epsilon}_2|| = \epsilon$, and neglecting for now the fact that $H_1$ and $H_2$ are not constants, the $\boldsymbol{\epsilon}_i$'s which maximize the $x$ coordinate of (8) are

$$\boldsymbol{\epsilon}_i = \epsilon \frac{\Sigma_i}{||\Sigma_i||}.$$

As $|\Sigma_i| = |\Sigma_i^\perp|$ and $\Sigma_i \cdot \Sigma_i^\perp = 0$, the $y$ coordinate of the displacement is zero. It is easy to show that when the $\boldsymbol{\epsilon}_i$'s computed above are all rotated by any angle $\theta$ the displacement stays the same, yielding a circular uncertainty region. The probability distribution of the projection of the fourth point is a linear combination of the distribution of the three circles. When they are distributed uniformly this yields a distribution similar to a bivariate Gaussian whose mean is the unperturbed projection of the fourth point. This is due to the fact that a linear combination of uniform distributions converges to a Gaussian distribution as the number of distributions increases. If the error in the three image points had a Gaussian distribution then a linear combination of the distributions would also be a Gaussian (a linear combination of Gaussians is a Gaussian).

When the changes in $H_1$ and $H_2$ due to perturbations in image coordinates are relatively small, the deviations of the uncertainty region from a circle are minor. The larger the perturbations of $H_1$ and $H_2$ the less circular the uncertainty region becomes. This will

happen when $H_1$ and $H_2$ are relatively small compared to their derivatives with respect to the coordinates of the three points. This happens for example when the plane in which the three points lie is close to parallel to the image plane ($H_1$ and $H_2$ are very small) or when the projections of the three points are very close to each other (the derivatives are very large).

We will now add the effects of the changes in $\mathbf{i}_0, \mathbf{i}_1$ and $\mathbf{i}_2$ on $H_1$ and $H_2$. Let $\boldsymbol{\xi}$ denote the vector of the six coordinates of $\mathbf{i}_0, \mathbf{i}_1$ and $\mathbf{i}_2$. We can evaluate the derivatives

$$\frac{\partial H_i}{\partial \boldsymbol{\xi}_j} \quad i = 1 \cdots 2 \;\; j = 1 \cdots 6,$$

by computing directly the derivatives of the equations in Section 2, or by estimating them using finite differences at a cost of six additional pose estimations. We use these derivatives for the following first order approximation of $H_1$ and $H_2$

$$H_i(\boldsymbol{\xi} + \Delta\boldsymbol{\xi}) = H_i(\boldsymbol{\xi}) + \sum_{j=1}^{6} \Delta\boldsymbol{\xi}_j \frac{\partial H_i}{\partial \boldsymbol{\xi}_j} + o(\epsilon^2).$$

Substituting the $H_i$'s in (6) by their first order approximation computed above, we get a first order approximation for the $x$ and $y$ coordinates of $\mathbf{m}_3$. To the coefficients of the $j$'th component of $\Delta\boldsymbol{\xi}$ shown in (8) we add

$$\gamma(y_1 - y_0)\frac{\partial H_2}{\partial \boldsymbol{\xi}_j} - \gamma(y_2 - y_0)\frac{\partial H_1}{\partial \boldsymbol{\xi}_j} \quad \text{and} \quad -\gamma(x_1 - x_0)\frac{\partial H_2}{\partial \boldsymbol{\xi}_j} + \gamma(x_2 - x_0)\frac{\partial H_1}{\partial \boldsymbol{\xi}_j}, \quad (9)$$

respectively. The coefficients of $\Delta\boldsymbol{\xi}$ are now

$$\begin{aligned}
\Sigma_0 &= (1 - \alpha - \beta + \gamma(y_1 - y_0)\tfrac{\partial H_2}{\partial x_0} - \gamma(y_2 - y_0)\tfrac{\partial H_1}{\partial x_0}, \\
&\quad \gamma(H_1 - H_2) + \gamma(y_1 - y_0)\tfrac{\partial H_2}{\partial y_0} - \gamma(y_2 - y_0)\tfrac{\partial H_1}{\partial y_0}), \\
\Sigma_1 &= (\alpha + \gamma(y_1 - y_0)\tfrac{\partial H_2}{\partial x_1} - \gamma(y_2 - y_0)\tfrac{\partial H_1}{\partial x_1}, \gamma H_2 + \gamma(y_1 - y_0)\tfrac{\partial H_2}{\partial y_1} - \gamma(y_2 - y_0)\tfrac{\partial H_1}{\partial y_1}), \\
\Sigma_2 &= (\beta + \gamma(y_1 - y_0)\tfrac{\partial H_2}{\partial x_2} - \gamma(y_2 - y_0)\tfrac{\partial H_1}{\partial x_2}, -\gamma H_1 + \gamma(y_1 - y_0)\tfrac{\partial H_2}{\partial y_2} - \gamma(y_2 - y_0)\tfrac{\partial H_1}{\partial y_2}), \\
\Sigma_0^\perp &= (-\gamma(H_1 - H_2) - \gamma(x_1 - x_0)\tfrac{\partial H_2}{\partial x_0} + \gamma(x_2 - x_0)\tfrac{\partial H_1}{\partial x_0}, \\
&\quad 1 - \alpha - \beta - \gamma(x_1 - x_0)\tfrac{\partial H_2}{\partial y_0} + \gamma(x_2 - x_0)\tfrac{\partial H_1}{\partial y_0}), \\
\Sigma_1^\perp &= (-\gamma H_2 - \gamma(x_1 - x_0)\tfrac{\partial H_2}{\partial x_1} + \gamma(x_2 - x_0)\tfrac{\partial H_1}{\partial x_1}, \alpha - \gamma(x_1 - x_0)\tfrac{\partial H_2}{\partial y_1} + \gamma(x_2 - x_0)\tfrac{\partial H_1}{\partial y_1}), \\
\Sigma_2^\perp &= (\gamma H_1 - \gamma(x_1 - x_0)\tfrac{\partial H_2}{\partial x_2} + \gamma(x_2 - x_0)\tfrac{\partial H_1}{\partial x_2}, \beta - \gamma(x_1 - x_0)\tfrac{\partial H_2}{\partial y_2} + \gamma(x_2 - x_0)\tfrac{\partial H_1}{\partial y_2}).
\end{aligned} \quad (10)$$

However as $\Sigma_i \cdot \Sigma_i^{\perp} \neq 0$, the estimate of the uncertainty region is no longer perfectly circular.

Using (10) we can compute a first order approximation of the radius of the uncertainty region in a way similar to [2]. As $\epsilon$ is not infinitesimal, higher order terms must be included in our estimate. We evaluate $\mathbf{i}_3$ using as the image positions $\mathbf{i}_0 + \boldsymbol{\epsilon}_0, \mathbf{i}_1 + \boldsymbol{\epsilon}_1$, and $\mathbf{i}_2 + \boldsymbol{\epsilon}_2$, which according to our first order approximation are supposed to produce the maximum displacement from the position of $\mathbf{i}_3$ computed using $\mathbf{i}_0$,$\mathbf{i}_1$, and $\mathbf{i}_2$. The distance between the two estimates of $\mathbf{i}_3$ is used as our estimate for the radius of the uncertainty region.

Our estimation method never overestimates the real radius of the uncertainty region because it is the result of a perturbation computed for a point triplet which are within their respective uncertainty regions.

One of the main applications of this method is in alignment algorithms[7]. In that case given a certain match uncertainty, regions for the projections of many model points have to be computed. Using our method all that needs to be done is to compute the derivatives of $H_1$ and $H_2$ once and then substitute into (8) and (9) the different extended affine coordinates $(\alpha, \beta, \gamma)$ of each point and compute the size of the uncertainty region at the cost of one pose estimation.

## 3.1 Experimental Results

In order to test our method we compare its results to the method presented in [2] and a method we think will produce close to the true uncertainty region but with a high computational cost.

In this method we use a nonlinear minimization algorithm [9] which uses the perturbed points obtained by our method as the starting position for the algorithm. The algorithm tries to find points on the three uncertainty circles which maximize the uncertainty in the position of the projection of $\mathbf{m}_3$ in the image. Because in many cases the initial estimates provided by our method are close to optimal, the number of pose evaluations that the algorithm needs to perform is relatively low and on average requires only 50 pose evaluations.

In order to test quality of the estimates we have to compare the false positive and false negative rates of the various methods. This is done by repeating the following experiment which has been used in [2] to test their method. Sets of four model points are randomly generated and projected orthographically to a $1000 \times 1000$ square image. The image points are then perturbed uniformly within a circle of radius five pixels. Using the first three pairs of model and image points, the uncertainty region for the fourth model point is estimated. We check to see if the fourth image point is within our uncertainty region. When the image point is found, we record the size of region in which we have been looking.

The performance of recognition algorithms which would use our algorithm depend on the false positive and false negative rates of our method. We therefore let the designer of the recognition algorithm have a choice between several values of false positive and false negative rates to choose from. We therefore multiply the results of our estimation algorithms by a user supplied parameter $\lambda$ and use the result as the estimate. The larger the value of $\lambda$ the higher the false positive rate and the lower the false negative rate. The designers of recognition algorithms will be able to choose the value of $\lambda$ which produces the most efficient algorithm.

We plotted a graph for the success rate and average uncertainty area for the method presented in [2], our method and the minimization method in Figure 2(a). To analyze this graph we have to check the "price" that has to be paid in the average area of the uncertainty region for a given success rate and vice versa. From this analysis we can see that the minimization algorithm performs the best for very low false negative rates. However due to its higher computation cost it is more important to compare the performance of our method to the performance of the method presented in [2]. For low false negative rates ($< 0.005$) it outperforms our method but for higher false negative rates our method performs better producing smaller uncertainty regions.

At first glance these results seem puzzling because that even though our method has an additional step in the algorithm to estimate the effect of higher order terms it does not always outperform [2] which uses only a first order approximation. After carefully

studying the results we conclude that in some rare cases ($< 0.005$) the perturbed point for which the projection of $\mathbf{m}_3$ was measured was misleadingly close to the unperturbed projection due to the effect of high-order terms yielding a small estimate of the size of the uncertainty region. This problem can be easily solved by computing the perturbation on an additional perturbed set by rotating the error vectors $\boldsymbol{\epsilon}_i$ by a constant angle $\theta$ which as mentioned above should also produce as good an estimate as the previous point where with high probability the higher order terms will not have a detrimental affect on the estimate of the uncertainty region.

In Figure 2(b) we re-plot the results of [2] and compare them to our method with one and two sets of perturbed points tested and the size of the uncertainty region taken as the maximum perturbation of the computed projections. As can be seen from the graph adding a second set improves the quality of the results yielding the best results when very low false negative rates are required. However, for higher false negative rates the method with one perturbed set is best. The only issue left to address is the issue of computational costs of the various algorithms and from our tests the computational costs of our algorithms are approximately the same as [2].

One of the strengths of the method presented in [2] is that it has been extended to deal with sets of matches of any size using linear programming. In the next section we will employ statistical analysis to extend our method (and the method presented in [2]) to deal also with larger sets of matches.

# 4 $n$th-Point Uncertainty Region

It has been shown [2, 4] that recognition algorithms which use a small number of features to estimate the pose of the object encounter large numbers of false positive matches due to the uncertainty in the pose which leads to large uncertainty regions in which the wrong image features may lie. Therefore it is important to derive methods to compute more exact poses based on more information which yield smaller uncertainty regions.

Before we start describing the algorithm consider the following typical example. Given

a match of size 4, we would like to estimate the uncertainty region of a fifth point. Before doing so we would like to compute what is the conditional probability distribution function of the position of this fifth point, whose true position is at $(0,0)$ in the image, given the **measured** position of three or four points in the image. In Table 1 and Figure 3 we show these conditional distributions and their means and variances. These distributions were computed using Alter's pose estimation algorithm described in Section 2 without using our uncertainty computation algorithm. The fifth conditional distribution which is based on evidence from all four points measured in the image has a better estimate for the position of the fifth point a smaller variance a higher peak and a smaller spread than the other four conditional distributions which are based on evidence of only three points. All distributions look similar to truncated Gaussian distributions. In our algorithm we would like to use the first four probability distributions for which can compute an estimate for their mean and their variance using the technique described in the previous section to compute an estimate for the mean and variance of the fifth probability distribution which is based on the cumulative evidence of all four points.

An estimate for the projection of $\mathbf{m}_n$ which is linear in the image coordinates of the measured points in the image, can be derived by extending (5) as follows:

$$\mathbf{m}_n = \sum_{0 \le i < j < n} \alpha_{ij}(\mathbf{m}_j - \mathbf{m}_i) + \sum_{0 \le i < j < k < n} \gamma_{ijk}(\mathbf{m}_j - \mathbf{m}_i) \times (\mathbf{m}_k - \mathbf{m}_i) + \sum_{0 \le i < n} \delta_i \mathbf{m}_i, \tag{11}$$

where $\alpha_{ij}, \gamma_{ijk}, \delta_i$ are unknown coefficients. Repeating the derivation preformed in the previous section will yield an estimator for the projection of $\mathbf{m}_n$. As in the previous section, when the changes in $H_i, \{0 < i < n\}$ are neglected, the distribution of the projection of $\mathbf{m}_n$ which results from this equation is circular symmetric and looks similar to a truncated Gaussian. When the changes in $H_i, \{0 < i < n\}$ are significant the distribution may become less circular. In order to ensure that the mean of this distribution is the projection of $\mathbf{m}_n$, the coefficients $\alpha_{ij}, \gamma_{ijk}, \delta_i$ must be chosen such that (11) holds. For $n = 3$ there is a unique set of coefficients $(\alpha, \beta, \gamma, 1)$ for which (11) holds. For $n > 3$ there are infinitely many such estimates and we would like to use the most efficient estimator (i.e., the one with the smallest variance). Finding that estimate is computationally very

expensive. We will therefore use the weighted average of equations of the type (5) (3-point estimates) which produces the minimal variance. We hope that the variance of this estimator will not be far from the minimal variance. For every weighted average (11) will hold. All that is required is to find the weights which minimize the variance. Using the same weights, the estimate of the projection of $\mathbf{m}_n$ is computed as the weighted average of the 3-point estimates.

Consider the example of a 4 point match shown in Figure 4. For each of the four matches of size three we can compute the uncertainty region. Note that the size of the uncertainty regions vary considerably (therefore using a constant size uncertainty region for an estimate of a three point match would produce a very poor estimate). For each of the four estimates we have the estimate of the position of the projection of the fifth point (denoted by a dot) and the radius of the uncertainty region which is inversely correlated with the certainty of the estimate. We would like to combine these four estimates into a single estimate of the uncertainty region giving a bigger weight to estimates with smaller uncertainty regions.

Formally speaking, given a match of size $n$ there are $\binom{n}{3}$ different matches of size three. For each one of them we can use our method (or the method presented in [2]) to estimate the position of the $n+1$'th point and estimate the radius of the uncertainty region. Putting this into a statistical context, we have $\binom{n}{3}$ random variables whose mean is the position of the $n+1$'th point and whose variance is approximately proportional to the area of the uncertainty region for which we have an estimate.

For each one of these random variables we have a single measurement computed from the measurements of its corresponding three points. The maximum likelihood estimate of the mean of the variable is that value and we assume that the variance we estimated for these perturbed measurements is very close to the variance of the unperturbed measurements had we been able to measure them.

Our goal is to combine all these estimates into a single estimate with the smallest possible variance (i.e. smallest uncertainty region).

A theorem in statistics [6] states the following.

Let $X_1, \cdots, X_m$ denote random variables that have means $\mu_1, \cdots, \mu_m$ and variances $\sigma_1^2, \cdots, \sigma_m^2$. Let $\rho_{ij}$, $i \neq j$, denote the correlation coefficient of $X_i$ and $X_j$ and let $K_1, \cdots, K_m$ denote real constants. The mean and the variance of the linear function

$$Y = \sum_{i=1}^{m} K_i X_i,$$

are respectively

$$\mu_Y = \sum_{i=1}^{m} K_i \mu_i,$$

and

$$\sigma_Y^2 = \sum_{i=1}^{m} K_i^2 \sigma_i^2 + 2 \sum\sum_{i<j} K_i K_j \rho_{ij} \sigma_i \sigma_j. \tag{12}$$

Our goal is to find the values for $K_1, \cdots, K_m$ yielding a linear combination of the $m$ variables which have the minimal variance. In our case all our variables have the same unknown mean (the projection of the $n+1$ model point) for which we have estimates. For $Y$ to have the same mean $\sum_{i=1}^{m} K_i$ must equal to 1. When $X_1, \cdots, X_m$ are mutually stochastically independent the minimal variance for $Y$ is achieved when

$$K_i = \frac{1}{\sigma_i^2} \frac{1}{\sum_{i=1}^{m} \frac{1}{\sigma_j^2}},$$

which as can be seen the weight of each variable is inversely proportional to its variance. However, as the match triples are usually not disjoint the $X_i$'s are not independent, therefore these values for the $K_i$'s will not be optimal.

In order to analyze the problem of finding the optimal weights $\mathbf{K}$ consider the following observations:

- Each of the $\binom{n}{3}$ estimation regions $X_i$ is approximately a linear combination of six coordinates where each of them has a variance of $\epsilon^2/4$ and each pair of them have a correlation coefficient $\rho_{ij}$ zero.

- The uncertainty region of each $X_i$ is a linear combination of three approximately circular uncertainty regions producing an approximately circular uncertainty region.

- A linear combination of the $X_i$'s when reordered by the projections of each model point is approximately a linear combination of $n$ circles (each due to one projected model point in the image) and is therefore also close to circular. Therefore we can limit ourselves to finding weights $\mathbf{K}$ that minimize the extent of the $x$ coordinate of the uncertainty region.

Using our linear approximation of the $x$ coordinate of the image position of the $n+1$'th point from a match of three points developed in Section 3 we can estimate the variance of the $i$'th estimate using (12). We shall denote by $\mu_{\boldsymbol{\xi}}$ the unknown image projection of the $n$ model points and by $\Delta\boldsymbol{\xi}$ their perturbations. The sought after $x$ coordinate of the projection of the $n+1$'th point will be denoted by $\mu_x$. For the $i$'th estimate let $M_i$ be a vector of coefficients of the coordinates $\boldsymbol{\xi}$, all but six of which are zero, such that if $X_i$ is the $i$'th estimate

$$X_i(\boldsymbol{\xi}) = X_i(\mu_{\boldsymbol{\xi}} + \Delta\boldsymbol{\xi}) \approx M_i(\mu_{\boldsymbol{\xi}}) + M_i\Delta\boldsymbol{\xi} = \mu_x + M_i\Delta\boldsymbol{\xi}.$$

The only problem with this model is that the $X_i$'s are not exactly linear functions of the coefficients $\boldsymbol{\xi}$ and thus our model for the variance of the linear combination of the $X_i$'s is not exact either. We have computed for each of the $X_i$'s the contribution of the high order terms to the radius of the uncertainty region in the final stage of our three point match algorithm. We therefore model the contribution of these high order term to the variance by an additional variable for each $X_i$ which we will denote by $u_i$ and whose variance is $\sigma_{ui}^2$. As we have performed only a first order analysis we are not able to determine the correlation coefficients between these variables and the other variables participating in the analysis (the coordinates of the measured points in the image and the other $u_i$'s).

The variance of the sum of weighted independent variables is found from the first term of (12) to be the sum of the squares of the weights times the variance of each variable. Using the notation described above and adding in the variance of the new variable, we get

$$\sigma^2_{X_i} = \sum_{l=1}^{2n} (\frac{\epsilon}{2} M_{i,l})^2 + \sigma^2_{ui} = \frac{\epsilon^2}{4} |M_i|^2 + \sigma^2_{ui}. \tag{13}$$

Similarly given two such variables $X_i$ and $X_j$, $Cov(X_i, Xj) = \rho_{ij}\sigma_i\sigma_j$ can be found from (12) to be the sum of the products of the weights multiplied by the variance, or using our notation

$$\begin{aligned} Cov(X_i, X_j) &= \frac{1}{2}(\sigma^2_{X_i+X_j} - \sigma^2_{X_i} - \sigma^2_{X_j}) \\ &\approx \frac{1}{2}(\frac{\epsilon^2}{4}|M_i + M_j|^2 + \sigma^2_{ui} + \sigma^2_{uj} - (\frac{\epsilon^2}{4}|M_i|^2 + \sigma^2_{ui}) - (\frac{\epsilon^2}{4}|M_j|^2 + \sigma^2_{uj})) \\ &= \sum_{l=1}^{2n} (\frac{\epsilon}{2} M_{i,l})(\frac{\epsilon}{2} M_{j,l}) = \frac{\epsilon^2}{4} M_i \cdot M_j. \end{aligned} \tag{14}$$

Generalizing this analysis for the weighted linear combination of the $X_i$'s yields

$$Y = \sum_{i=1}^{m} K_i X_i = \sum_{i=1}^{m} K_i X_i(\mu_{\boldsymbol{\xi}}) + \sum_{i=1}^{m} K_i M_i \Delta\boldsymbol{\xi}.$$

Writing this in matrix form where the matrix $M = (M_1 \cdots M_m)$ and $\mu_x = X_i(\boldsymbol{\xi})$ yields

$$Y = \mu_x + M\mathbf{K}\Delta\boldsymbol{\xi} + \sum_{i=1}^{m} K_i u_i,$$

where $MK$ is the vector of coefficients of the coordinates $\boldsymbol{\xi}$. We then plug (13) and (14) into (12) yielding

$$\sigma^2_Y = \sum_{i=1}^{m} \frac{\epsilon^2}{4} |M_i K_i|^2 + 2 \sum\sum_{i<j} \frac{\epsilon^2}{4} K_i K_j M_i \cdot M_j + \sum_{i=1}^{m} K_i^2 \sigma^2_{ui}, = \frac{\epsilon^2}{4} |M\mathbf{K}|^2 + \sum_{i=1}^{m} K_i^2 \sigma^2_{ui}. \tag{15}$$

The $u_i$'s can be easily added into the matrix $M$ by adding an additional $m$ variables where each $u_i$ appears only in its corresponding $M_i$. We now have to find a $\mathbf{K}$ that minimizes (15) under the linear constraint $\sum_{i=1}^{m} K_i = 1$. $K_1$ is eliminated from (15) by substituting it with $1 - \sum_{i=2}^{m} K_i$ yielding

$$\sigma^2_Y = \frac{\epsilon^2}{4} \left| ((M_2 - M_1) \cdots (M_m - M_1)) \begin{pmatrix} K_2 \\ \vdots \\ K_m \end{pmatrix} - (-M_1) \right|^2. \tag{16}$$

We shall denote by

$$A = ((M_2 - M_1) \cdots (M_m - M_1)) \quad \text{by} \quad \mathbf{x} = \begin{pmatrix} K_2 \\ \vdots \\ K_m \end{pmatrix} \quad \text{and by} \quad \mathbf{b} = (-M_1)$$

Thus (16) can be written in the new notation as

$$\min_{\mathbf{x}} \frac{\epsilon^2}{4} (A\mathbf{x} - \mathbf{b})^2,$$

which is the formulation of a standard linear least squares problem which is solved exactly using standard numerical methods [9]. Initially it might seem that the $K_i$'s are not bounded. They could have arbitrarily large negative or positive values (as long as their sum equals 1) which would magnify the small error in our linear model of the $X_i$'s and increase $\sigma_Y^2$. But as we have modeled the contribution to the variance of the high order terms by the $u_i$'s the solution we get deals with this problem also. This can be seen by examining (15) and seeing that the $\sigma_{ui}$'s act as a penalties against large negative or positive $K_i$'s. When the $u_i$'s are not incorporated into the model (even though their variance is very small) the problem of the unbounded $K_i$'s is very significant yielding considerably worse results.

Note that although we demonstrated this method on image points whose original uncertainty is uniformly distributed, the same results are obtained for other spherically symmetric distributions such as the Gaussian distribution. All that has to be changed is the term $\frac{\epsilon^2}{4}$ which has to be substituted with the variance of the new distribution and in the estimate of the variance of each of the $X_i$'s. The technique can also be used when other techniques are used to estimate the linear approximation of a point from a 3 point to combine several such estimates.

The value obtained for $\sigma_Y$ and the vector $\mathbf{K}$ are returned by the algorithm. The center of the uncertainty region is computed by the $\mathbf{K}$-weighted linear combination of the centers of the $\binom{n}{3}$ uncertainty regions. The user can choose the value of a parameter $\lambda$ which is the radius of the uncertainty region in standard deviations.

The results for match sizes four,five, and six are traced in Figures 5,6 and 7 respectively for several values of $\lambda$. We also plot the percent of projected points within the uncertainty region as a function of $\lambda$. From these plots its clear that choosing value $\lambda$ between 2.5 and 3.0 will yield a 0% false negative rate which is compatible with what can be expected for a quasi-Gaussian distribution. We plot the results presented in [2] and our results. In the results presented in [2], the uncertainty regions are bounded by rectangles. Using the same method but by bounding them with polygons with more than four sides, a more accurate estimate could be achieved, but would require additional runs of a linear program. To demonstrate the quality of our results consider the following example. In order for 98% of the uncertainty regions to contain the image of the model point for a match of size four, the method presented in [2] requires that the average area of the uncertainty region be 1750 pixels, where our method requires an average area of only 480 pixels. Our method never requires an average area of 1750 pixels, because it reaches 100% success rate for an average area of 880 pixels.

It is hard to compare our method to [5] which does not compute projection uncertainty regions but pose uncertainty regions. Another difference between the methods is that their method is incremental in the size of the match where as ours looks at all combinations of matches of size three and therefore will use (implicitly) combinations which produce small uncertainty regions. It has not been shown that the incremental method will produce the same quality of results irrespective of the order of the match and therefore when an initial match of size three is given which produces a large uncertainty region it will probably produce worse results then if the initial match produced a small uncertainty region. The method presented in [2] also has this characteristic.

In Figure 8 we plot the size of the uncertainty region as a function of the size of the match for several values of the success rate. This graph which is plotted on a logarithmic scale demonstrates how dramatically the size of the uncertainty region shrinks as the size of the match increases.

Finally we have tested our system on an image of a real object whose corners we have measured by hand. In Figure 9, the uncertainty regions are overlayed over the image.

We assume that $\epsilon = 7.7$ pixels and chose $\lambda = 3.0$. At first we computed the uncertainty regions of all the points using a match of size three. We then added additional points to our match and computed the uncertainty regions for the remaining points. We continue this process until all the points measured in the image are part of the match. The figure demonstrates how the uncertainty decreases significantly when the size of the match increases.

# 5 Discussion and Future Work

We have presented an efficient method for estimating the uncertainty region of the projection of a model point in the image when a match of three other model points is given. We then incorporated our method into a general statistical framework to deal with the case when more points are matched. Our method estimates the uncertainty regions better than previously published methods by finding the optimal linear combination of three point estimates which produces a very good estimate for the position of the projected model point, and a very small variance. An interesting question to pursue is whether to produce a good estimate all three point combination uncertainty regions have to be estimated or a subset of these combinations will suffice.

Future work will be dedicated to using this method in recognition systems. The false negative and false positive rates of our method can be chosen by the user of our method by choosing the value of $\lambda$. Therefore, further analysis has to be performed to find what combination of them and which variant of the method will make the recognition system most efficient. Another important question to be considered is how to extend a three point basis by adding to it additional model and image points yielding an efficient method to prune the largest amount of false positive matches at the smallest computational cost.

# Acknowledgments

This work was supported in part by the Koret Foundation.

# References


[1] T. D. Alter. 3D pose from 3 points using weak-perspective. *IEEE Trans. Patt. Anal. Mach. Intell.*, 16(8):802–808, August 1994.

[2] T. D. Alter and D. W. Jacobs. Error propagation in full 3D-from-2D object recognition. In *Proc. IEEE Conf. Comp. Vision Patt. Recog.*, pages 892–898, Seattle, Washington, June 1994.

[3] T.D. Alter and W.E.L. Grimson. Fast and robust 3D recognition by alignment. In *Proc. Int. Conf. Comp. Vision*, pages 113–120, 93.

[4] W. E. L. Grimson, D. P. Huttenlocher, and T. D. Alter. Recognizing 3D objects from 2D images; an error analysis. In *Proc. IEEE Conf. Comp. Vision Patt. Recog.*, pages 316–321, Champaign, Illinois, June 1992.

[5] Y. Hel-Or and M. Werman. Absolute orientation from uncertain point data: A unified approach. In *CVPR92*, pages 77–82, 1992.

[6] R.V. Hogg and A.T. Craig. *Introduction to Mathematical Statistics.* Macmillan Publishing, New York, 1978.

[7] D. Huttenlocher and S. Ullman. Recognizing 3D solid objects by alignment with an image. *Int. J. of Comp. Vision*, 5(2):195–212, 1990.

[8] D.W. Jacobs. Optimal matching of planar models in 3D scenes. In *Proc. IEEE Conf. Comp. Vision Patt. Recog.*, pages 269–274, 91.

[9] D. Kahaner, C. Moler, and S. Nash. *Numerical Methods and Software.* Prentice Hall, Englewood Cliffs, NJ, 1989.

[10] I. Shimshoni and J. Ponce. Probabilistic 3D object recognition. In *Proc. Int. Conf. Comp. Vision*, pages 488–493, Boston, Massachusetts, June 1995.

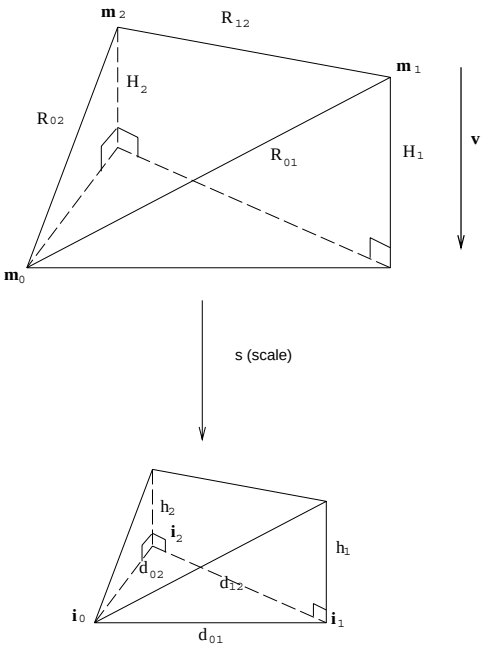


Figure 1: Model points $\mathbf{m}_0$, $\mathbf{m}_1$, and $\mathbf{m}_2$ undergoing orthographic projection plus scale to produce image points $\mathbf{i}_0$, $\mathbf{i}_1$ and $\mathbf{i}_2$ (adapted from [1]).

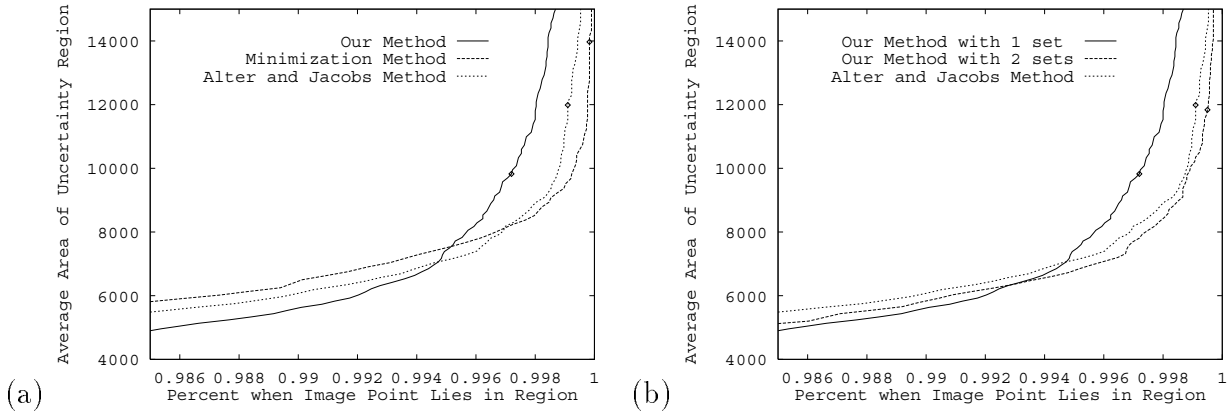


Figure 2: Plots where three points are given and the position of the fourth point is estimated. The graphs show the percentage of the times that the fourth point's image shows up in the predicted error region and the average area of the uncertainty region. The $\diamond$s on the graph denote the value for $\lambda = 1$. (a) Comparing our method with one perturbed set to the minimization method and the method presented in [2]. (b) Comparing our method with one and two perturbed sets to the method presented in [2].

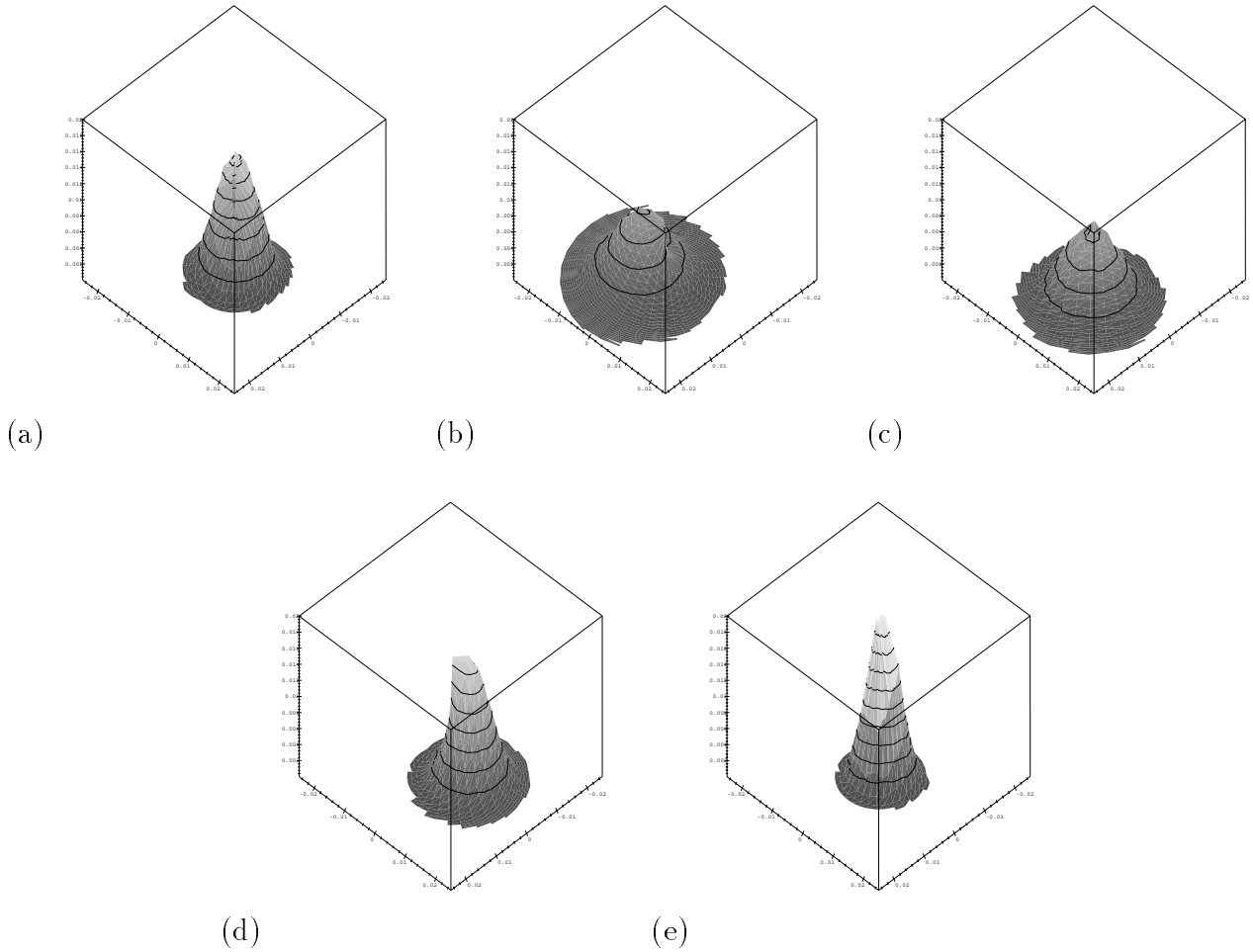

Figure 3: The five conditional probability distributions for the position of the fifth point based on different evidence. (a) (1,2,3); (b) (1,2,4); (c) (2,3,4); (d) (1,3,4); (e) (1,2,3,4). Note that the fifth distribution has a better estimate of the center, a higher peak and a smaller spread then the other four distributions.

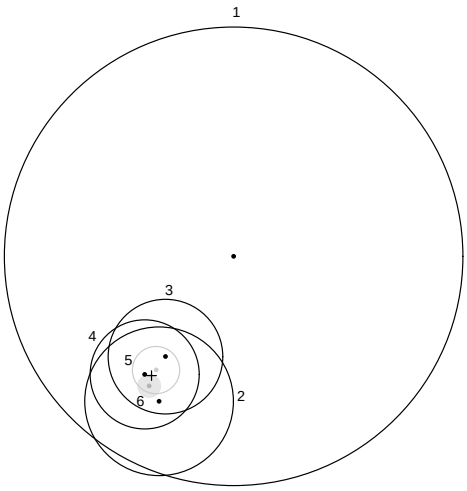


Figure 4: Given a four point match the four uncertainty regions of the four point triples (marked by 1-4), the estimated uncertainty region (1 standard deviation) (5), the true uncertainty region of the fifth point due to measurement errors (6), and the position of the measured fifth point in the image marked by a +.

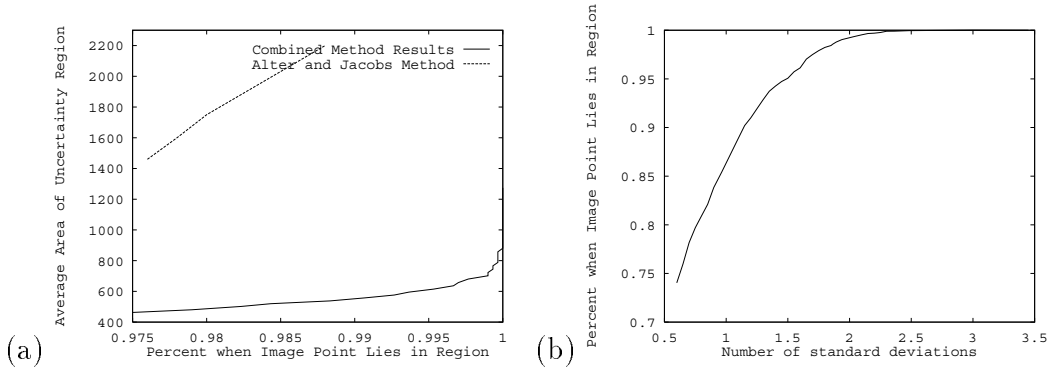


Figure 5: (a) A plot where four points are given and the position of the fifth point is estimated. The graph shows the percentage of times that the fifth point's image shows up in the predicted error region and the average area of the uncertainty region. (b) The value of $\lambda$ (the number of standard deviations) plotted against the percent of projected points which lie within the uncertainty region.

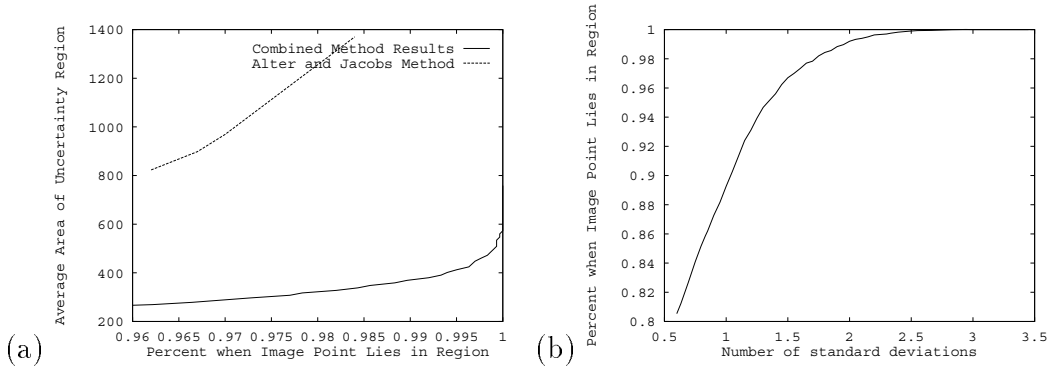


Figure 6: (a) A plot where five points are given and the position of the sixth point is estimated. The graph shows the percentage of times that the sixth point's image shows up in the predicted error region and the average area of the uncertainty region. (b) The value of $\lambda$ (the number of standard deviations) plotted against the percent of projected points which lie within the uncertainty region.

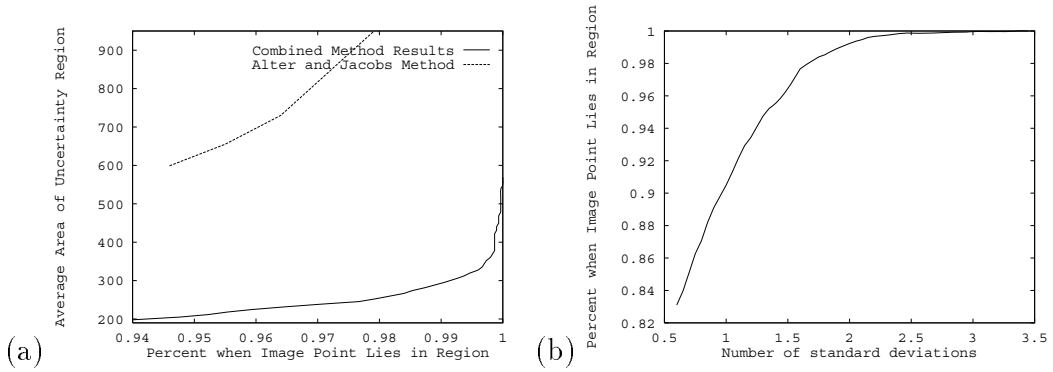


Figure 7: (a) A plot where six points are given and the position of the seventh point is estimated. The graph shows the percentage of times that the seventh point's image shows up in the predicted error region and the average area of the uncertainty region. (b) The value of $\lambda$ (the number of standard deviations) plotted against the percent of projected points which lie within the uncertainty region.

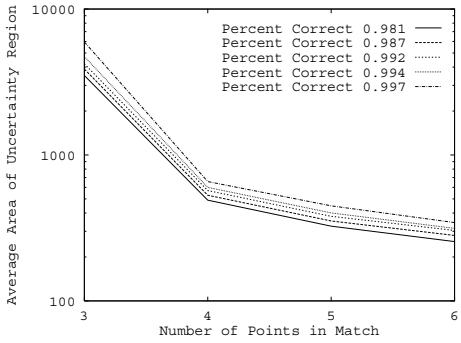


Figure 8: For a given probability for the point in the image to be in the uncertainty region, the average area of the uncertainty region is shown in logarithmic scale for different sizes of sets of matched points.

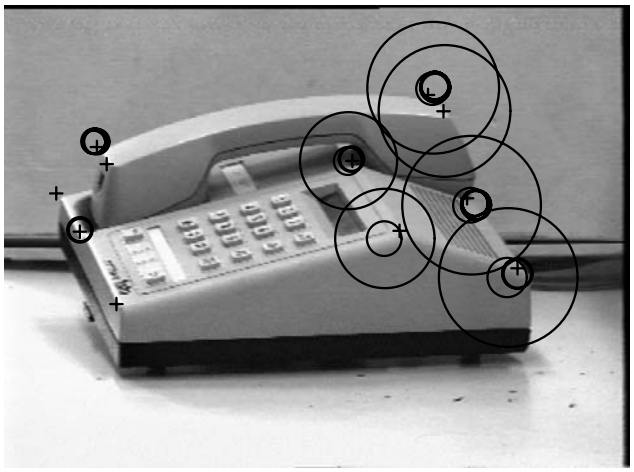

Figure 9: Uncertainty regions shrink as more points are added to the match. The larger circles are computed when only three points are matched. As more points are added the uncertainty regions shrink but still contain the measured point.

| Points used | Estimated Position | $\sigma_x$ | $\sigma_y$ | $\sigma_x\sigma_y$ |
|---|---|---|---|---|
| 1,2,3 | (-0.00236, -0.00176) | 0.0041 | 0.0041 | 1.71e-5 |
| 1,2,4 | ( 0.00333, -0.00491) | 0.0061 | 0.0061 | 3.71e-5 |
| 2,3,4 | ( 0.00482, 0.00399) | 0.0056 | 0.0056 | 3.13e-5 |
| 1,3,4 | (-0.00232, 0.00403) | 0.0044 | 0.0044 | 1.90e-5 |
| 1,2,3,4 | (-0.00038, 0.00066) | 0.0034 | 0.0033 | 1.10e-5 |

Table 1: For each match of size three or four, the conditional probability distribution of the position of fifth point (whose true position is $(0, 0)$) is computed. The table shows the evidence from which points participated in the computation of the distribution, the estimated position of the fifth point, the standard deviations in the $x$ and $y$ direction and their product which is proportional to the area of the uncertainty region. Note that the fifth distribution has a better estimate for the position of the fifth point and a smaller variance than the other distributions based on three points.

# Figure Legends

1. Model points $\mathbf{m}_0$, $\mathbf{m}_1$, and $\mathbf{m}_2$ undergoing orthographic projection plus scale to produce image points $\mathbf{i}_0$, $\mathbf{i}_1$ and $\mathbf{i}_2$ (adapted from [1]).

2. Plots where three points are given and the position of the fourth point is estimated. The graphs show the percentage of the times that the fourth point's image shows up in the predicted error region and the average area of the uncertainty region. The $\diamond$s on the graph denote the value for $\lambda = 1$. (a) Comparing our method with one perturbed set to the minimization method and the method presented in [2]. (b) Comparing our method with one and two perturbed sets to the method presented in [2].

3. The five conditional probability distributions for the position of the fifth point based on different evidence. (a) (1,2,3); (b) (1,2,4); (c) (2,3,4); (d) (1,3,4); (e) (1,2,3,4). Note that the fifth distribution has a better estimate of the center, a higher peak and a smaller spread then the other four distributions.

4. Given a four point match the four uncertainty regions of the four point triples (marked by 1-4), the estimated uncertainty region (1 standard deviation) (5), the true uncertainty region of the fifth point due to measurement errors (6), and the position of the measured fifth point in the image marked by a +.

5. (a) A plot where four points are given and the position of the fifth point is estimated. The graph shows the percentage of times that the fifth point's image shows up in the predicted error region and the average area of the uncertainty region. (b) The value of $\lambda$ (the number of standard deviations) plotted against the percent of projected points which lie within the uncertainty region.

6. (a) A plot where five points are given and the position of the sixth point is estimated. The graph shows the percentage of times that the sixth point's image shows up in the predicted error region and the average area of the uncertainty region. (b)

The value of $\lambda$ (the number of standard deviations) plotted against the percent of projected points which lie within the uncertainty region.

7. (a) A plot where six points are given and the position of the seventh point is estimated. The graph shows the percentage of times that the seventh point's image shows up in the predicted error region and the average area of the uncertainty region. (b) The value of $\lambda$ (the number of standard deviations) plotted against the percent of projected points which lie within the uncertainty region.

8. For a given probability for the point in the image to be in the uncertainty region, the average area of the uncertainty region is shown in logarithmic scale for different sizes of sets of matched points.

9. Uncertainty regions shrink as more points are added to the match. The larger circles are computed when only three points are matched. As more points are added the uncertainty regions shrink but still contain the measured point.