5

Algorithm Evaluation and Error Analysis

This chapter describes methods for assessing and quantifying the results of estimation algorithms. Often it is not sufficient to simply have an estimate of a variable or transformation. Instead some measure of confidence or uncertainty is also required.

Two methods for computing this uncertainty (covariance) are outlined here. The first is based on linear approximations and involves concatenating various Jacobian expressions. The second is the easier to implement Monte Carlo method.

5.1 Bounds on performance

Once an algorithm has been developed for the estimation of a certain type of transformation it is time to test its performance. This may be done by testing it on real or on synthetic data. In this section, testing on synthetic data will be considered, and a methodology for testing will be sketched.

We recall the notational convention:

- A quantity such as $\mathbf{x}$ represents a measured image point.
- Estimated quantities are represented by a hat, such as $\hat{\mathbf{x}}$ or $\hat{\mathtt{H}}$.
- True values are represented by a bar, such as $\bar{\mathbf{x}}$ or $\bar{\mathtt{H}}$.

Typically, testing will begin with the synthetic generation of a set of image correspondences $\bar{\mathbf{x}}_i \leftrightarrow \bar{\mathbf{x}}'_i$ between two images. The number of such correspondences will vary. Corresponding points will be chosen in such a way that they correspond via a given fixed projective transformation $\overline{\mathtt{H}}$, and the correspondence is exact, in the sense that $\bar{\mathbf{x}}'_i = \overline{\mathtt{H}}\bar{\mathbf{x}}_i$ precisely, up to machine accuracy.

Next, artificial Gaussian noise will be added to the image measurements by perturbing both the x- and y-coordinates of the point by a zero-mean Gaussian random variable with known variance. The resulting noisy points are denoted $\mathbf{x}_i$ and $\mathbf{x}'_i$. A suitable Gaussian random number generator is given in [Press-88]. The estimation algorithm is then run to compute the estimated quantity. For the 2D projective transformation problem considered in chapter 4, this means the projective transformation itself, and also perhaps estimates of the correct original noise-free image points. The algorithm is then evaluated according to how closely the computed model matches the (noisy) input data, or alternatively, how closely the estimated model agrees with the original

noise-free data. This procedure should be carried out many times with different noise (i.e. a different seed for the random number generator, though each time with the same noise variance) in order to obtain a statistically meaningful performance evaluation.

5.1.1 Error in one image

To illustrate this, we continue our investigation of the problem of 2D homography estimation. For simplicity we consider the case where noise is added to the coordinates of the second image only. Thus, $\mathbf{x}_i = \bar{\mathbf{x}}_i$ for all i. Let $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$ be a set of noisy matched points between two images, generated from a perfectly matched set of data by injection of Gaussian noise with variance σ^2 in each of the two coordinates of the second (primed) image. Let there be n such matched points. From this data, a projective transformation $\hat{\mathtt{H}}$ is estimated using any one of the algorithms described in chapter 4. Obviously, the estimated transformation $\hat{\mathtt{H}}$ will not generally map $\mathbf{x}_i$ to $\mathbf{x}'_i$, nor $\bar{\mathbf{x}}_i$ to $\bar{\mathbf{x}}'_i$ precisely, because of the injected noise in the coordinates of $\mathbf{x}'_i$. The RMS (root-mean-squared) residual error

$$\epsilon_{\text{res}} = \left(\frac{1}{2n} \sum_{i=1}^{n} d(\mathbf{x}'_i, \hat{\mathbf{x}}'_i)^2 \right)^{1/2} \tag{5.1}$$

measures the average difference between the noisy input data ($\mathbf{x}'_i$) and the estimated points $\hat{\mathbf{x}}'_i = \hat{\mathtt{H}}\bar{\mathbf{x}}_i$. It is therefore appropriately called *residual error*. It measures how well the computed transformation matches the input data, and as such is a suitable quality measure for the estimation procedure.

The value of the residual error is *not* in itself an absolute measure of the quality of the solution obtained. For instance, consider the 2D projectivity problem in the case where the input data consists of just 4 matched points. Since a projective transformation is defined uniquely and exactly by 4 point correspondences, any reasonable algorithm will compute an $\hat{\mathtt{H}}$ that matches the points exactly, in the sense that $\mathbf{x}'_i = \hat{\mathtt{H}}\mathbf{x}_i$. This means that the residual error is zero. One cannot expect any better performance from an algorithm than this.

Note that $\hat{\mathtt{H}}$ matches the projected points to the input data $\mathbf{x}'_i$, and not to the original noise-free data, $\bar{\mathbf{x}}'_i$. In fact, since the difference between the noise-free and the noisy coordinates has variance σ^2, in the minimal four-point case the residual difference between projected points $\hat{\mathtt{H}}\mathbf{x}_i$ and the noise-free data $\bar{\mathbf{x}}'_i$ also has variance σ^2. Thus, in the case of 4 points, the model fits the noisy input data perfectly (i.e. the residual is zero), but does not give a very close approximation to the true noise-free values.

With more than 4 point matches, the value of the residual error will increase. In fact, intuitively, one expects that as the number of measurements (matched points) increases, the estimated model should agree more and more closely with the noise-free true values. Asymptotically, the variance should decrease in inverse proportion to the number of point matches. At the same time, the residual error will increase.

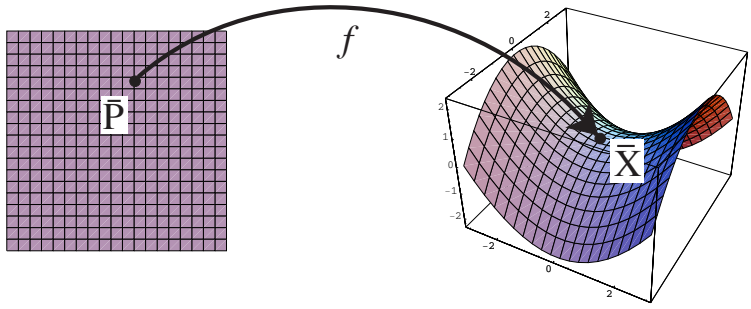

Fig. 5.1. *As the values of the parameters* $\mathbf{P}$ *vary, the function image traces out a surface* S_{M} *through the true value* $\overline{\mathbf{X}}$.

5.1.2 Error in both images

In the case of error in both images, the residual error is

$$\epsilon_{\mathrm{res}} = \frac{1}{\sqrt{4n}} \left(\sum_{i=1}^{n} d(\mathbf{x}_i, \hat{\mathbf{x}}_i)^2 + \sum_{i=1}^{n} d(\mathbf{x}'_i, \hat{\mathbf{x}}'_i)^2 \right)^{1/2} \tag{5.2}$$

where $\hat{\mathbf{x}}_i$ and $\hat{\mathbf{x}}'_i$ are estimated points such that $\hat{\mathbf{x}}'_i = \hat{\mathtt{H}}\hat{\mathbf{x}}_i$.

5.1.3 Optimal estimators (MLE)

Bounds for estimation performance will be considered in a general framework, and then specialized to the two cases of error in one or both images. The goal is to derive formulae for the expected residual error of the Maximum Likelihood Estimate (MLE). As described previously, minimization of geometric error is equivalent to MLE, and so the goal of any algorithm implementing minimization of geometric error should be to achieve the theoretical bound given for the MLE. Another algorithm minimizing a different cost function (such as algebraic error) can be judged according to how close it gets to the bound given by the MLE.

A general estimation problem is concerned with a function f from $\mathbb{R}^M$ to $\mathbb{R}^N$ as described in section 4.2.7(*p*101), where $\mathbb{R}^M$ is a parameter space, and $\mathbb{R}^N$ is a space of measurements. Consider now a point $\overline{\mathbf{X}} \in \mathbb{R}^N$ for which there exists a vector of parameters $\overline{\mathbf{P}} \in \mathbb{R}^M$ such that $f(\overline{\mathbf{P}}) = \overline{\mathbf{X}}$ (i.e. a point $\overline{\mathbf{X}}$ in the range of f with preimage $\overline{\mathbf{P}}$). In the context of 2D projectivities with measurements in the second image only, this corresponds to a noise-free set of points $\bar{\mathbf{x}}'_i = \overline{\mathtt{H}}\bar{\mathbf{x}}_i$. The x- and y-components of the n points $\bar{\mathbf{x}}'_i, i = 1, \ldots, n$ constitute the N-vector $\overline{\mathbf{X}}$ with $N = 2n$, and the parameters of the homography constitute the vector $\overline{\mathbf{P}}$ which may be an 8- or 9-vector depending on the parametrization of $\overline{\mathtt{H}}$.

Let $\mathbf{X}$ be a measurement vector chosen according to an isotropic Gaussian distribution with mean the true measurement $\overline{\mathbf{X}}$ and variance $N\sigma^2$ (this notation means that each of the N components has variance σ^2). As the value of the parameter vector $\mathbf{P}$ varies in a neighbourhood of the point $\overline{\mathbf{P}}$, the value of the function $f(\mathbf{P})$ traces out a surface S_{M} in $\mathbb{R}^N$ through the point $\overline{\mathbf{X}}$. This is illustrated in figure 5.1. The surface S_{M}

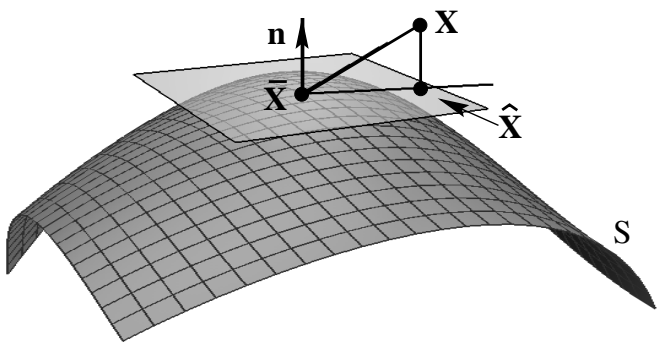

Fig. 5.2. *Geometry of the errors in measurement space using the tangent plane approximation to* S_{M}. *The estimated point* $\widehat{\mathbf{X}}$ *is the closest point on* S_{M} *to the measured point* $\mathbf{X}$. *The residual error is the distance between the measured point* $\mathbf{X}$ *and* $\widehat{\mathbf{X}}$. *The estimation error is the distance from* $\widehat{\mathbf{X}}$ *to the true point* $\overline{\mathbf{X}}$.

is given by the range of f. The dimension of the surface as a submanifold of $\mathbb{R}^N$ is equal to d, where d is the number of essential parameters (that is the number of degrees of freedom, or minimum number of parameters). In the single-image error case, this equals 8, since the mapping determined by the matrix H is independent of scale.

Now, given a measurement vector $\mathbf{X}$, the maximum likelihood (ML) estimate $\widehat{\mathbf{X}}$ is the point on S_{M} closest to $\mathbf{X}$. The ML estimator is the one that returns this closest point to $\mathbf{X}$ that lies on this surface. Denote this ML estimate by $\widehat{\mathbf{X}}$.

We now assume that in the neighbourhood of $\overline{\mathbf{X}}$, the surface is essentially planar and is well approximated by the tangent surface – at least for neighbourhoods around $\overline{\mathbf{X}}$ of the order of magnitude of noise variance. In this linear approximation, the ML estimate $\widehat{\mathbf{X}}$ is the foot of the perpendicular from $\mathbf{X}$ onto the tangent plane. The residual error is the distance from the point $\mathbf{X}$ to the estimated value $\widehat{\mathbf{X}}$. Furthermore, the distance from $\widehat{\mathbf{X}}$ to (the unknown) $\overline{\mathbf{X}}$ is the distance from the optimally estimated value to the true value as seen in figure 5.2. Our task is to compute the expected value of these errors.

Computing the expected ML residual error has now been abstracted to a geometric problem as follows. The *total variance* of an N-dimensional Gaussian distribution is the trace of the covariance matrix, that is the sum of variances in each of the axial directions. This is, of course, unchanged by a change of orthogonal coordinate frame. For an N-dimensional isotropic Gaussian distribution with independent variances σ^2 in each variable, the total variance is $N\sigma^2$. Now, given an isotropic Gaussian random variable defined on $\mathbb{R}^N$ with total variance $N\sigma^2$ and mean the true point $\overline{\mathbf{X}}$, we wish to compute the expected distance of the random variable from a dimension d hyperplane passing through $\overline{\mathbf{X}}$. The projection of a Gaussian random variable in $\mathbb{R}^N$ onto the d-dimensional tangent plane gives the distribution of the *estimation error* (the difference between the estimated value and the true result). Projection onto the

$(N-d)$-dimensional normal to the tangent surface gives the distribution of the residual error.

By a rotation of axes if necessary, one may assume, without loss of generality, that the tangent surface coincides with the first d coordinate axes. Integration over the remaining axial directions provides the following result.

Result 5.1. *The projection of an isotropic Gaussian distribution defined on $\mathbb{R}^N$ with total variance $N\sigma^2$ onto a subspace of dimension s is an isotropic Gaussian distribution with total variance $s\sigma^2$.*

The proof of this is straightforward, and is omitted. We apply this in the two cases where $s = d$ and $s = N - d$ to obtain the following results.

Result 5.2. *Consider an estimation problem where N measurements are to be modelled by a function depending on a set of d essential parameters. Suppose the measurements are subject to independent Gaussian noise with standard deviation σ in each measurement variable.*

(i) *The RMS* **residual error** *(distance of the measured from the estimated value) for the ML estimator is*

$$\epsilon_{\text{res}} = E[\|\hat{\mathbf{X}} - \mathbf{X}\|^2/N]^{1/2} = \sigma(1 - d/N)^{1/2} \tag{5.3}$$

(ii) *The RMS* **estimation error** *(distance of the estimated from the true value) for the ML estimator is*

$$\epsilon_{\text{est}} = E[\|\hat{\mathbf{X}} - \overline{\mathbf{X}}\|^2/N]^{1/2} = \sigma(d/N)^{1/2} \tag{5.4}$$

where $\mathbf{X}$, $\hat{\mathbf{X}}$ and $\overline{\mathbf{X}}$ are respectively the measured, estimated and true values of the measurement vector.

Result 5.2 follows directly from result 5.1 by dividing by N to get the variance per measurement, then taking a square root to get standard deviation, instead of variance.

These values give lower bounds for residual error against which a particular estimation algorithm may be measured.

2D homography – error in one image. For the 2D projectivity estimation problem considered in this chapter, assuming error in the second image only, we have $d = 8$ and $N = 2n$, where n is the number of point matches. Thus, we have for this problem

$$\begin{aligned} \epsilon_{\text{res}} &= \sigma\,(1 - 4/n)^{1/2} \\ \epsilon_{\text{est}} &= \sigma\,(4/n)^{1/2}\,. \end{aligned} \tag{5.5}$$

Graphs of these errors as n varies are shown in figure 5.3.

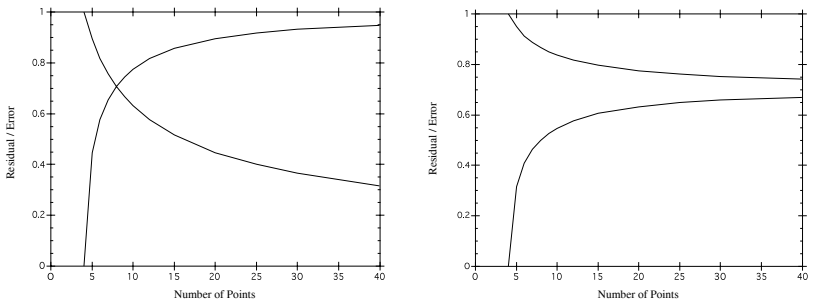

Fig. 5.3. *Optimal error when noise is present in (a) one image, and in (b) both images as the number of points varies. An error level of one pixel is assumed. The descending curve shows the estimation error* $\epsilon_{\rm est}$ *and the ascending curve shows the residual error* $\epsilon_{\rm res}$.

Error in both images. In this case, $N = 4n$ and $d = 2n + 8$. As before, assuming linearity of the tangent surface in the neighbourhood of the true measurement vector $\widehat{\mathbf{X}}$, result 5.2 gives the following expected errors.

$$\begin{aligned} \epsilon_{\rm res} &= \sigma\left(\frac{n-4}{2n}\right)^{1/2} \\ \epsilon_{\rm est} &= \sigma\left(\frac{n+4}{2n}\right)^{1/2}. \end{aligned} \tag{5.6}$$

Graphs of these errors as n varies are also shown in Figure 5.3.

An interesting observation to be made from this graph is that the asymptotic error with respect to the true values is $\sigma/\sqrt{2}$, and not 0 as in the case of error in one image. This result is expected, since in effect, one has two measurements of the position of each point, one in each image, related by the projective transformation. With two measurements of a point the variance in the estimate of the point position decreases by a factor of $\sqrt{2}$. By contrast, in the previous case where errors occur in one image only, one has one exact measurement for each point (i.e. in the first image). Thus, as the transformation H is estimated with greater and greater accuracy, the exact position of the point in the second image becomes known with uncertainty asymptotically approaching 0.

Mahalanobis distance. The formulae quoted above were derived under the assumption that the error distribution in measurement space was an isotropic Gaussian distribution, meaning that errors in each coordinate were independent. This assumption is not essential. We may assume any Gaussian distribution of error, with covariance matrix Σ. The formulae of result 5.2 remain true with ϵ being replaced with the expected Mahalanobis distance $E[\|\widehat{\mathbf{X}} - \mathbf{X}\|^2_\Sigma/N]^{1/2}$. The standard deviation σ also disappears, since it is taken care of by the Mahalanobis distance.

This follows from a simple change of coordinates in the measurement space $\mathbb{R}^N$ to make the covariance matrix equal to the identity. In this new coordinate frame, Mahalanobis distance becomes the same as Euclidean distance.

5.1.4 Determining the correct convergence of an algorithm

The relations given in (5.3) and (5.4) give a simple way of determining correct convergence of an estimation algorithm, without the need to determine the number of degrees of freedom of the problem. As seen in figure 5.2, the measurement space corresponding to the model specified by the parameter vector $\mathbf{P}$ forms a surface S_{M}. If near the noise-free data $\overline{\mathbf{X}}$ the surface is nearly planar, then it may be approximated by its tangent plane, and the three points $\widehat{\mathbf{X}}$, $\mathbf{X}$ and $\overline{\mathbf{X}}$ form a right-angled triangle. In most estimation problems this assumption of planarity will be very close to correct at the scale set by typical noise magnitude. In this case, the Pythagorean equality may be written as

$$\|\mathbf{X} - \overline{\mathbf{X}}\|^2 = \|\mathbf{X} - \widehat{\mathbf{X}}\|^2 + \|\overline{\mathbf{X}} - \widehat{\mathbf{X}}\|^2 \tag{5.7}$$

In evaluating an algorithm with synthetic data, this equality allows a simple test to see whether the algorithm has converged to the optimal value. If the estimated value $\widehat{\mathbf{X}}$ satisfies this equality, then it is a strong indication that the algorithm has found the true global minimum. Note that it is unnecessary in applying this test to determine the number of degrees of freedom of the problem. A few more properties are listed:

- This test can be used to determine on a run-by-run basis whether the algorithm has succeeded. Thus, with repeated runs, it allows an estimate of the percentage success rate for the algorithm.
- This test can only be used for synthetic data, or at least data for which the true measurements $\overline{\mathbf{X}}$ are known.
- The equality (5.7) depends on the assumption that the surface S_{M} consisting of valid measurements is locally planar. If the equality is not satisfied for a particular run of the estimation algorithm, then this is because the surface is not planar, or (far more likely) because the algorithm is failing to find the best solution.
- The test (5.7) is a test for the algorithm finding the global, not a local solution. If $\widehat{\mathbf{X}}$ settles to a local cost minimum, then the right-hand-side of (5.7) is likely to be much larger than the left-hand-side. The condition is unlikely to be satisfied entirely by chance if the algorithm finds the incorrect point $\widehat{\mathbf{X}}$.

5.2 Covariance of the estimated transformation

In the previous section the ML estimate was considered, and how its expected average error may be computed. Comparing the achieved residual error or estimation error of an algorithm against the ML error is a good way of evaluating the performance of a particular estimation algorithm, since it compares the results of the algorithm against the best that may be achieved (the optimum estimate) in the absence of any other prior information.

Nevertheless, the chief concern is how accurately the transformation itself has been computed. The uncertainty of the estimated transformation depends on many factors, including the number of points used to compute it, the accuracy of the given point matches, as well as the configuration of the points in question. To illustrate the importance of the configuration suppose the points used to compute the transformation are

close to a degenerate configuration; then the transformation may not be computed with great accuracy. For instance, if the transformation is computed from a set of points that lie close to a straight line, then the behaviour of the transformation in the dimension perpendicular to that line is not accurately determined. Thus, whereas the achievable residual error and estimation error were seen to be dependent only on the number of point correspondences and their accuracy, by contrast, the accuracy of the computed transformation is dependent on the particular points. The uncertainty of the computed transformation is conveniently captured in the *covariance matrix* of the transformation. Since $\mathtt{H}$ is a matrix with 9 entries, its covariance matrix will be a 9×9 matrix. In this section it will be seen how this covariance matrix may be computed.

5.2.1 Forward propagation of covariance

The covariance matrix behaves in a pleasantly simple manner under affine transformations, as described in the following theorem.

Result 5.3. *Let* $\mathbf{v}$ *be a random vector in* $\mathbb{R}^M$ *with mean* $\bar{\mathbf{v}}$ *and covariance matrix* Σ*, and suppose that* $f : \mathbb{R}^M \to \mathbb{R}^N$ *is an affine mapping defined by* $f(\mathbf{v}) = f(\bar{\mathbf{v}}) + \mathtt{A}(\mathbf{v} - \bar{\mathbf{v}})$*. Then* $f(\mathbf{v})$ *is a random variable with mean* $f(\bar{\mathbf{v}})$ *and covariance matrix* $\mathtt{A}\Sigma\mathtt{A}^\mathsf{T}$*.*

Note that it is not assumed that $\mathtt{A}$ is a square matrix. Instead of giving a proof of this theorem, we give an example.

Example 5.4. Let x and y be independent random variables with mean 0 and standard deviations of 1 and 2 respectively. What are the mean and standard deviation of $x' = f(x, y) = 3x + 2y - 7$?

The mean is $\bar{x}' = f(0, 0) = -7$. Next we compute the variance of x'. In this case, Σ is the matrix $\begin{bmatrix} 1 & 0 \\ 0 & 4 \end{bmatrix}$ and $\mathtt{A}$ is the matrix $[3\ 2]$. Thus, the variance of x' is $\mathtt{A}\Sigma\mathtt{A}^\mathsf{T} = 25$. Thus $3x + 2y - 7$ has standard deviation 5. △

Example 5.5. Let $x' = 3x+2y$ and $y' = 3x-2y$. Find the covariance matrix of (x', y'), given that x and y have the same distribution as before.

In this case, the matrix $\mathtt{A} = \begin{bmatrix} 3 & 2 \\ 3 & -2 \end{bmatrix}$. One computes $\mathtt{A}\Sigma\mathtt{A}^\mathsf{T} = \begin{bmatrix} 25 & -7 \\ -7 & 25 \end{bmatrix}$. Thus, one sees that both x' and y' have variance 25 (standard deviation 5), whereas x' and y' are negatively correlated, with covariance $E[x'y'] = -7$. △

Non-linear propagation. If $\mathbf{v}$ is a random vector in $\mathbb{R}^M$ and $f : \mathbb{R}^M \to \mathbb{R}^N$ is a non-linear function acting on $\mathbf{v}$, then we may compute an approximation to the mean and covariance of $f(\mathbf{v})$ by assuming that f is approximately affine in the vicinity of the mean of the distribution. The affine approximation to f is $f(\mathbf{v}) \approx f(\bar{\mathbf{v}}) + \mathtt{J}(\mathbf{v} - \bar{\mathbf{v}})$, where $\mathtt{J}$ is the partial derivative (Jacobian) matrix $\partial f / \partial \mathbf{v}$ evaluated at $\bar{\mathbf{v}}$. Note that $\mathtt{J}$ has dimension $N \times M$. Then we have the following result.

Result 5.6. *Let* $\mathbf{v}$ *be a random vector in* $\mathbb{R}^M$ *with mean* $\bar{\mathbf{v}}$ *and covariance matrix* Σ*,*

and let $f : \mathbb{R}^M \to \mathbb{R}^N$ be differentiable in a neighbourhood of $\bar{\mathbf{v}}$. Then up to a first-order approximation, $f(\mathbf{v})$ is a random variable with mean $f(\bar{\mathbf{v}})$ and covariance $\mathtt{J}\Sigma\mathtt{J}^\mathsf{T}$, where $\mathtt{J}$ is the Jacobian matrix of f, evaluated at $\bar{\mathbf{v}}$.

The extent to which this result gives a good approximation to the actual mean and variance of $f(\bar{\mathbf{v}})$ depends on how closely the function f is approximated by a linear function in a region about $\bar{\mathbf{v}}$ commensurate in size with the support of the probability distribution of $\mathbf{v}$.

Example 5.7. Let $\mathbf{x} = (x, y)^\mathsf{T}$ be a Gaussian random vector with mean $(0, 0)^\mathsf{T}$ and covariance matrix $\sigma^2\mathrm{diag}(1, 4)$. Let $x' = f(x, y) = x^2 + 3x - 2y + 5$. Then one may compute the true values of the mean and standard deviation of $f(x, y)$ according to the formulae

$$\begin{aligned} \bar{x}' &= \int\int_{-\infty}^{\infty} P(x,y) f(x,y) dx dy \\ \sigma_{x'}^2 &= \int\int_{-\infty}^{\infty} P(x,y)(f(x,y) - \bar{x}')^2 dx dy \end{aligned}$$

where

$$P(x, y) = \frac{1}{4\pi\sigma^2} e^{-(x^2+y^2/4)/2\sigma^2}$$

is the Gaussian probability distribution (A2.1–*p*565). One obtains

$$\begin{aligned} \bar{x}' &= 5 + \sigma^2 \\ \sigma_{x'}^2 &= 25\sigma^2 + 2\sigma^4. \end{aligned}$$

Applying the approximation given by result 5.6, and noting that $\mathtt{J} = [3 \ -2]$, we find that the estimated values are

$$\begin{aligned} \bar{x}' &= 5 \\ \sigma_{x'}^2 &= \sigma^2 [3 \ -2] \begin{bmatrix} 1 & \\ & 4 \end{bmatrix} [3 \ -2]^\mathsf{T} = 25\sigma^2. \end{aligned}$$

Thus, as long as σ is small, this is a close approximation to the correct values of the mean and variance of x'. The following table shows the true and approximated values for the mean and standard deviation of $f(x, y)$ for two different values of σ.

	$\sigma = 0.25$		$\sigma = 0.5$	
	$\bar{x}'$	$\sigma_{x'}$	$\bar{x}'$	$\sigma_{x'}$
estimate	5.0000	1.25000	5.00	2.5000
true	5.0625	1.25312	5.25	2.5249

For reference, in the case $\sigma = 0.25$, one sees that as long as $|x| < 2\sigma$ (about 95% of the total distribution) the value $f(x, y) = x^2 + 3x - 2y + 5$ differs from its linear approximation $3x - 2y + 5$ by no more than $x^2 < 0.25$. △

Example 5.8. More generally, assuming that x and y are independent zero-mean Gaussian random variables, one may compute that for the function $f(x,y) = ax^2 + bxy + cy^2 + dx + ey + f$,

$$\begin{aligned} \text{mean} &= a\sigma_x^2 + c\sigma_y^2 + f \\ \text{variance} &= 2a^2\sigma_x^4 + b^2\sigma_x^2\sigma_y^2 + 2c^2\sigma_y^4 + d^2\sigma_x^2 + e^2\sigma_y^2 \end{aligned}$$

which are close to the estimated values mean $= f$, variance $= d^2\sigma_x^2 + e^2\sigma_y^2$ as long as σ_x and σ_y are small. △

5.2.2 Backward propagation of covariance

The material in this and the following section 5.2.3 is more advanced. The examples in section 5.2.4 show the straightforward application of the results of these sections, and can be read first.

Consider a differentiable mapping f from a "parameter space", $\mathbb{R}^M$ to a "measurement space" $\mathbb{R}^N$, and let a Gaussian probability distribution be defined on $\mathbb{R}^N$ with covariance matrix Σ. Let S_{M} be the image of the mapping f. We assume that $M < N$ and that S_{M} has the same dimension M as the parameter space $\mathbb{R}^M$. Thus we are not considering the over-parametrized case at present. A vector $\mathbf{P}$ in $\mathbb{R}^M$ represents a parametrization of the point $f(\mathbf{P})$ on S_{M}. Finding the point on S_{M} closest in Mahalanobis distance to a given point $\mathbf{X}$ in $\mathbb{R}^N$ defines a map from $\mathbb{R}^N$ to the surface S_{M}. We call this mapping $\eta : \mathbb{R}^N \to S_{\text{M}}$. Now, f is by assumption invertible on the surface S_{M}, and we define $f^{-1} : S_{\text{M}} \to \mathbb{R}^M$ to be the inverse function.

By composing the map $\eta : \mathbb{R}^N \to S_{\text{M}}$ and $f^{-1} : S_{\text{M}} \to \mathbb{R}^M$ we have a mapping $f^{-1} \circ \eta : \mathbb{R}^N \to \mathbb{R}^M$. This mapping assigns to a measurement vector $\mathbf{X}$, the set of parameters $\mathbf{P}$ corresponding to the ML estimate $\widehat{\mathbf{X}}$. In principle we may propagate the covariance of the probability distribution in the measurement space $\mathbb{R}^N$ to compute a covariance matrix for the set of parameters $\mathbf{P}$ corresponding to ML estimation. Our goal is to apply result 5.3 or result 5.6.

We consider first the case where the mapping f is an affine mapping from $\mathbb{R}^M$ into $\mathbb{R}^N$. We will show next that the mapping $f^{-1} \circ \eta$ is also an affine mapping, and a specific form will be given for $f^{-1} \circ \eta$, thereby allowing us to apply result 5.3 to compute the covariance of the estimated parameters $\widehat{\mathbf{P}} = f^{-1} \circ \eta(\mathbf{X})$.

Since f is affine, we may write $f(\mathbf{P}) = f(\overline{\mathbf{P}}) + \mathtt{J}(\mathbf{P} - \overline{\mathbf{P}})$, where $f(\overline{\mathbf{P}}) = \overline{\mathbf{X}}$ is the mean of the probability distribution on $\mathbb{R}^N$. Since we are assuming that the surface $S_{\text{M}} = f(\mathbb{R}^M)$ has dimension M, the rank of $\mathtt{J}$ is equal to its column dimension. Given a measurement vector $\mathbf{X}$, the ML estimate $\widehat{\mathbf{X}}$ minimizes $\|\mathbf{X} - \widehat{\mathbf{X}}\|_\Sigma = \|\mathbf{X} - f(\widehat{\mathbf{P}})\|_\Sigma$. Thus, we seek $\widehat{\mathbf{P}}$ to minimize this latter quantity. However,

$$\|\mathbf{X} - f(\widehat{\mathbf{P}})\|_\Sigma = \|(\mathbf{X} - \overline{\mathbf{X}}) - \mathtt{J}(\widehat{\mathbf{P}} - \overline{\mathbf{P}})\|_\Sigma$$

and this is minimized (see (A5.2–*p*591) in section A5.2.1(*p*591)) when

$$(\widehat{\mathbf{P}} - \overline{\mathbf{P}}) = (\mathtt{J}^\mathsf{T}\Sigma^{-1}\mathtt{J})^{-1}\mathtt{J}^\mathsf{T}\Sigma^{-1}(\mathbf{X} - \overline{\mathbf{X}}) \ .$$

Writing $\overline{\mathbf{P}} = f^{-1}\overline{\mathbf{X}}$ and $\widehat{\mathbf{P}} = f^{-1}\widehat{\mathbf{X}}$, we see that

$$\begin{aligned} f^{-1} \circ \eta(\mathbf{X}) &= \widehat{\mathbf{P}} \\ &= (\mathtt{J}^\mathsf{T}\Sigma^{-1}\mathtt{J})^{-1}\mathtt{J}^\mathsf{T}\Sigma^{-1}(\mathbf{X} - \overline{\mathbf{X}}) + f^{-1}(\overline{\mathbf{X}}) \\ &= (\mathtt{J}^\mathsf{T}\Sigma^{-1}\mathtt{J})^{-1}\mathtt{J}^\mathsf{T}\Sigma^{-1}(\mathbf{X} - \overline{\mathbf{X}}) + f^{-1} \circ \eta(\overline{\mathbf{X}}) \ . \end{aligned}$$

This shows that $f^{-1} \circ \eta$ is affine and $(\mathtt{J}^\mathsf{T}\Sigma^{-1}\mathtt{J})^{-1}\mathtt{J}^\mathsf{T}\Sigma^{-1}$ is its linear part. Applying result 5.3, we see that the covariance matrix for $\widehat{\mathbf{P}}$ is

$$\begin{aligned} \left[(\mathtt{J}^\mathsf{T}\Sigma^{-1}\mathtt{J})^{-1}\mathtt{J}^\mathsf{T}\Sigma^{-1}\right] \Sigma \left[(\mathtt{J}^\mathsf{T}\Sigma^{-1}\mathtt{J})^{-1}\mathtt{J}^\mathsf{T}\Sigma^{-1}\right]^\mathsf{T} &= (\mathtt{J}^\mathsf{T}\Sigma^{-1}\mathtt{J})^{-1}\mathtt{J}^\mathsf{T}\Sigma^{-1}\Sigma\Sigma^{-1}\mathtt{J}(\mathtt{J}^\mathsf{T}\Sigma^{-1}\mathtt{J})^{-1} \\ &= (\mathtt{J}^\mathsf{T}\Sigma^{-1}\mathtt{J})^{-1}, \end{aligned}$$

recalling that Σ is symmetric. We have proved the following theorem.

Result 5.9 Backward transport of covariance – affine case. *Let $f : \mathbb{R}^M \to \mathbb{R}^N$ be an affine mapping of the form $f(\mathbf{P}) = f(\overline{\mathbf{P}}) + \mathtt{J}(\mathbf{P} - \overline{\mathbf{P}})$, where $\mathtt{J}$ has rank M. Let $\mathbf{X}$ be a random variable in $\mathbb{R}^N$ with mean $\overline{\mathbf{X}} = f(\overline{\mathbf{P}})$ and covariance matrix Σ. Let $f^{-1} \circ \eta : \mathbb{R}^N \to \mathbb{R}^M$ be the mapping that maps a measurement $\mathbf{X}$ to the set of parameters corresponding to the ML estimate $\widehat{\mathbf{X}}$. Then $\widehat{\mathbf{P}} = f^{-1} \circ \eta(\mathbf{X})$ is a random variable with mean $\overline{\mathbf{P}}$ and covariance matrix*

$$\Sigma_{\mathbf{P}} = (\mathtt{J}^\mathsf{T}\Sigma_{\mathbf{X}}^{-1}\mathtt{J})^{-1}. \tag{5.8}$$

In the case where f is not affine, an approximation to the mean and covariance may be obtained by approximating f by an affine function in the usual way, as follows.

Result 5.10 Backward transport of covariance – non-linear case. *Let $f : \mathbb{R}^M \to \mathbb{R}^N$ be a differentiable mapping and let $\mathtt{J}$ be its Jacobian matrix evaluated at a point $\overline{\mathbf{P}}$. Suppose that $\mathtt{J}$ has rank M. Then f is one-to-one in a neighbourhood of $\overline{\mathbf{P}}$. Let $\mathbf{X}$ be a random variable in $\mathbb{R}^N$ with mean $\overline{\mathbf{X}} = f(\overline{\mathbf{P}})$ and covariance matrix $\Sigma_{\mathbf{X}}$. Let $f^{-1} \circ \eta : \mathbb{R}^N \to \mathbb{R}^M$ be the mapping that maps a measurement $\mathbf{X}$ to the set of parameters corresponding to the ML estimate $\widehat{\mathbf{X}}$. Then to first-order, $\widehat{\mathbf{P}} = f^{-1} \circ \eta(\mathbf{X})$ is a random variable with mean $\overline{\mathbf{P}}$ and covariance matrix $(\mathtt{J}^\mathsf{T}\Sigma_{\mathbf{X}}^{-1}\mathtt{J})^{-1}$.*

5.2.3 Over-parametrization

One may generalize result 5.9 and result 5.10 to the case of redundant sets of parameters – the over-parametrized case. In this case, the mapping f from the parameter space $\mathbb{R}^M$ to measurement space $\mathbb{R}^N$ is not locally one-to-one. For instance, in the case of estimating a 2D homography as discussed in section 4.5(p110) there is a mapping $f(\mathbf{P})$ where $\mathbf{P}$ is a 9-vector representing the entries of the homography matrix $\mathtt{H}$. Since the homography has only 8 degrees of freedom, the mapping f is not one-to-one. In particular, for any constant k, the matrix $k\mathtt{H}$ represents the same map, and so the image coordinate vectors $f(\mathbf{P})$ and $f(k\mathbf{P})$ are equal.

In the general case of a mapping $f : \mathbb{R}^M \to \mathbb{R}^N$ the Jacobian matrix $\mathtt{J}$ does not have full rank M, but rather a smaller rank $d < M$. This rank d is called the number of *essential parameters*. The matrix $\mathtt{J}^\mathsf{T}\Sigma_{\mathbf{X}}^{-1}\mathtt{J}$ in this case has dimension M but rank

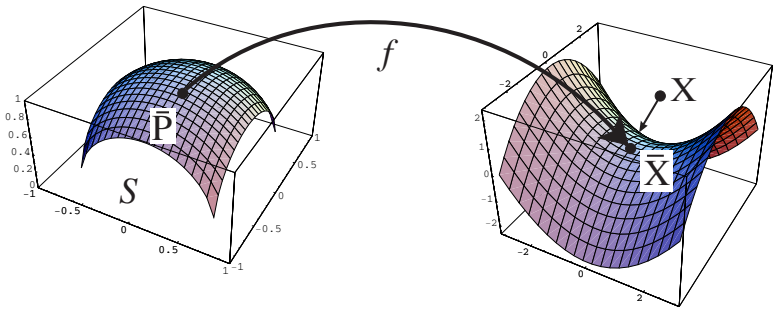

Fig. 5.4. **Back propagation (over-parametrized).** *Mapping f maps constrained parameter surface to measurement space. A measurement $\mathbf{X}$ is mapped (by a mapping η) to the closest point on the surface $f(S_{\mathrm{P}})$ and then back via f^{-1} to the parameter space, providing the ML estimate of the parameters. The covariance of $\mathbf{X}$ is transferred via $f^{-1} \circ \eta$ to a covariance of the parameters.*

$d < M$. The formula (5.8), $\Sigma_{\mathbf{P}} = (\mathtt{J}^{\mathsf{T}}\Sigma_{\mathbf{X}}^{-1}\mathtt{J})^{-1}$, clearly does not hold, since the matrix on the right side is not invertible.

In fact, it is clear that without any further restriction, the elements of the estimated vector $\widehat{\mathbf{P}}$ may vary without bound, namely through multiplication by an arbitrary constant k. Hence the elements have infinite variance. It is usual to restrict the estimated homography matrix $\mathtt{H}$ or more generally the parameter vector $\mathbf{P}$ by imposing some constraint. The usual constraint is that $\|\mathbf{P}\| = 1$ though other constraints are possible, such as demanding that the last parameter should equal 1 (see section 4.4.2(*p*105)). Thus, the parameter vector $\mathbf{P}$ is constrained to lie on a surface in the parameter space $\mathbb{R}^9$, or generally $\mathbb{R}^M$. In the first case the surface $\|\mathbf{P}\| = 1$ is the unit sphere in $\mathbb{R}^M$. The constraint $P_m = 1$ represents a plane in $\mathbb{R}^M$. In the general case we may assume that the estimated vector $\mathbf{P}$ lies on some submanifold of $\mathbb{R}^M$ as in the following theorem.

Result 5.11. Backward transport of covariance – over-parametrized case. *Let $f : \mathbb{R}^M \to \mathbb{R}^N$ be a differentiable mapping taking a parameter vector $\overline{\mathbf{P}}$ to a measurement vector $\overline{\mathbf{X}}$. Let S_{P} be a smooth manifold of dimension d embedded in $\mathbb{R}^M$ passing through point $\overline{\mathbf{P}}$, and such that the map f is one-to-one on the manifold S_{P} in a neighbourhood of $\overline{\mathbf{P}}$, mapping S_{P} locally to a manifold $f(S_{\mathrm{P}})$ in $\mathbb{R}^N$. The function f has a local inverse, denoted f^{-1}, restricted to the surface $f(S_{\mathrm{P}})$ in a neighbourhood of $\overline{\mathbf{X}}$. Let a Gaussian distribution on $\mathbb{R}^N$ be defined with mean $\overline{\mathbf{X}}$ and covariance matrix $\Sigma_{\mathbf{X}}$ and let $\eta : \mathbb{R}^N \to f(S_{\mathrm{P}})$ be the mapping that takes a point in $\mathbb{R}^N$ to the closest point on $f(S_{\mathrm{P}})$ with respect to Mahalanobis norm $\|\cdot\|_{\Sigma_{\mathbf{X}}}$. Via $f^{-1} \circ \eta$ the probability distribution on $\mathbb{R}^N$ with covariance matrix $\Sigma_{\mathbf{X}}$ induces a probability distribution on $\mathbb{R}^M$ with covariance matrix, to first-order equal to*

$$\Sigma_{\mathbf{P}} = (\mathtt{J}^{\mathsf{T}}\Sigma_{\mathbf{X}}^{-1}\mathtt{J})^{+\mathtt{A}} = \mathtt{A}(\mathtt{A}^{\mathsf{T}}\mathtt{J}^{\mathsf{T}}\Sigma_{\mathbf{X}}^{-1}\mathtt{J}\mathtt{A})^{-1}\mathtt{A}^{\mathsf{T}} \tag{5.9}$$

where $\mathtt{A}$ is any $m \times d$ matrix whose column vectors span the tangent space to S_{P} at $\overline{\mathbf{P}}$.

This is illustrated in figure 5.4. The notation $(\mathtt{J}^{\mathsf{T}}\Sigma_{\mathbf{X}}^{-1}\mathtt{J})^{+\mathtt{A}}$, defined by (5.9), is discussed further in section A5.2(*p*590).

Proof. The proof of result 5.11 is straightforward. Let d be the number of essential parameters. One defines a map $g : \mathbb{R}^d \to \mathbb{R}^M$ mapping an open neighbourhood U in $\mathbb{R}^d$ to an open set of S_P containing the point $\overline{\mathbf{P}}$. Then the combined mapping $f \circ g : \mathbb{R}^d \to \mathbb{R}^N$ is one-to-one on the neighbourhood U. Let us denote the partial derivative matrices of f by $\mathtt{J}$ and of g by $\mathtt{A}$. The matrix of partial derivatives of $f \circ g$ is then $\mathtt{JA}$. Result 5.10 now applies, and one sees that the probability distribution function with covariance matrix Σ on $\mathbb{R}^N$ may be transported backwards to a covariance matrix $(\mathtt{A}^\mathsf{T}\mathtt{J}^\mathsf{T}\Sigma^{-1}\mathtt{JA})^{-1}$ on $\mathbb{R}^d$. Transporting this forwards again to $\mathbb{R}^M$, applying result 5.6, we arrive at the covariance matrix $\mathtt{A}(\mathtt{A}^\mathsf{T}\mathtt{J}^\mathsf{T}\Sigma^{-1}\mathtt{JA})^{-1}\mathtt{A}^\mathsf{T}$ on S_P. This matrix, which will be denoted here by $(\mathtt{J}^\mathsf{T}\Sigma^{-1}\mathtt{J})^{+\mathtt{A}}$, is related to the pseudo-inverse of $(\mathtt{J}^\mathsf{T}\Sigma^{-1}\mathtt{J})$ as defined in section A5.2(*p*590). The expression (5.9) is not dependent on the particular choice of the matrix $\mathtt{A}$ as long as the column span of $\mathtt{A}$ is unchanged. In particular, if $\mathtt{A}$ is replaced by $\mathtt{AB}$ for any invertible $d \times d$ matrix $\mathtt{B}$, then the value of (5.9) does not change. Thus, any matrix $\mathtt{A}$ whose columns span the tangent space of S_P at $\overline{\mathbf{P}}$ will do. □

Note that the proof gives a specific way of computing a matrix $\mathtt{A}$ spanning the tangent space – namely the Jacobian matrix of g. In many instances, as we will see, there are easier ways of finding $\mathtt{A}$. Note that the covariance matrix (5.9) is singular. In particular, it has dimension M and rank $d < M$. This is because the variance of the estimated parameter set in directions orthogonal to the constraint surface S_P is zero – there can be no variation in that direction. Note that whereas $\mathtt{J}^\mathsf{T}\Sigma^{-1}\mathtt{J}$ is non-invertible, the $d \times d$ matrix $\mathtt{A}^\mathsf{T}\mathtt{J}^\mathsf{T}\Sigma^{-1}\mathtt{JA}$ has rank d and is invertible.

An important case occurs when the constraint surface is locally orthogonal to the null-space of the Jacobian matrix. Denote by $N_L(\mathtt{X})$ the left null-space of matrix $\mathtt{X}$, namely the space of all vectors $\mathbf{x}$ such that $\mathbf{x}^\mathsf{T}\mathtt{X} = \mathbf{0}$. Then (as shown in section A5.2-(*p*590)), the *pseudo-inverse* $\mathtt{X}^+$ is given by

$$\mathtt{X}^+ = \mathtt{X}^{+\mathtt{A}} = \mathtt{A}(\mathtt{A}^\mathsf{T}\mathtt{X}\mathtt{A})^{-1}\mathtt{A}^\mathsf{T}$$

if and only if $N_L(\mathtt{A}) = N_L(\mathtt{X})$. The following result then derives directly from result 5.11.

Result 5.12. *Let $f : \mathbb{R}^M \to \mathbb{R}^N$ be a differentiable mapping taking $\overline{\mathbf{P}}$ to $\overline{\mathbf{X}}$, and let $\mathtt{J}$ be the Jacobian matrix of f. Let a Gaussian distribution on $\mathbb{R}^N$ be defined at $\overline{\mathbf{X}}$ with covariance matrix $\Sigma_\mathbf{X}$ and let $f^{-1} \circ \eta : \mathbb{R}^N \to \mathbb{R}^M$ as in result 5.11 be the mapping taking a measurement $\mathbf{X}$ to the MLE parameter vector $\mathtt{P}$ constrained to lie on a surface S_P locally orthogonal to the null-space of $\mathtt{J}$. Then $f^{-1} \circ \eta$ induces a distribution on $\mathbb{R}^M$ with covariance matrix, to first-order equal to*

$$\Sigma_\mathbf{P} = (\mathtt{J}^\mathsf{T}\Sigma_\mathbf{X}^{-1}\mathtt{J})^+. \tag{5.10}$$

Note that the restriction that $\mathbf{P}$ be constrained to lie on a surface locally orthogonal to the null-space of $\mathtt{J}$ is in many cases the natural constraint. For instance, if $\mathbf{P}$ is a *homogeneous* parameter vector (such as the entries of a homogeneous matrix), the restriction is satisfied for the usual constraint $\|\mathbf{P}\| = 1$. In such a case, the constraint surface is the unit sphere, and the tangent plane at any point is perpendicular to the parameter vector. On the other hand, since $\mathtt{P}$ is a homogeneous vector, the function

$f(\mathtt{P})$ is invariant to changes of scale, and so $\mathtt{J}$ has a null-vector in the radial direction, thus perpendicular to the constraint surface.

In other cases, it is often not critical what restriction we place on the parameter set for the purpose of computing the covariance matrix of the parameters. In addition, since the pseudo-inversion operation is its own inverse, we can retrieve the original matrix from its pseudo-inverse, according to $\mathtt{J}^\mathsf{T}\Sigma_\mathbf{X}^{-1}\mathtt{J} = \Sigma_\mathbf{P}^+$. One can then compute the covariance matrix corresponding to any other subspace, according to

$$(\mathtt{J}^\mathsf{T}\Sigma_\mathbf{X}^{-1}\mathtt{J})^{+\mathtt{A}} = (\Sigma_\mathbf{P}^+)^{+\mathtt{A}}$$

where the columns of $\mathtt{A}$ span the constrained subspace of parameter space.

5.2.4 Application and examples

Error in one image. Let us consider the application of this theory to the problem of finding the covariance of an estimated 2D homography $\mathtt{H}$. First, we look at the case where the error is limited to the second image. The 3×3 matrix $\mathtt{H}$ is represented by a 9-dimensional parameter vector which will be denoted by $\mathbf{h}$ instead of $\mathbf{P}$ so as to remind us that it is made up of the entries of $\mathtt{H}$. The covariance of the estimated $\hat{\mathbf{h}}$ is a 9×9 symmetric matrix. We are given a set of matched points $\bar{\mathbf{x}}_i \leftrightarrow \mathbf{x}'_i$. The points $\bar{\mathbf{x}}_i$ are fixed true values, and the points $\mathbf{x}'_i$ are considered as random variables subject to Gaussian noise with variance σ^2 in each component, or if desired, with a more general covariance. The function $f : \mathbb{R}^9 \to \mathbb{R}^{2n}$ is defined as mapping a 9-vector $\mathbf{h}$ representing a matrix $\mathtt{H}$ to the $2n$-vector made up of the coordinates of the points $\mathbf{x}'_i = \mathtt{H}\bar{\mathbf{x}}_i$. The coordinates of $\mathbf{x}'_i$ make up a composite vector in $\mathbb{R}^N$, which we denote by $\mathbf{X}'$. As we have seen, as $\mathbf{h}$ varies, the point $f(\mathbf{h})$ traces out an 8-dimensional surface S_P in $\mathbb{R}^{2n}$. Each point $\mathbf{X}'$ on the surface represents a set of points $\mathbf{x}'_i$ consistent with the first-image points $\bar{\mathbf{x}}_i$. Given a vector of measurements $\mathbf{X}'$, one selects the closest point $\widehat{\mathbf{X}}'$ on the surface S_P with respect to Mahalanobis distance. The pre-image $\hat{\mathbf{h}} = f^{-1}(\widehat{\mathbf{X}}')$, subject to constraint $\|\mathbf{h}\| = 1$, represents the estimated homography matrix $\hat{\mathtt{H}}$, estimated using the ML estimator. From the probability distribution of values of $\mathbf{X}'$ one wishes to derive the distribution of the estimated $\hat{\mathbf{h}}$. The covariance matrix $\Sigma_\mathbf{h}$ is given by result 5.12. This covariance matrix corresponds to the constraint $\|\mathbf{h}\| = 1$.

Thus, a procedure for computing the covariance matrix of the estimated transformation is as follows.

(i) Estimate the transformation $\hat{\mathtt{H}}$ from the given data.
(ii) Compute the Jacobian matrix $\mathtt{J}_f = \partial\mathbf{X}'/\partial\mathbf{h}$, evaluated at $\hat{\mathbf{h}}$.
(iii) The covariance matrix of the estimated $\mathbf{h}$ is given by (5.10): $\Sigma_\mathbf{h} = (\mathtt{J}_f^\mathsf{T}\Sigma_{\mathbf{X}'}^{-1}\mathtt{J}_f)^+$.

We investigate the two last steps of this method in slightly more detail.

Computation of the derivative matrix. Consider first the Jacobian matrix $\mathtt{J} = \partial\mathbf{X}'/\partial\mathbf{h}$. This matrix has a natural decomposition into blocks so that $\mathtt{J} = (\mathtt{J}_1^\mathsf{T}, \mathtt{J}_2^\mathsf{T}, \ldots, \mathtt{J}_i^\mathsf{T}, \ldots, \mathtt{J}_n^\mathsf{T})^\mathsf{T}$ where $\mathtt{J}_i = \partial\mathbf{x}'_i/\partial\mathbf{h}$. A formula for $\partial\mathbf{x}'_i/\partial\mathbf{h}$ is given in

(4.21–*p*129):

$$\mathtt{J}_i = \partial \mathbf{x}_i'/\partial \mathbf{h} = \frac{1}{w_i'} \begin{bmatrix} \tilde{\mathbf{x}}_i^\mathsf{T} & \mathbf{0}^\mathsf{T} & -x_i'\tilde{\mathbf{x}}_i^\mathsf{T} \\ \mathbf{0}^\mathsf{T} & \tilde{\mathbf{x}}_i^\mathsf{T} & -y_i'\tilde{\mathbf{x}}_i^\mathsf{T} \end{bmatrix} \tag{5.11}$$

where $\tilde{\mathbf{x}}_i^\mathsf{T}$ represents the vector $(x_i, y_i, 1)$.

Stacking these matrices on top of each other for all points $\mathbf{x}_i$ gives the derivative matrix $\partial \mathbf{X}'/\partial \mathbf{h}$. An important case is when the image measurements $\mathbf{x}_i'$ are independent random vectors. In this case $\Sigma = \mathrm{diag}(\Sigma_1, \ldots, \Sigma_n)$ where each Σ_i is the 2×2 covariance matrix of the i-th measured point $\mathbf{x}_i'$. Then one computes

$$\Sigma_{\mathbf{h}} = (\mathtt{J}^\mathsf{T} \Sigma_{\mathbf{X}'}^{-1} \mathtt{J})^+ = \left(\sum_i \mathtt{J}_i^\mathsf{T} \Sigma_i^{-1} \mathtt{J}_i \right)^+ . \tag{5.12}$$

Example 5.13. We consider the simple numerical example of a point correspondence containing just 4 points as follows:

$$\begin{aligned} \mathbf{x}_1 = (1,0)^\mathsf{T} &\leftrightarrow (1,0)^\mathsf{T} = \mathbf{x}_1' \\ \mathbf{x}_2 = (0,1)^\mathsf{T} &\leftrightarrow (0,1)^\mathsf{T} = \mathbf{x}_2' \\ \mathbf{x}_3 = (-1,0)^\mathsf{T} &\leftrightarrow (-1,0)^\mathsf{T} = \mathbf{x}_3' \\ \mathbf{x}_4 = (0,-1)^\mathsf{T} &\leftrightarrow (0,-1)^\mathsf{T} = \mathbf{x}_4' \end{aligned}$$

namely, the identity map on the points of a projective basis. We assume that points $\mathbf{x}_i$ are known exactly, and points $\mathbf{x}_i'$ have one pixel standard deviation in each coordinate direction. This means that the covariance matrix $\Sigma_{\mathbf{x}_i'}$ is the identity.

Obviously, the computed homography will be the identity map. For simplicity we normalize (scale it) so that it is indeed the identity matrix, and hence $\|\mathtt{H}\|^2 = 3$ instead of the usual normalization $\|\mathtt{H}\| = 1$. In this case, all the w_i' in (5.11) are equal to 1. The matrix $\mathtt{J}$ is easily computed from (5.11) to equal

$$\mathtt{J} = \left[\begin{array}{ccc|ccc|ccc} 1 & 0 & 1 & 0 & 0 & 0 & -1 & 0 & -1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ \hline 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & -1 & -1 \\ \hline -1 & 0 & 1 & 0 & 0 & 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 \\ \hline 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 & 0 & -1 & 1 \end{array}\right] .$$

Then

$$\mathtt{J}^\mathsf{T}\mathtt{J} = \left[\begin{array}{ccc|ccc|ccc} 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2 \\ 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4 & 0 & 0 & 0 & -2 & 0 & 0 \\ \hline 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & -2 \\ 0 & 0 & 0 & 0 & 0 & 4 & 0 & -2 & 0 \\ \hline 0 & 0 & -2 & 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -2 & 0 & 2 & 0 \\ -2 & 0 & 0 & 0 & -2 & 0 & 0 & 0 & 4 \end{array}\right] . \tag{5.13}$$

To take the pseudo-inverse of this matrix, we may use (5.9) where $\mathtt{A}$ is a matrix with columns spanning the tangent plane to the constraint surface. Since $\mathtt{H}$ is computed subject to the condition $\|\mathtt{H}\|^2 = 3$, which represents a hypersphere, the constraint surface is perpendicular to the vector $\mathbf{h}$ corresponding to the computed homography $\mathtt{H}$. A Householder matrix $\mathtt{A}$ (see section A4.1.2(*p*580)) corresponding to the vector $\mathbf{h}$ has the property that $\mathtt{A}\mathbf{h} = (0, \ldots, 0, 1)^\mathsf{T}$, so the first 8 columns of $\mathtt{A}$ (denoted $\mathtt{A}_1$)are perpendicular to $\mathbf{h}$ as required. This allows the pseudo-inverse to be computed exactly without using SVD. Applying (5.9) the pseudo-inverse is computed to be

$$\Sigma_\mathbf{h} = (\mathtt{J}^\mathsf{T}\mathtt{J})^{+\mathtt{A}_1} = \mathtt{A}_1(\mathtt{A}_1^\mathsf{T}(\mathtt{J}^\mathsf{T}\mathtt{J})\mathtt{A}_1)^{-1}\mathtt{A}_1^\mathsf{T} = \frac{1}{18}\left[\begin{array}{ccc|ccc|ccc} 5 & 0 & 0 & 0 & -4 & 0 & 0 & 0 & -1 \\ 0 & 9 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 9 & 0 & 0 & 0 & 9 & 0 & 0 \\ \hline 0 & 0 & 0 & 9 & 0 & 0 & 0 & 0 & 0 \\ -4 & 0 & 0 & 0 & 5 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 9 & 0 & 9 & 0 \\ \hline 0 & 0 & 9 & 0 & 0 & 0 & 18 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 9 & 0 & 18 & 0 \\ -1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 2 \end{array}\right]. \tag{5.14}$$

The diagonals give the individual variances of the entries of $\mathtt{H}$. △

This computed covariance is used to assess the accuracy of point transfer in example 5.14.

5.2.5 Error in both images

In the case of error in both images, computation of the covariance of the transformation is a bit more complicated. As seen in section 4.2.7(*p*101), one may define a set of $2n+8$ parameters, where 8 parameters describe the transformation matrix and $2n$ parameters $\hat{\mathbf{x}}_i$ represent estimates of the points in the first image. More conveniently, one may over-parametrize by using 9 parameters for the transformation $\mathtt{H}$. The Jacobian matrix naturally splits up into two parts as $\mathtt{J} = [\mathtt{A} \mid \mathtt{B}]$ where $\mathtt{A}$ and $\mathtt{B}$ are the derivatives with respect to the camera parameters and the points $\mathbf{x}_i$ respectively. Applying (5.10) one computes

$$\mathtt{J}^\mathsf{T}\Sigma_\mathbf{x}^{-1}\mathtt{J} = \begin{bmatrix} \mathtt{A}^\mathsf{T}\Sigma_\mathbf{x}^{-1}\mathtt{A} & \mathtt{A}^\mathsf{T}\Sigma_\mathbf{x}^{-1}\mathtt{B} \\ \mathtt{B}^\mathsf{T}\Sigma_\mathbf{x}^{-1}\mathtt{A} & \mathtt{B}^\mathsf{T}\Sigma_\mathbf{x}^{-1}\mathtt{B} \end{bmatrix}.$$

The pseudo-inverse of this matrix is the covariance of the parameter set and the top-left block of this pseudo-inverse is the covariance of the entries of $\mathtt{H}$. A detailed discussion of this is given in section A6.4.1(*p*608), where it is shown how one can make use of the block structure of the Jacobian to simplify the computation.

In example 5.13 on estimating the covariance of $\mathtt{H}$ from four points in the previous section, the covariance turns out to be $\Sigma_\mathbf{h} = 2(\mathtt{J}^\mathsf{T}\Sigma_\mathbf{x}^{-1}\mathtt{J})^+$, namely twice the covariance computed for noise in one image only. This assumes that points are measured with the same covariance in both images. This simple relationship between the covariances in the one and two-image cases does not generally hold.

5.2.6 Using the covariance matrix in point transfer

Once one has the covariance, one may compute the uncertainty associated with a given point transfer. Consider a new point $\mathbf{x}$ in the first image, not used in the computation of the transformation, $\mathtt{H}$. The corresponding point in the second image is $\mathbf{x}' = \mathtt{H}\mathbf{x}$. However, because of the uncertainty in the estimation of $\mathtt{H}$, the correct location of the point $\mathbf{x}'$ will also have associated uncertainty. One may compute this uncertainty from the covariance matrix of $\mathtt{H}$.

The covariance matrix for the point $\mathbf{x}'$ is given by the formula

$$\Sigma_{\mathbf{x}'} = \mathtt{J}_{\mathbf{h}} \Sigma_{\mathbf{h}} \mathtt{J}_{\mathbf{h}}^{\mathsf{T}} \tag{5.15}$$

where $\mathtt{J}_{\mathbf{h}} = \partial \mathbf{x}'/\partial \mathbf{h}$. A formula for $\partial \mathbf{x}'/\partial \mathbf{h}$ is given in (4.21–*p*129).

If in addition, the point $\mathbf{x}$ itself is measured with some uncertainty, then one has instead

$$\Sigma_{\mathbf{x}'} = \mathtt{J}_{\mathbf{h}} \Sigma_{\mathbf{h}} \mathtt{J}_{\mathbf{h}}^{\mathsf{T}} + \mathtt{J}_{\mathbf{x}} \Sigma_{\mathbf{x}} \mathtt{J}_{\mathbf{x}}^{\mathsf{T}} \tag{5.16}$$

assuming that there is no cross-correlation between $\mathbf{x}$ and $\mathbf{h}$, which is reasonable, since point $\mathbf{x}$ is assumed to be a new point not used in the computation of the transformation $\mathtt{H}$. A formula for the Jacobian matrix $\mathtt{J}_{\mathbf{x}} = \partial \mathbf{x}'/\partial \mathbf{x}$ is given in (4.20–*p*129).

The covariance matrix $\Sigma_{\mathbf{x}'}$ given by (5.15) is expressed in terms of the covariance matrix $\Sigma_{\mathbf{h}}$ of the transformation $\mathtt{H}$. We have seen that this covariance matrix $\Sigma_{\mathbf{h}}$ depends on the particular constraint used in estimating $\mathtt{H}$, according to (5.9). It may therefore appear that $\Sigma_{\mathbf{x}'}$ also depends on the particular method used to constrain $\mathtt{H}$. It may however be verified that these formulae are independent of the particular constraint $\mathtt{A}$ used to compute the covariance matrix $\Sigma_{\mathbf{P}} = (\mathtt{J}^{\mathsf{T}} \Sigma_{\mathbf{X}}^{-1} \mathtt{J})^{+\mathtt{A}}$.

Example 5.14. To continue example 5.13, let the computed 2D homography $\mathtt{H}$ be given by the identity matrix, with covariance matrix $\Sigma_{\mathbf{h}}$ as in (5.14). Consider an arbitrary point $(\mathbf{x}, \mathbf{y})$ mapped to the point $\mathbf{x}' = \mathtt{H}\mathbf{x}$. In this case the covariance matrix $\Sigma_{\mathbf{x}'} = \mathtt{J}_{\mathbf{h}} \Sigma_{\mathbf{h}} \mathtt{J}_{\mathbf{h}}^{\mathsf{T}}$ may be computed symbolically to equal

$$\Sigma_{\mathbf{x}'} = \begin{bmatrix} \sigma_{x'x'} & \sigma_{x'y'} \\ \sigma_{x'y'} & \sigma_{y'y'} \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 2 - x^2 + x^4 + y^2 + x^2y^2 & xy(x^2 + y^2 - 2) \\ xy(x^2 + y^2 - 2) & 2 - y^2 + y^4 + x^2 + x^2y^2 \end{bmatrix}.$$

Note that $\sigma_{x'x'}$ and $\sigma_{y'y'}$ are even functions of x and y, whereas $\sigma_{x'y'}$ is an odd function. This is a consequence of the symmetry about the x and y axes of the point set used to compute $\mathtt{H}$. Also note that $\sigma_{x'x'}$ and $\sigma_{y'y'}$ differ by swapping x and y, which is a further consequence of the symmetry of the defining point set.

As may be seen, the variance $\sigma_{x'x'}$ varies as the fourth power of x, and hence the standard deviation varies as the square. This shows that extrapolating the values of transformed points $\mathbf{x}' = \mathtt{H}\mathbf{x}$ far beyond the set of points used to compute $\mathtt{H}$ is not reliable. More specifically, the RMS uncertainty of the position of $\mathbf{x}'$ is equal to $(\sigma_{x'x'} + \sigma_{y'y'})^{1/2} = \sqrt{\text{trace}(\Sigma_{\mathbf{x}'})}$ which one finds is equal to $(1 + (x^2 + y^2)^2)^{1/2} = (1 + r^4)^{1/2}$, where r is the radial distance from the origin. Note the interesting fact that the RMS error is only dependent on the radial distance. In fact, one may verify that the probability distribution for point $\mathbf{x}'$ depends only on the radial distance of $\mathbf{x}'$, its

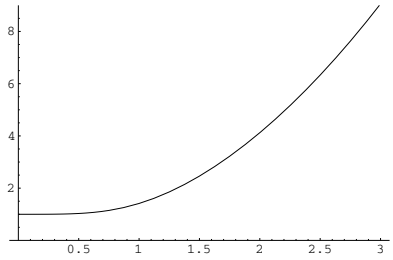

Fig. 5.5. *RMS error in the position of a projected point* $\mathbf{x}'$ *as a function of radial distance of* $\mathbf{x}'$ *from the origin. The homography* H *is computed from 4 evenly spaced points on a unit circle around the origin with errors in the second image only. The RMS error is proportional to the assumed error in the points used to compute* H*, and the vertical axis is calibrated in terms of this assumed error.*

two principal axes pointing radially and tangentially. Figure 5.5 shows the graph of the RMS error in $\mathbf{x}'$ as a function of r. △

This example has computed the covariance of a transferred point in the minimal case of four point correspondences. For more than four correspondences, the situation is not substantially different. Extrapolation beyond the set of points used to compute the homography is unreliable. In fact, one may show that if H is computed from n points evenly spaced around a unit circle (instead of 4 as in the computation above) then the RMS error is equal to $\sigma_{x'x'} + \sigma_{y'y'} = 4(1 + r^4)/n$, so the error exhibits the same quadratic growth.

5.3 Monte Carlo estimation of covariance

The method of estimating covariance discussed in the previous sections has relied on an assumption of linearity. In other words, it has been assumed that the surface $f(\mathbf{h})$ is locally flat in the vicinity of the estimated point, at least over a region corresponding to the approximate extent of the noise distribution. It has also been assumed that the method of estimation of the transformation H was the Maximum Likelihood Estimate. If the surface is not entirely flat then the estimate of covariance may be incorrect. In addition, a particular estimation method may be inferior to the ML estimate, thereby introducing additional uncertainty in the values of the estimated transformation H.

A general (though expensive) method of getting an estimate of the covariance is by exhaustive simulation. Assuming that the noise is drawn from a given noise distribution, one starts with a set of point matches corresponding perfectly to a given transformation. One then adds noise to the points and computes the corresponding transformation using the chosen estimation procedure. The covariance of the transformation H or a further transferred point is then computed statistically from multiple trials with noise drawn from the assumed noise distribution. This is illustrated for the case of the identity mapping in figure 5.6.

Both the analytical and the Monte Carlo methods of estimating covariance of the transformation H may be applied to the estimation of covariance from real data for which one does not know the true value of H. From the given data, an estimate of H and the corresponding true values of the points $\mathbf{x}'_i$ and $\mathbf{x}_i$ are computed. Then the

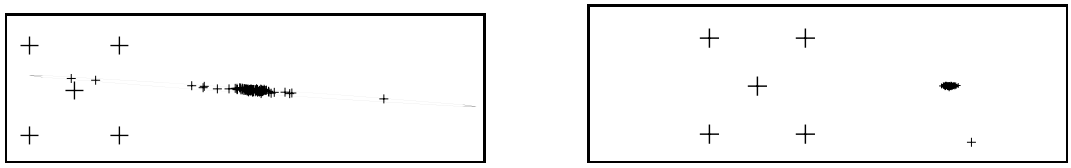

Fig. 5.6. *Transfer of a point under the identity mapping for the normalized and unnormalized DLT algorithm. See also figure 4.4(*p*109) for further explanation.*

covariance is computed as if the estimated values were the true values of the matched data points and the transformation. The resulting covariance matrix is assumed to be the covariance of the true transformation. This identification is based on the assumption that the true values of the data points are sufficiently close to the estimated values that the covariance matrix is essentially unaffected.

5.4 Closure

An extended discussion of bias and variance of estimated parameters is given in appendix 3(*p*568).

5.4.1 The literature

The derivations throughout this chapter have been considerably simplified by only using first-order Taylor expansions, and assuming Gaussian error distributions. Similar ideas (ML, covariance ...) can be developed for other distributions by using the Fisher Information matrix. Related reading may be found in Kanatani [Kanatani-96], Press *et al.* [Press-88], and other statistical textbooks.

Criminisi *et al.* [Criminisi-99b] give many examples of the computed covariances in point transfer as the correspondences used to determine the homography vary in number and position.

5.4.2 Notes and exercises

(i) Consider the problem of computing a best line fit to a set of 2D points in the plane using orthogonal regression. Suppose that N points are measured with independent standard deviations of σ in each coordinate. What is the expected RMS distance of each point from a fitted line? **Answer :** $\sigma\left((n-2)/n\right)^{1/2}$.

(ii) (Harder) : In section 18.5.2(*p*450) a method is given for computing a projective reconstruction from a set of $n+4$ point correspondences across m views, where 4 of the point correspondences are presumed to be known to be from a plane. Suppose the 4 planar correspondences are known exactly, and the other n image points are measured with 1 pixel error (each coordinate in each image). What is the expected residual error of $\|\mathbf{x}_j^i - \hat{\mathrm{P}}^i\mathbf{X}_j\|$?