Do we really have to consider covariance matrices for image features?

Yasushi Kanazawa
Department of Knowledge-based Information Engineering
Toyohashi University of Technology
Toyohashi, Aichi 441-8580 Japan
kanazawa@tutkie.tut.ac.jp

Kenichi Kanatani
Department of Computer Science
Okayama University
Okayama 700-8535 Japan
kanatani@suri.it.okayama-u.ac.jp

Abstract

Many studies have been made in the past for optimization using covariance matrices of feature points. We first describe how to compute the covariance matrix of a feature point from the gray levels by integrating existing methods. Then, we experimentally examine if thus computed covariance matrices really reflect the accuracy of the feature points. To test this, we do subpixel template matching and compute the homography and the fundamental matrix. Our conclusion is rather surprising, pointing out important elements often overlooked.

1. Introduction

Detecting feature points is a first step in many vision applications such as 3-D reconstruction and image mosaicing. Since the detected feature points have some uncertainty, many proposed algorithms for geometric computation involve optimization based on the covariance matrices of the feature points [3, 8, 10, 11]. And various techniques have been proposed for evaluating the covariance matrix from the gray levels [1, 11, 14, 15]. Here, we raise two questions:

1. *Does the covariance matrix thus computed really reflect the accuracy of feature locations?*
2. *Does the accuracy of geometric computation really improve by using the covariance matrix computed from the gray levels?*

Surely, the effectiveness of using covariance matrices can be confirmed by simulations, but this is rather obvious because the noise is generated *according to the assumed covariance matrices*. For a real image, no obvious relationship exists between feature uncertainty and the gray levels.

In this paper, we first describe how to compute the covariance matrix of a feature point from the gray levels by integrating existing methods. Then, we experimentally examine if thus computed covariance matrix really reflects the feature uncertainty by doing subpixel template matching and computing the homography and the fundamental matrix. Our conclusion is rather surprising, pointing out important elements often overlooked.

2. Covariance Matrices

Feature points, whether they are extracted by an image operator [4, 16] or by hand, have some uncertainty. Let $(\bar{x}, \bar{y})$ be the true position of a feature point (x, y). If the errors $\Delta x = x - \bar{x}$ and $\Delta y = y - \bar{y}$ are regarded as random variables, their covariance matrix is written as

$$\begin{pmatrix} E[\Delta x^2] & E[\Delta x \Delta y] \\ E[\Delta y \Delta x] & E[\Delta y^2] \end{pmatrix}, \tag{1}$$

where $E[\,\cdot\,]$ denotes expectation. In the following, we express the covariance matrix in the form $\sigma^2 \Sigma^0$. This is because covariance matrices can usually be evaluated only up to scale. We call σ the *noise level*, and Σ^0 the *normalized covariance matrix*.

In most geometric computations, the optimal solution depends only on the normalized covariance matrix Σ^0 [3, 8, 10, 11]. If the noise characteristics do not depend on position or orientation, it suffices to use the default value

$$\Sigma^0 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}. \tag{2}$$

Otherwise, we need to compute Σ^0 from the gray levels. Basically, there exist two approaches for this: the residual-based approach [11, 14] and the derivative-based approach [1, 15].

3. Residual-Based Approach

Let $I(i, j)$ be the gray level of pixel (i, j). We evaluate the residual of self-matching in the form

$$J(x, y) = \frac{1}{2} \sum_{(i,j) \in \mathcal{N}_p} w_{ij} \Big(I(i + x, j + y) - I(i, j) \Big)^2, \tag{3}$$

where $\mathcal{N}_p$ is a square neighborhood centered on the point p in question, and w_{ij} is an appropriate (say, Gaussian) weight. If we regard $I(i, j)$ as a continuous function via an appropriate interpolation, Eq. (3) is a function of real numbers x and y. We approximate it by a quadratic function over a neighborhood $\mathcal{X}$ of the origin $(0, 0)$ in the form

$$\begin{aligned} g(x, y) &= \frac{1}{2}(n_1 x^2 + 2 n_2 xy + n_3 y^2) \\ &= \frac{1}{2} \begin{pmatrix} x & y \end{pmatrix} H \begin{pmatrix} x \\ y \end{pmatrix}, \end{aligned} \tag{4}$$

where we define the Hessian H by

$$H = \begin{pmatrix} n_1 & n_2 \\ n_2 & n_3 \end{pmatrix}. \tag{5}$$

The coefficients n_1, n_2, and n_3 are determined by least squares in the form

$$\iint_{\mathcal{X}} w(x,y)\Big(J(x,y) - g(x,y)\Big)^2 dxdy \to \min, \tag{6}$$

where $w(x,y)$ is a positive weight; we may use the Gaussian function $w(x,y) = e^{-(x^2+y^2)/\sigma^2}$ or the Gibbs function $w(x,y) = e^{-J(x,y)/\sigma^2}$, where σ is an appropriate constant.

Differentiating Eq. (6) with respect to x and y and letting the result be 0, we obtain the following normal equation:

$$\frac{1}{2}\boldsymbol{A}\boldsymbol{n} = \boldsymbol{b}. \tag{7}$$

The matrix $\boldsymbol{A}$ and the vector $\boldsymbol{b}$ are defined by

$$\boldsymbol{A} = \iint_{\mathcal{X}} w(x,y)\boldsymbol{m}(x,y)\boldsymbol{m}(x,y)^\top dxdy,$$

$$\boldsymbol{b} = \iint_{\mathcal{X}} w(x,y)J(x,y)\boldsymbol{m}(x,y)dxdy, \tag{8}$$

where $\boldsymbol{m}(x,y) = (x^2, 2xy, y^2)^\top$. The integration is done numerically. From the resulting solution $\boldsymbol{n}$, we compute the Hessian H by Eq. (5). The normalized covariance matrix Σ^0 is defined as the inverse of the Hessian H [11, 14]:

$$\Sigma^0 = H^{-1}. \tag{9}$$

4. Derivative-Based Approach

If the neighborhood $\mathcal{X}$ is small, we can approximate the function J via a Taylor expansion in the form

$$J = \frac{1}{2}\sum_{(i,j)\in\mathcal{N}_p} w_{ij}(I_i x + I_j y)^2 = \frac{1}{2}\begin{pmatrix} x & y \end{pmatrix} H \begin{pmatrix} x \\ y \end{pmatrix}, \tag{10}$$

where I_i and I_j denote partial derivatives of I. The Hessian H now takes the following form:

$$H = \begin{pmatrix} \sum_{(i,j)\in\mathcal{N}_p} w_{ij} {I_i}^2 & \sum_{(i,j)\in\mathcal{N}_p} w_{ij} I_i I_j \\ \sum_{(i,j)\in\mathcal{N}_p} w_{ij} I_j I_i & \sum_{(i,j)\in\mathcal{N}_p} w_{ij} {I_j}^2 \end{pmatrix}. \tag{11}$$

The normalized covariance matrix Σ^0 is defined as its inverse [1, 15]:

$$\Sigma^0 = H^{-1} \tag{12}$$

The derivatives I_i and I_j are computed by applying appropriate differentiation filters [2].

5. Justification

Both approaches define the normalized covariance matrix as the inverse of the Hessian H; the only difference is whether it is computed globally over the neighborhood $\mathcal{X}$ or locally via the Taylor expansion.

The Hessian H defines the curvature of the residual surface (Fig.1). A large curvature implies a large variation in the gray levels around p, so it would be easy to locate the feature position by template matching. A small curvature implies almost uniform gray levels around p, so it would be difficult to locate the position by template matching. Thus, it makes sense to regard the inverse of the Hessian H as indicating the feature uncertainty.

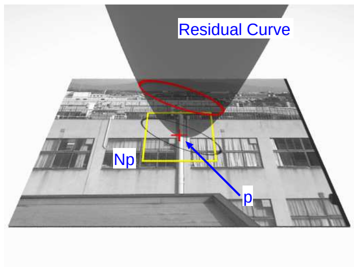


Figure 1: The residual surface.

Mathematically, the problem is formulated as follows. Given a function $I(u,v)$ of two real variables and data $\{I_{ij}\}$, $(i,j) \in \mathcal{N}_p$, we are to find (x,y) such that $I_{ij} \approx I(i+x, j+y)$, $(i,j) \in \mathcal{N}_p$. This problem is regarded as statistical inference if we introduce the model

$$I_{ij} = I(i+x, j+y) + \varepsilon_{ij}, \tag{13}$$

where ε_{ij} is independent Gaussian noise of zero mean. Suppose the relation $I_{ij} \approx I(i+x, j+y)$, $(i,j) \in \mathcal{N}_p$ holds better as (i,j) is closer to the center of $\mathcal{N}_p$. This can be modeled by assuming that ε_{ij} has standard deviation σ_{ij} depending on (i,j). The maximum likelihood estimation is obtained by minimizing

$$\frac{1}{2}\sum_{(i,j)\in\mathcal{N}_p} w_{ij}\Big(I_{ij} - I(i+x, j+y)\Big)^2, \tag{14}$$

where $w_{ij} = \sigma^2/\sigma_{ij}^2$. If the true solution is $(x,y) = 0$, the matrix H given by Eq. (11) can be interpreted to be the *Fisher information matrix*, hence the matrix $\Sigma^{(0)}$ to be the *Cramer-Rao lower bound* on the normalized covariance matrix [5].

The assumption that their normalized covariance matrix can be approximated by the inverse of the Hessian H is equivalent to saying that they are approximately subject to a Gaussian distribution of the form

$$p(x,y) \approx \frac{1}{2\pi\sigma^2} e^{-\begin{pmatrix} x & y \end{pmatrix} H \begin{pmatrix} x \\ y \end{pmatrix}/2\sigma^2}. \tag{15}$$

From the definition of the Hessian H, this is also equivalent to assuming that

$$p(x,y) \approx \frac{1}{2\pi\sigma^2} e^{-J(x,y)/\sigma^2}. \tag{16}$$

It has been empirically known that the distribution of the solution that minimizes the least-squares residual, often referred to as *energy* on the analogy to physics, can be approximated by a Gibbs distribution in this form [9]. Indeed,

(a) (b) (c)

Figure 2: (a) Feature points and their covariance matrices. (b) Feature points chosen by hand. (c) Feature points chosen by SUSAN.

Singh [14] estimated the normalized covariance matrix Σ^0 as the covariance matrix of the values $e^{-J(x,y)/\sigma^2}$ sampled from the neighborhood $\mathcal{X}$.

If $I(i+x, j+y)$ can be approximated by its Taylor expansion to the first order, we have from Eq. (14)

$$\frac{1}{2}\sum_{(i,j)\in\mathcal{N}_p} w_{ij}\Big(I_{ij} - I - I_i x - I_j y\Big)^2 \to \min. \qquad (17)$$

This is nothing but detecting optical flow by imposing the *gradient constraints*

$$I_i x + I_j y = I_{ij} - I \qquad (18)$$

in the weighted least-squares sense. The matrix Σ^0 given by Eq. (12) computed from the Hessian J given by Eq. (11) can be regarded as the covariance matrix of the detected flow [12]. If the Hessian H happens to be singular, the gray levels are uniform in at least one direction, causing what is known as the *aperture problem*.

6. Real Image Covariance Matrices

The above analysis may sound satisfactory, but there is a wide gap between theoretical idealization and reality. The question is: *Can we really connect the gray levels to the covariance matrix that characterizes stochastic properties?* We now examine this experimentally and point out important elements that we often overlook.

Fig. 2(a) shows an image of an outdoor scene. We arbitrarily placed a 3×3 grid in it and computed the covariance matrices at the vertices; they are displayed as ellipses centered on the vertices, indicating the standard deviation along each orientation (they are appropriately scaled). Those in solid lines were computed by the residual-based method; those in dashed lines by the derivative-based method. As we see, they are large in homogeneous regions, small in regions with fine structures, and elongated along object boundaries.

Now observe Fig. 2(b), where the feature points are chosen by hand. The resulting ellipses are more or less circles of uniform size. This is more conspicuous when a feature detector is used. Fig. 2(c) displays the covariance matrices of the points detected by SUSAN [16].

This observation is explained as follows. When required to choose feature points, humans tend to look for salient points such as corners and isolated points, avoiding points in homogeneous regions or on edges [13]. Feature detectors are so designed as to find points around which there are large variations in the gray levels. Thus, both humans and feature detectors implicitly evaluate the covariance matrix and find "good" features with structures of almost the same degree in all directions. For example, the Harris operator [4] directly computes the Hessian H given by Eq. (11). The Kanade-Lucas-Tomasi tracker also computes Eq. (11) in the course of tracking [15, 17].

It follows that *as long as feature points are chosen by hand or a feature detector*, it is practically sufficient to use the default value (2); the covariance matrix need not be evaluated from the gray levels (the evaluation has already been done by the human brain and the feature detector). We conclude that it is when feature points are specified *randomly* or *independently of the image content* that we need to evaluate the covariance matrix from the gray levels.

7. Similarity Template Matching

The next question is: When a feature point is specified randomly or independently of the image content, *does the covariance matrix computed from the gray levels really indicate the accuracy of locating it?* To examine this, we conduct template matching experiments.

Using two images of the same scene, we match a randomly specified point p in one image to the other. Cutting out a neighborhood $\mathcal{N}_p$ of p, we define a template $T(i,j)$ centered on it and do variable template matching by taking account of rotations and scale changes.

We place the template $T(i,j)$ at a pixel (a,b) close to the corresponding point in the other image $I(i,j)$ and adjust the translation, the rotation, and the scale (in real applications, such an approximate point can be found by a coarse-to-fine strategy, but this is irrelevant in our context, so we omit this stage). Suppose the template $T(i,j)$ matches the image $I(i,j)$ if it is translated by (x,y), rotated by angle θ, and scaled s times from this position. The values (x,y), θ, and s are determined by minimizing

$$J(x,y,\theta,s) = \sum_{(i,j)\in\mathcal{N}_p} w_{ij}\Big(T(i,j) - \mathcal{T}^{(a,b)}_{(x,y,\theta,s)} I(i,j)\Big)^2, \qquad (19)$$

(a) (b)

Figure 3: (a) Two images of an outdoor scene. The left grid is mapped to the right image and corrected with subpixel accuracy. (b) Correlation between the covariance matrix and the template matching accuracy: the residual-based approach ($\times$); the flow-based approach ($+$).

where we define the similarity mapping $\mathcal{T}^{(a,b)}_{(x,y,\theta,s)}$ by

$$\mathcal{T}^{(a,b)}_{(x,y,\theta,s)} I(i,j) = I(a+x+s(i\cos\theta - j\sin\theta), \\ b+y+s(i\sin\theta + j\cos\theta)). \qquad (20)$$

We numerically search the four-dimensional parameter space of x, y, θ, and s for the minimum by recursively subdividing the search region.

8. Accuracy of Feature Matching

Fig. 3(a) shows two images of an outdoor scene. We arbitrarily placed a 3×3 grid in the left image; the ellipses at the vertices describe their covariance matrices (those in solid lines were computed by the residual-based method; those in dashed lines by the derivative-based method). In the right image, we manually defined a grid (in solid lines) that approximately corresponds to the grid in the left image and corrected each vertex by the similarity template matching. The corrected grid is shown in dashed lines.

We did this type of matching experiment many times in different feature locations and in different images. We measured the matching accuracy by the deviation $(\Delta x, \Delta y)$ from the true position $(\bar{x}, \bar{y})$ to the corrected position (x, y). The true position $(\bar{x}, \bar{y})$ was estimated by noting that the scene is very far away and the two images are related by a homography. We carefully chose by hand 53 feature points (not shown here), from which we computed the homography (the details are given shortly). We mapped the vertices in the left image to the right image via the computed homography and regarded the resulting positions as the true corresponding points.

Fig. 3(b) plots on logarithmic scales the absolute deviation $\sqrt{\Delta x^2 + \Delta y^2}$ vs. the root-mean-square error $\sqrt{\mathrm{tr}\Sigma^0}$ ($\propto \sqrt{E[\Delta x^2 + \Delta y^2]}$) predicted from the gray levels. As we can see, the plots cluster roughly along a line with slope 1, although there are large deviations. This implies that the accuracy of template matching has more or less positive correlation with the covariance matrix computed from the gray levels. We conclude that the covariance matrix computed from the gray levels *does* indicate the feature uncertainty.

9. Covariance-Based Optimization

If the covariance matrix computed from the gray levels indicates the feature uncertainty, this does not mean that they are useful in improving the accuracy of geometric computation. The remaining question is: *Does the covariance matrix computed from the gray levels really improve the accuracy of the optimization using them?* We now examine this by computing the homography and the fundamental matrix from two real images.

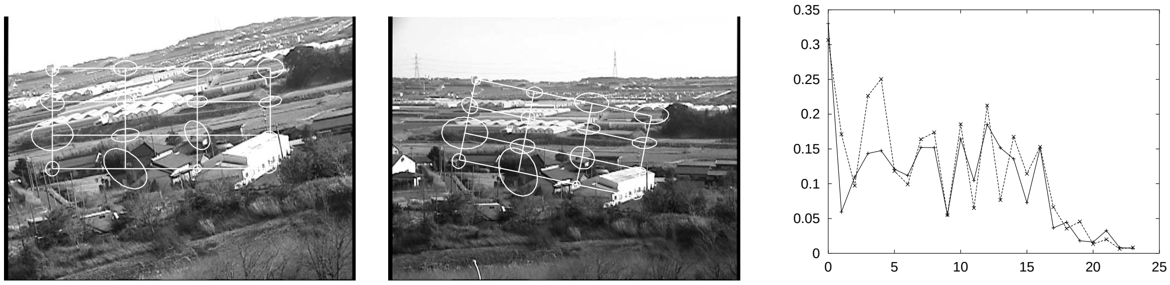
Suppose we are given two sets of corresponding points $\{(x_\alpha, y_\alpha)\}$ and $\{(x'_\alpha, y'_\alpha)\}$ and their covariance matrices $\{\Sigma^0_\alpha\}$ and $\{\Sigma^0_\alpha{}'\}$, α = 1, ..., N. We represent the points by vectors

$$\boldsymbol{x}_\alpha = \begin{pmatrix} x_\alpha/f_0 \\ y_\alpha/f_0 \\ 1 \end{pmatrix}, \qquad \boldsymbol{x}'_\alpha = \begin{pmatrix} x'_\alpha/f_0 \\ y'_\alpha/f_0 \\ 1 \end{pmatrix}, \qquad (21)$$

where f_0 is an appropriate scale factor chosen so that x_α/f_0, y_α/f_0, x'_α/f_0, and y'_α/f_0 have order 1. Since the third components are 1, their normalized covariance matrices are singular matrices of rank 2 in the form

$$V_0[\boldsymbol{x}_\alpha] = \begin{pmatrix} \Sigma^0_\alpha & \boldsymbol{0} \\ \boldsymbol{0}^\top & 0 \end{pmatrix}, \quad V_0[\boldsymbol{x}'_\alpha] = \begin{pmatrix} \Sigma^0_\alpha{}' & \boldsymbol{0} \\ \boldsymbol{0}^\top & 0 \end{pmatrix}. \qquad (22)$$

According to the discussion in Section 5, the derivative-based approach may be more consistent with the mathematical justification than the residual-based approach. In actual computation, however, differentiation must be approximated by a difference filter operation, which is susceptible to image noise and greatly influenced by the associated smoothing operation. The residual-based approach, on the other hand, relies on integration, which is stably approximated by summation, and is more robust than the derivative-based approach. Since no marked difference is found in the final results, we adopt the residual-based approach in the subsequent experiments.



(a) (b)

Figure 4: (a) Corresponding points and their covariance matrices. (b) Errors in the homography computation using the covariance matrices evaluated from the gray levels (solid lines) and their default values (dashed lines).

10. Homography Computation

If we take two images of a planar object or a far-away scene, the two images are related by a mapping called *homography*: the corresponding points are related by

$$\boldsymbol{x}'_\alpha = Z[\boldsymbol{H}\boldsymbol{x}_\alpha], \tag{23}$$

where $\boldsymbol{H}$ is a nonsingular matrix, and $Z[\,\cdot\,]$ designates normalization to make the third component 1. The matrix $\boldsymbol{H}$ is also called the *homography* whenever no confusion can arise. In the presence of Gaussian noise, an optimal estimate of $\boldsymbol{H}$ is obtained by minimizing the following function [5]:

$$J(\boldsymbol{H}) = \frac{1}{N}\sum_{\alpha=1}^{N}(\boldsymbol{x}'_\alpha \times \boldsymbol{H}\boldsymbol{x}_\alpha, \boldsymbol{W}_\alpha(\boldsymbol{x}'_\alpha \times \boldsymbol{H}\boldsymbol{x}_\alpha)), \tag{24}$$

$$\boldsymbol{W}_\alpha = \Big(\boldsymbol{x}'_\alpha \times \boldsymbol{H}V_0[\boldsymbol{x}_\alpha]\boldsymbol{H}^\top \times \boldsymbol{x}'_\alpha + (\boldsymbol{H}\boldsymbol{x}_\alpha) \times V_0[\boldsymbol{x}'_\alpha] \times (\boldsymbol{H}\boldsymbol{x}_\alpha)\Big)^-_2. \tag{25}$$

Here, $(\boldsymbol{a}, \boldsymbol{b})$ denotes the inner product of vectors $\boldsymbol{a}$ and $\boldsymbol{b}$; $\boldsymbol{a} \times \boldsymbol{A}$ is the matrix whose columns are the vector products of $\boldsymbol{a}$ and the columns of $\boldsymbol{A}$; $\boldsymbol{A} \times \boldsymbol{a}$ is the matrix whose rows are the vector products of $\boldsymbol{a}$ and the rows of $\boldsymbol{A}$. The operation $(\,\cdot\,)^-_r$ means the (Moore-Penrose) generalized inverse computed after replacing the smallest $n - r$ eigenvalues by zeros.

Using the same images in Fig. 4 (the order was interchanged), we arbitrarily placed a 3×3 grid in the left image and mapped it to a roughly corresponding position in the right image by hand. Then, we corrected the vertices by the similarity template matching. The covariance matrices computed from the gray levels are displayed as ellipses at the vertices.

We optimized the homography in two ways using the technique given in [7]: we obtained the solution $\boldsymbol{H}$ using the covariance matrices computed from the gray levels and the solution $\boldsymbol{H}_0$ using the default value (2). We see that

$$\|\boldsymbol{H} - \bar{\boldsymbol{H}}\| = 0.00718999, \quad \|\boldsymbol{H}_0 - \bar{\boldsymbol{H}}\| = 0.00835789,$$

where $\bar{\boldsymbol{H}}$ is the true value, which was estimated by using 53 points carefully chosen by hand (not shown here). All matrices are normalized to unit norm.

Fig. 4(b) shows 24 instances of $\|\boldsymbol{H} - \bar{\boldsymbol{H}}\|$ and $\|\boldsymbol{H}_0 - \bar{\boldsymbol{H}}\|$ for different images and different grid positions. We see that the accuracy increases more or less for $\boldsymbol{H}$ over $\boldsymbol{H}_0$, while no noticeable increase in accuracy was observed when feature points were detected by hand or by a feature detector. We conclude that although the gain may be slight, the accuracy of the homography computation *does* improve using the covariance matrices from the gray levels *if feature points are given randomly or independently of the image content*,

11. Fundamental Matrix Computation

Corresponding points in two images of the same scene satisfy the following *epipolar equation*:

$$(\boldsymbol{x}_\alpha, \boldsymbol{F}\boldsymbol{x}'_\alpha) = 0. \tag{26}$$

Here, $\boldsymbol{F}$ is a singular matrix of rank 2 called the *fundamental matrix*. In the presence of Gaussian noise, an optimal estimate of $\boldsymbol{F}$ is obtained by minimizing the following function [5]:

$$J(\boldsymbol{F}) = \frac{1}{N}\sum_{\alpha=1}^{N} W_\alpha(\boldsymbol{x}_\alpha, \boldsymbol{F}\boldsymbol{x}'_\alpha)^2, \tag{27}$$

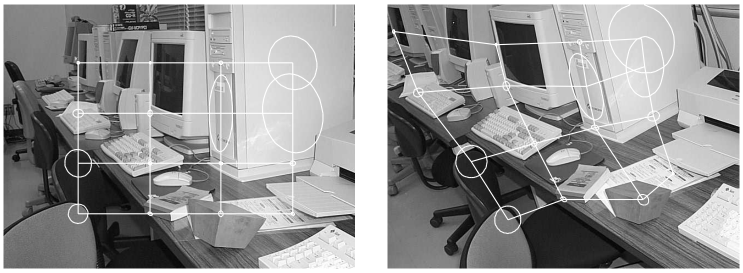
$$W_\alpha = \frac{1}{(\boldsymbol{x}'_\alpha, \boldsymbol{F}^\top V_0[\boldsymbol{x}_\alpha]\boldsymbol{F}\boldsymbol{x}'_\alpha) + (\boldsymbol{x}_\alpha, \boldsymbol{F}V_0[\boldsymbol{x}'_\alpha]\boldsymbol{F}^\top \boldsymbol{x}_\alpha)}. \tag{28}$$

Fig. 5 shows two images of an indoor scene. We placed a 3×3 grid in the left image and mapped it to a roughly corresponding position in the right image by hand. Then, we corrected the vertices by the similarity template matching. The covariance matrices computed from the gray levels are displayed as ellipses at the vertices.

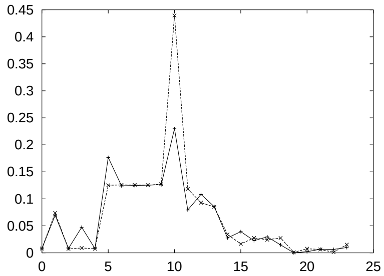
We optimized the fundamental matrix in two ways using the technique given in [6]: we obtained the solution $\boldsymbol{F}$ using the covariance matrices computed from the gray levels and the solution $\boldsymbol{F}_0$ using the default value (2). We see that

$$\|\boldsymbol{F} - \bar{\boldsymbol{F}}\| = 0.0098058, \quad \|\boldsymbol{F}_0 - \bar{\boldsymbol{F}}\| = 0.0151409,$$

where $\bar{\boldsymbol{F}}$ is the true value, which was estimated by using 68 points carefully chosen by hand (not shown here). All matrices are normalized to unit norm.



(a)



(b)

Figure 5: (a) Covariance matrices and corresponding points. (b) Errors in the fundamental matrix computation using the covariance matrices evaluated from the gray levels (solid lines) and their default values (dashed lines).

Fig. 5(b) shows 24 instances of $\|\boldsymbol{F} - \bar{\boldsymbol{F}}\|$ and $\|\boldsymbol{F}_0 - \bar{\boldsymbol{F}}\|$ for different images and different grid positions. Again, the accuracy increases more or less for $\boldsymbol{F}$ over $\boldsymbol{F}_0$, while no noticeable increase in accuracy was observed when feature points were detected by hand or by a feature detector. We conclude that although the gain may be slight, the accuracy of the fundamental matrix computation *does* improve using the covariance matrices from the gray levels *if feature points are given randomly or independently of the image content*.

12. Concluding Remarks

We described how to compute the covariance matrix of a feature point from the gray levels by integrating existing methods. Then, we experimentally examined if the thus computed covariance matrix really reflects the feature uncertainty by doing subpixel template matching and computing the homography and the fundamental matrix. Our conclusion is as follows:

1. It suffices to assume the default value (2) if feature points are chosen by hand or by a feature detector.
2. If feature points are specified randomly or independently of the image content, then:

(a) the covariance matrix computed from the gray levels have positive correlation with the accuracy of locating the feature position by template matching;
(b) the accuracy of geometric computation can improve by optimization based on the covariance matrix computed from the gray levels;
(c) the correlation and the accuracy improvement are very small, however.

So, we need not worry about the covariance matrix for 3-D reconstruction using a feature detector. On the other hand, if we are to reconstruct the 3-D shape of a surface (e.g, a human face in video images or a mountain in aerial images), a rectangular or triangular mesh is defined in one image and adjusted in the other image by template matching. In such applications, the covariance matrix computed from the gray levels plays a positive role in increasing the accuracy of 3-D computation.

References

[1] W. Förstner, Reliability analysis of parameter estimation in linear models with applications to mensuration problems in computer vision, *Comput. Vision Graphics Image Process.*, **40** (1987), 273–310.

[2] M. Elad, P. Teo and Y. Hel-Or, Optimal filters for gradient-based motion estimation, *Proc. 7th Int. Conf. Comput. Vision*, September 1999, Kerkya, Greece, pp. 559–565.

[3] M. Irani and P. Anandan, Factorization with uncertainty, *Proc. 6th Euro. Conf. Comput. Vision*, June–July 2000, Dublin, Ireland, Vol. 1, pp. 539–553.

[4] C. Harris and M. Stephens, A combined corner and edge detector, *Proc. 4th Alvey Vision Conf.*, August 1988, Manchester, pp. 147–151.

[5] K. Kanatani, *Statistical Optimization for Geometric Computation: Theory and Practice*, Elsevier Science, Amsterdam, 1996.

[6] K. Kanatani, Optimal fundamental matrix computation: Algorithm and reliability analysis, *Proc. 6th Symp. Sensing via Image Information*, June 2000, Yokohama, Japan, pp. 291 – 298.

[7] K. Kanatani, N. Ohta, and Y. Kanazawa, Optimal homography computation with a reliability measure, *IEICE Trans. Inf. & Syst.*, **E83-D**-7 (2000), 1369–1374.

[8] Y. Kanazawa and K. Kanatani, Stabilizing image mosaicing by model selection, *2nd Workshop 3D Structure from Multiple Images of Large-Scale Environments and Applications to Virtual and Augmented Reality*, July 2000, Dublin, Ireland, pp. 10–17.

[9] S. Z. Li, *Markov Random Field Modeling in Computer Vision*, Springer, Tokyo, 1995.

[10] C. Matsunaga and K. Kanatani, Calibration of a moving camera using a planer pattern: Optimal computation, reliability evaluation and stabilization by model selection, *Proc. 6th Euro. Conf. Comput. Vision*, June–July 2000, Dublin, Ireland, Vol. 2, pp. 595–609,

[11] D. D. Morris and T. Kanade, A unified factorization algorithm for points, line segments and planes with uncertainty models, *Proc. 6th Int. Conf. Comput. Vision*, January 1998, Bombay, India, pp. 696–702.

[12] N. Ohta, Image movement detection with reliability indices, *IEICE Trans.*, **E74**-10 (1991), 3379–3388.

[13] D. Reisfeld, H. Wolfson and Y. Yeshurun, Contex-free attentional operators: The generalized symmetry transform, *Int. J. Comput. Vision*, **14** (1995), 119–130.

[14] A. Singh, An estimation-theoretic framework for image-flow computation, *Proc. 3rd Int. Conf. Comput. Vision*, December 1990, Osaka, Japan, pp. 168–177.

[15] J. Shi and C. Tomasi, Good features to track, *Proc. IEEE Conf. Comput. Vision Patt. Recogn.*, June 1994, Seattle, WA, pp. 593–600.

[16] S. M. Smith and J. M. Brady, SUSAN—A new approach to low level image processing, *Int. J. Comput. Vision*, **23**-1 (1997), 45–78.

[17] C. Tomasi and T. Kanade, *Detection and Tracking of Point Features*, CMU Tech. Rep. CMU-CS-91-132, April 1991; http://vision.stanford.edu/ birch/klt/.