Symmetry Detection Through Local Skewed Symmetries

Tat-Jen Cham and Roberto Cipolla

CUED/F-INFENG/TR 183

November 1994

(Revised from **TR 165**, Feb 1994)

Note: A journal version of this report will appear in the Image and Vision Computing Journal in 1995.

Department of Engineering,
University of Cambridge,
Cambridge CB2 1PZ
England.

Email: tjc@eng.cam.ac.uk, cipolla@eng.cam.ac.uk

Symmetry Detection Through Local Skewed Symmetries

Tat-Jen Cham and Roberto Cipolla

Department of Engineering
University of Cambridge
Cambridge CB2 1PZ, England

November 1994

We explore how global symmetry can be detected prior to segmentation and under noise and occlusion. The definition of local symmetries is extended to affine geometries by considering the tangents and curvatures of local structures, and a quantitative measure of local symmetry known as *symmetricity* is introduced, which is based on Mahalanobis distances from the tangent-curvature states of local structures to the local skewed symmetry state-subspace. These symmetricity values, together with the associated local axes of symmetry, are spatially related in the *local skewed symmetry field (LSSF)*. In the implementation, a fast, local symmetry detection algorithm allows initial hypotheses for the symmetry axis to be generated through the use of a modified Hough transform. This is then improved upon by maximising a global symmetry measure based on accumulated local support in the LSSF — a straight active contour model is used for this purpose. This produces useful estimates for the axis of symmetry *and* the angle of skew in the presence of contour fragmentation, artifacts and occlusion.

1 Motivation

Symmetry, like the other properties described in the Gestalt principles of perceptual grouping [22], represents information redundancy which may be used to overcome noise and occlusion. The reason why it is rarely used in computational perceptual organisation [19, 22] is that symmetry is much harder to detect compared to other grouping properties such as continuity and token similarity. Nevertheless, our visual system seems to be able to recover symmetry from images even when the symmetry is badly distorted [2].

Two dimensional symmetry can be exploited in various ways in computer vision. For example, the symmetry axis transform [3] and smoothed local symmetries [4] are intended for 2-D shape description. Other applications to symmetry detection, or asymmetry detection, range from medical image analysis [24] to feature detection [27].

Much of the research literature is concerned with bilateral symmetry detection. Abstract mathematical theory [1, 10] has been used but in most cases is only intended to handle perfect set of figures. On the other hand Marola [20] catered for deviations from perfect symmetry. Others adopt different approaches, eg. Labonté *et al.* [18] who detected symmetry in textured patterns over a number of stages. Directional correlations were used by Masuda *et al.* [21] to extract partial symmetry, while Reisfeld *et al.* [27] employed a low-level symmetry operator to detect small salient features.

Under normal viewing conditions however, we are more likely to encounter skewed symmetry [16], which is planar bilateral symmetry transformed affinely when viewed under *weak* perspective [29].

Brady and Yuille [5] proposed maximising a compactness measure to backproject a closed image contour displaying skewed symmetry, ie. parallelograms are interpreted as squares, and ellipses as circles. Friedberg [11, 12] described a method based on moments to detect skewed symmetry axes of 2-D shapes, which assumes prior segmentation of objects and hence difficult to apply to real images. Glachet *et al.* [13] proposed a technique for recovering symmetric axes in line drawings under full perspective projection, involving recovery of accurate straight line segments from the convex-hull of the figure, pairing them and predicting the symmetry parameters through the use of cross-ratios. Ponce [26] exploited straight-spined Brooks Ribbons and their curvature relationships to pick out symmetrical pairs of points. Yuen [33] suggested carrying out Hough Transforms on the sets of mid-points of point-pairs lying at a particular orientation and doing this over a series of discretized orientations, but did not implement it. Mukherjee *et al.* [25] exploited bitangents in objects to define affine bases, from which relative invariants are used to compare pairs of contours with bitangents for symmetry. The transformation matrix relating the affine bases is then tested for conformity with the 3-DOF structure required for affine-transformed reflections. Van Gool *et al.* constructed the *Arc Length Space (ALS)* [32] to analyse symmetric contours, and also studied their invariant signatures [31]. A more detailed survey of these methods may be found in [7].

Many of these methods apply only to objects which are easy to segment, or are special cases, and do not consider circumstances when symmetry will be useful to perceptual grouping, such as in cases where objects are occluded, or have fragmented contours.

In this paper a local approach to skewed symmetry detection is adopted to handle occlusion. Sensitivity to noise inherent in local measurements is coped with through

- non-committal classification,
- uncertainty analysis, and
- arriving at global decisions by accumulation of independent, local support.

In particular we

- review the number of parameters (ie. degrees of freedom) needed to specify symmetry bases under different geometries,

- extend the classification of local symmetries to affine transformations based on the number of such parameters,
- define a continuous measure of local symmetry (*symmetricity*) rather than a binary accept/reject threshold,
- spatially represent these measures for all pairs of contour points in the form of the *local skewed symmetry field*,
- develop and implement a three stage symmetry detection scheme designed to handle noise and occlusion, based on the above theories as well as introducing a separate, novel symmetry detection algorithm as the first stage of the process.

2 Theoretical Framework

2.1 Transformations of 2D Reflectional Symmetry

Image contours of planar objects may be considered under three classes of 2D transformations: *scaled Euclidean*, *affine* or *projective*. A 2D scaled Euclidean transformation requires *four* parameters for unique specification (scale, rotation and translation in 2D), an affine transformation requires *six* (including magnitude and axis of pure shear) and a projective transformation requires *eight* (additional location of horizon to specify the perspective distortion).

As symmetry is similar to having two views of half the object, symmetry bases under these transformations require less parameters to specify:

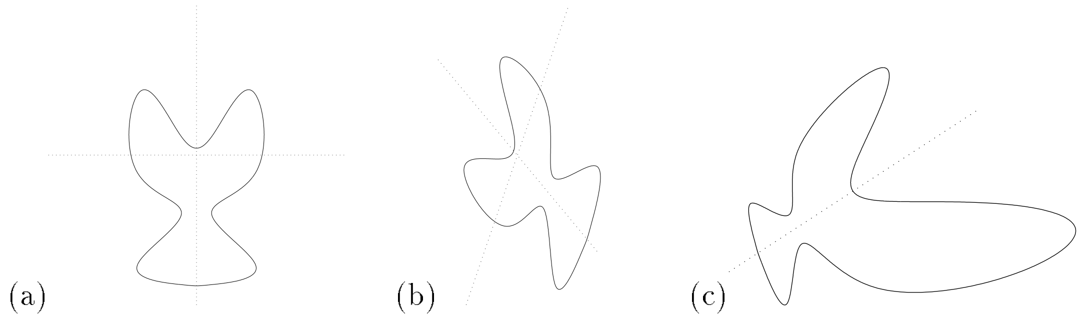


Figure 1: *Symmetry under various transformations: the figures show the same symmetrical shape under different transformations: (a) scaled Euclidean, (b) affine, and (c) projective. Symmetry bases under these transformations require 2, 3 and 4 parameters respectively for unique specification.*

Scaled Euclidean Two parameters are required for the axis of symmetry. The lines joining symmetric points are perpendicular to this line.

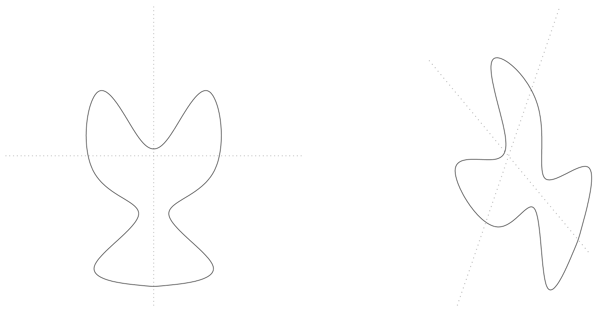
Affine Three parameters are required to additionally define the angle of skew, which is not perpendicular to the axis of symmetry.

Projective Four parameters are needed because instead of an angle of skew, we have a vanishing point for lines of skew.

A symmetric shape under these transformations is shown in figure 1. Since skewed symmetry and stereo are similar, the two cases share similar difficulties such as the correspondence problem. A common task after identifying a shape as belonging to affine or projective symmetry is to find the transformation under its class which maps the shape to one exhibiting scaled Euclidean symmetry. This is known as *backprojection* or *deskewing*.

3 Fundamental Properties of Skewed Symmetry

In this section, the basic properties of skewed symmetry are presented. Many of these are intuitively obvious and mentioned in a number of articles including [16, 14], but for the sake of completeness, listed below.



S S'

Figure 2: Symmetric shape S and affine-transformed contour S'

Suppose S is a 2D bilaterally symmetric shape, and its affine-transformed contour is denoted by S' (figure 2), then:

- *The axis of skewed symmetry on S' is straight.* This results from the fact that projective and affine transformations preserve collinearity, ie. straight lines map to straight lines.
- *Lines joining symmetrical pairs of points on S' are not necessarily orthogonal to the axis of symmetry.* The lines joining symmetric point pairs on S are orthogonal to the axis of symmetry. However affine/projective transformations do not preserve angles (while scaled euclidean transformations do).
- *Lines joining symmetrical pairs of points on S' are parallel to each other. These are termed lines of skew.* Since lines joining symmetric point pairs on S are orthogonal to the symmetry axis, they are therefore *parallel* to each other.

Affine transformations (but not projective) preserve parallelism. This allows an *angle of skew* to be associated with S', which is defined to be the angle of deviation of the lines of skew from being perpendicular to the symmetry axis.

- *The the mid-points of the lines of skew on S' lie on the symmetry axis.* The symmetry axis bisects the lines joining symmetric point pairs on S. Affine transformations preserve ratio of lengths on a straight line, hence mid-points on S remains as mid-points on S'. Projective symmetries do not have this property.

In addition, a number of properties are presented below which are incorporated into key ideas discussed in later sections.

Property 1 *Tangents of a symmetric pair of points belonging to S' will intersect at the symmetry axis if they are not parallel, in which case they intersect at the symmetry axis at infinity.*

Figure 3 describes property 1 pictorially. No formal proof is given, but if one accepts that tangents from symmetric pairs of points on S intersect at the symmetry axis, a simple intuitive proof may be suggested: *The intersection point is shared by the two tangents and the symmetry axis. Projective transformations preserves the collinearity of the 'double' points (defining the tangent) at the contour with the intersection point at the symmetry axis in S', as well as the collinearity of the points on the symmetry axis. Hence the intersection point will always maintain the 'link' between the the three lines.*

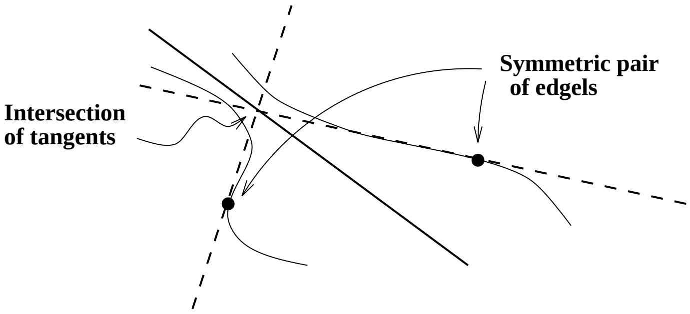


Figure 3: Tangent pairs intersect on the axis of symmetry.

Property 2 *The orientation of the skewed axis of symmetry for S' must lie between the extrapolation directions of the two tangents at every pair of symmetric points on S'.*

This property constraints the direction of the axis of symmetry at every intersection point of tangents belonging to symmetric contour point pairs, and is also invariant under projective transformation as well.

The following proposition states that the angle of skew may be calculated based on the tangent directions at every such intersection point.

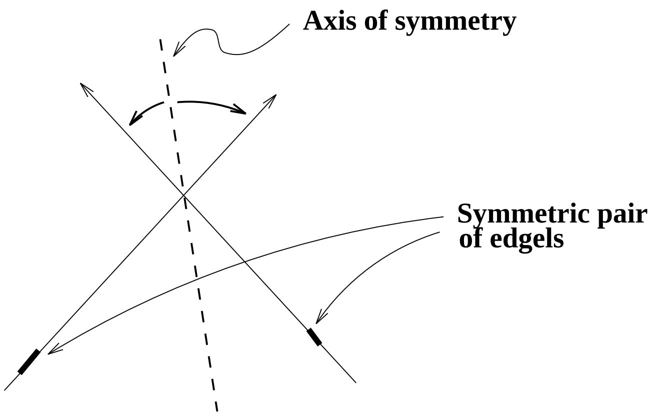


Figure 4: Constraints on the direction of the symmetry axis

Proposition 1 *Given only the directions of tangents belonging to a symmetric point pair in S', and the direction of the skewed symmetry axis, we can compute a unique direction of skew for S', given by*

$$\tan\theta = \frac{1}{2}(\cot\gamma - \cot\beta) \quad (1)$$

where θ, β and γ are as shown in figure 5.

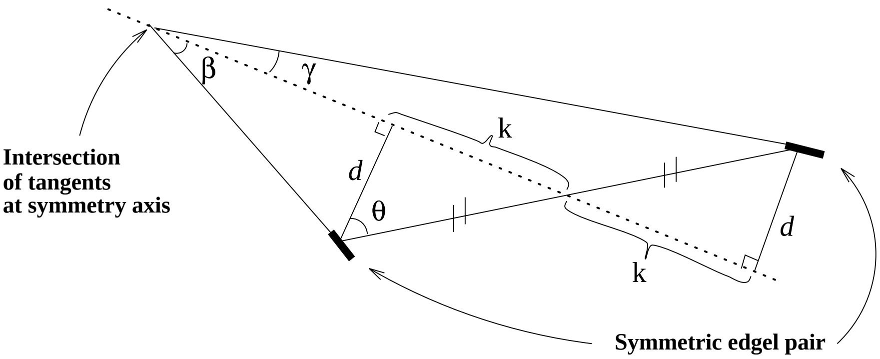


Figure 5: Notation for proposition 1

Referring to figure 5,

$$k = d\ \tan\theta \quad (2)$$

But

$$\frac{d}{\tan\gamma} - \frac{d}{\tan\beta} = 2k \quad (3)$$

Eliminating k we have

$$\tan\theta = \frac{1}{2}\left(\frac{1}{\tan\gamma} - \frac{1}{\tan\beta}\right) = \frac{1}{2}(\cot\gamma - \cot\beta) \quad (4)$$

3.1 Local Symmetry Relation of Curvatures to Tangent Angles

Since there is only one direction of skew per symmetry axis under affine transformation, it follows that the direction of skew computed for all tangent-pair intersections lying on the symmetry axis should theoretically be the same. Under noisy conditions this may be computed as a mean, with an associated variance.

Backprojection (section 2.1) may be used to deskew S'. Although it may appear that the planar orientation of the shape is uniquely determined, this is not true [16] and only a one-parameter family of shapes with orthogonal symmetry axes may be recovered. All that is needed is to determine the affine coordinates of each point using the axes of symmetry and skew as basis vectors, and project them to a *canonical frame* where the basis vectors are *orthogonal.*

3.2 Local Skewed Symmetries

There are two justifications for the use of local measures:

- local measures are less sensitive to occlusions compared to global ones, and
- the correspondence problem is reduced by comparing pairs of contour points and their derivatives rather than comparing larger sets of points.

These two factors are important when attempting to include symmetry as a viable grouping technique in computational perceptual organisation.

For a point on a curve, its description may be expressed as a Taylor series expansion [8]. The zeroth order term gives the location of the point, the first order term the orientation of the tangent, the second order term the curvature, etc. The locations of two skewed symmetric points provide two constraints on two parameters: the angle of skew and the mid-point between them (the axis of symmetry must pass through this mid-point). Orientation information gives another constraint to fully define the axis of symmetry by property 1, which states that the intersection of tangents to a pair of skewed symmetric contour points lie on the axis of symmetry.

Having uniquely defined the symmetry basis consisting of an axis of symmetry and an angle of skew for the pair of points, we can verify the basis through a further constraint based on curvature:

Proposition 2 *Suppose κ_a and κ_b are the curvatures (inverses of the radii of curvature R_a and R_b shown in figure 6) and σ_a and σ_b are the respective angles between the curve normals and the angle of skew, as shown in figure 6. Then if the two points are skewed symmetric, the following must hold:*

$$\frac{\kappa_a}{\kappa_b} = \left(\frac{\cos \sigma_a}{\cos \sigma_b}\right)^3 \tag{5}$$

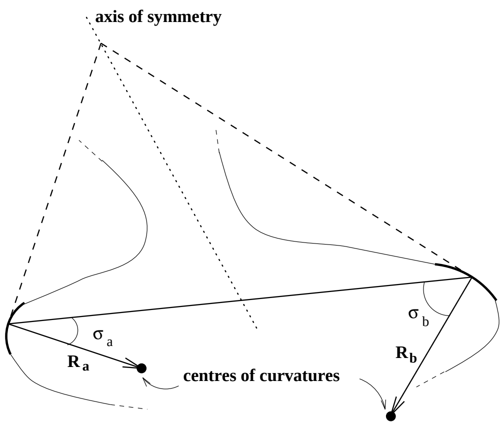


Figure 6: *Proposition 2 relates the curvatures of a pair symmetric contour points to the angles between the normals and the axis of skew, as shown above. This provides the additional constraint mentioned in definition 1 to define local skewed symmetries.*

The result is similar to that given by Ponce [26][1], but the derivation is obtained independently via a different approach and shown in appendix A.1. We can extend this constraint verification process to the highest order derivative to which the curve is locally differentiable. This allows us to **extend the definition of local symmetries to different transformation groups**:

Definition 1 *A local symmetry for a particular transformation group refers to a pair of contour points defining a unique symmetry basis within that group. The required number of parameters is obtained through the derivatives of the contours at those points. Additionally, the symmetry basis must also be consistent with at least one other independent constraint provided by the next higher order derivative.*

Notice that the *smoothed local symmetries* proposed by Brady and Asada [4] satisfy this definition, since a symmetry basis under scaled Euclidean transformations may be uniquely defined by the positions of two points; the requirement that the orientations of the tangents to the contours must be equal with respect to the line joining the points, provides the additional independent constraint. We can also define **local skewed symmetries**, for which the positions and the tangents at the symmetric contour points define a unique symmetry basis under affine transformations, and the curvatures provide the extra constraint to be satisfied in the form of (5).

Another important point to note is that pairs of points on straight line segments are *not* distinguishable because there are no second derivatives — in fact every pair of points on two straight line segments are local skewed symmetries. This is analogous

[1] Ponce acknowledged typographical errors in two of the final three equations given in [26], replacing cosines with sines.

to the local *aperture problem* because of the lack of distinguishable points[2].

3.3 Symmetricity

The reliability of derivatives extracted from image features degrades considerably with increasing order of derivative, and attempting to classify pairs of points on contours as local skewed symmetries is unlikely to be accurate, even if splines are used to fit smoothly varying contours. It is therefore proposed that a continuous measure of local symmetry be adopted based on (5).

It is possible to rewrite (5) as $\kappa_a \cos^3 \sigma_b = \kappa_b \cos^3 \sigma_a$, and select the terms on both sides of the equation as state variables. In figure 7, the state space with state variables $v = \kappa_a \cos^3 \sigma_b$ and $u = \kappa_b \cos^3 \sigma_a$ is shown for a pair of contour points. The state of the contour-point pair will map onto a single spot ($\mathbf{u_o}$) in this state space, and the straight line $v = u$ is the *subspace for which local skewed symmetry holds*. Noise perturbs the positions of contour-point pairs in this state space, and therefore the proposed symmetry measure is based on the shortest distance to the subspace. Since the error variances are not the same nor necessarily independent for both state variables, then on the assumption that the errors are Gaussian[3], *Mahalanobis* distances [30] is used instead of the perpendicular (Euclidean) distances.

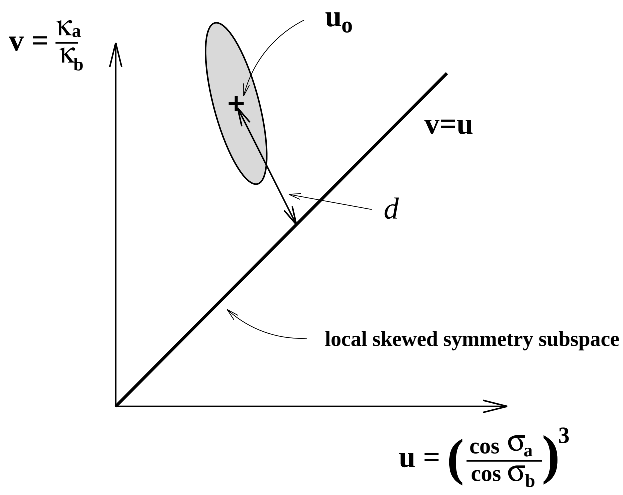


Figure 7: *In the above symmetricity state-space the straight line $v = u$ represents the local skewed symmetry subspace. The symmetricity between a pair of contour points is inversely proportional to (1 + the shortest Mahalanobis distance squared (d^2) from their observed combined state position ($\mathbf{u_o}$) to the subspace).*

Assuming that the covariance matrix $\mathbf{K}$ between u and v may be obtained (eg.

[2] we will however be able to determine if line segments, are symmetrical *if* we know the exact end points, without the need for curvature.

[3] The errors in observation data (orientation, etc.) are assumed to be Gaussian. If these errors are small, then considering only first order terms in the Taylor series expansion of the expressions for the state variables, the errors in the state variables are also approximately Gaussian.

via automated B-spline fitting; see A.4), then the Mahalanobis distance squared d^2 is given by

$$d^2 = (\mathbf{u} - \mathbf{u_o})' \mathbf{K}^{-1} (\mathbf{u} - \mathbf{u_0}) \tag{6}$$

where $\mathbf{u} = [\, u\, v\,]'$ and $\mathbf{u_o}$ is the observed state of the contour-point pair. The shortest Mahalanobis distance to the local skewed symmetry subspace may be found by substituting $v = u$ into (6) and minimising the quadratic equation. These steps are further elaborated in appendix A.2.

This leads us to the following redefinition for *symmetricity*.

Definition 2 *Given the shortest* Mahalanobis *distance d from the observed state of the pair of contour points a and b, to the local skewed symmetry subspace within the state-space spanned by u and v as described above, the* **symmetricity** *between a and b is given by*

$$\Psi_{a,b} = \frac{1}{1 + kd^2} \tag{7}$$

where k is a empirical constant determining the scale of the Mahalanobis distance.

3.4 The Local Skewed Symmetry Field

In order to *spatially* represent the symmetry evaluation for each pair of contour points in an intuitive and useful way, we define a **local skewed symmetry field** for the set of contours in an image:

Definition 3 *The local skewed symmetry field (LSSF) for a set of contours is formed by:*

- *assigning the symmetricity for a pair of points on the contours to the location of the mid-point of the line joining the pair of points. If more than one pair of points maps to this location, only the pair with highest symmetricity is retained. This is the magnitude of the field at the location.*
- *assigning a direction vector representing the direction of the local axis of symmetry ; This is termed the 'first direction' of the LSSF.*
- *assigning another direction vector parallel to the line joining the pair of points, representing the direction of skew; the 'second direction'.*

Figure 8 shows two planar symmetric objects to which cubic B-splines are manually fitted to the edge maps (obtained from the Canny edgefinder [6]) in order to obtain the derivatives for computation of the local skewed symmetry fields.

The magnitude and first direction components of the local skewed symmetry fields are shown in figure 9.

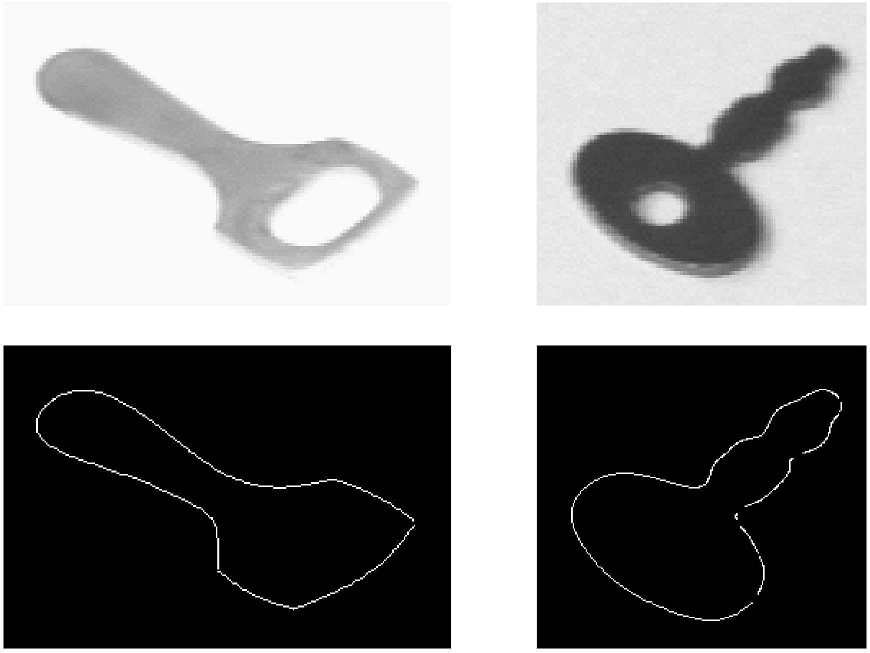


Figure 8: *In order to obtain curvature information to calculate the local skewed symmetry field, cubic B-splines are fitted to the output from the Canny edge-detector.*

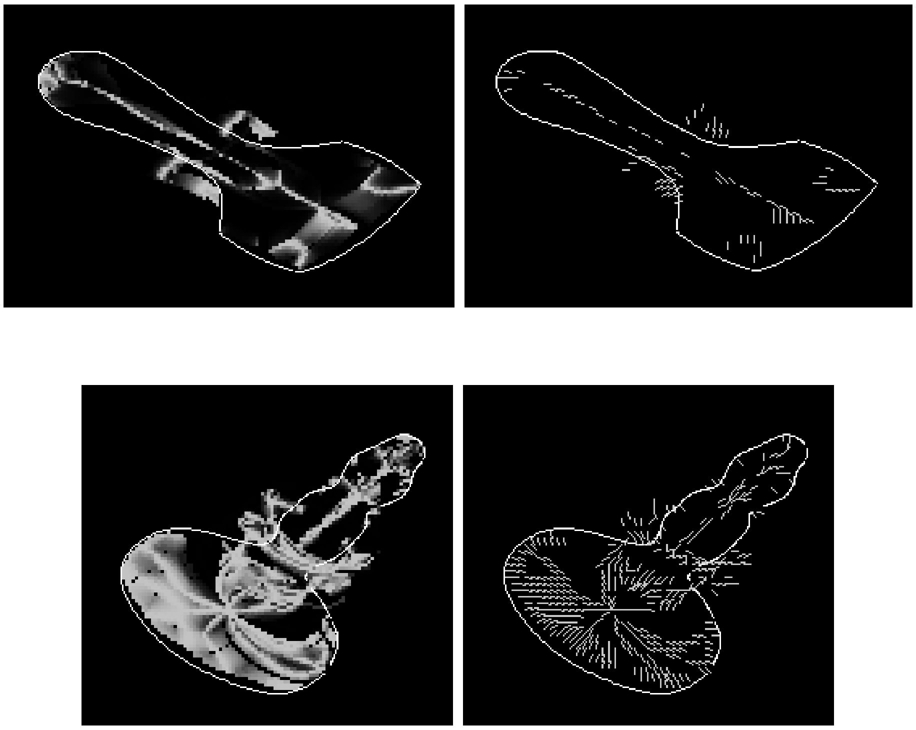


Figure 9: *Two components of the local skewed symmetry fields for the object contours in figure 8 are represented separately: the symmetricity values are shown on the left image, the first directions (local axes of symmetry) are mapped on the right. These local skewed symmetry fields should not be judged solely on the appearance of the either map: when combined together they specify the effectiveness of each local symmetry when considering a particular orientation, eg. the direction of the hypothesised axis of symmetry.*

Of importance is the fact that the peaks and ridges in the fields are the suggested local skewed symmetries. In the total absence of errors, these ridges will be analogous to the symmetry set for smoothed local symmetries[4] [4]. The symmetricity values represent estimations of the likelihood for pairs of points to be local symmetries, and provide some robustness when noise is introduced.

These properties allow it to degrade more gracefully than if we were to use global measures or strict local measures. This makes it useful as a tool for recovering the global axis of symmetry from imperfect local skewed symmetries.

Recent research [28] has also independently proposed a similar field for local *bilateral* symmetries, which is applied to locating useful points for fixation. Also, the magnitudes for the local symmetries were calculated based on sinusoids instead of derived error measures as in the case of symmetricity mentioned above.

4 Implementation

4.1 Symmetry Detection Strategy

In images of objects with complicated contours, it is hard to automatically infer the axis of symmetry purely from its LSSF, much less do so in the presence of noise and occlusion. A vast amount of computation will be required to analyse and compare different portions of the LSSF. Our strategy is therefore to:

1. Obtain initial hypotheses for the axis of symmetry from a fast but approximate symmetry detection process. The symmetry detection process has to use local measures as well to avoid problems of occlusion and incomplete contours.
2. Compute the local skewed symmetry field with the aid of B-splines (to obtain curvature information from the *edge map* of the image).
3. Improve the accuracy of the axis of symmetry and angle of skew by maximising a global symmetry measure from local support in the LSSF.

Steps 1 and 2 can in principle be carried out in parallel.

4.2 Initial Symmetry Detection (Hough Transform) Technique

A novel, fast method is introduced for the initial symmetry detection. In the same spirit of the paper it allows local structures to vote independently for the global axis

[4] The ridges in the field bear some similarity to the symmetry set for smoothed local symmetries. However the ridges of the LSSF are *not* affine-transformed smoothed local symmetries because smoothed local symmetries are not classified by curvatures.

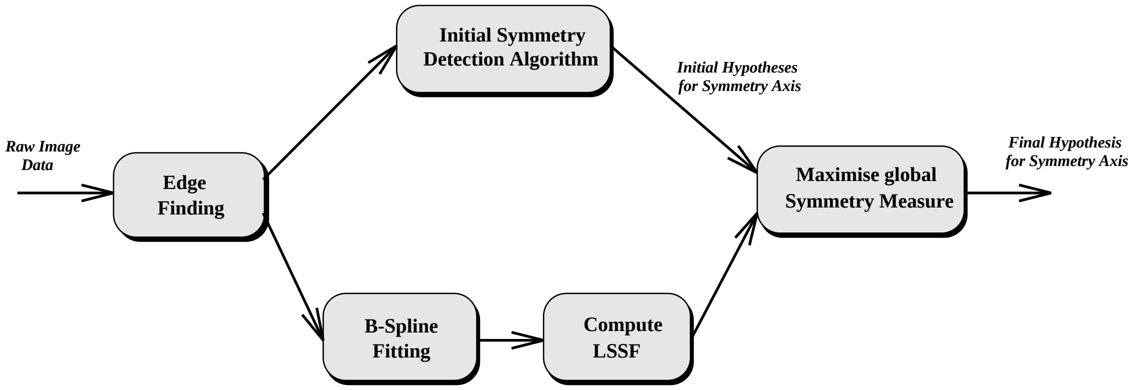


Figure 10: *Flow diagram for symmetry detection strategy*

of symmetry and angle of skew, but in this case, the local structures are limited to *low-curvature segments* and *straight edges*.

Using the edge map of the image as input, the symmetry detection process comprises of 4 steps as shown in figure 11:

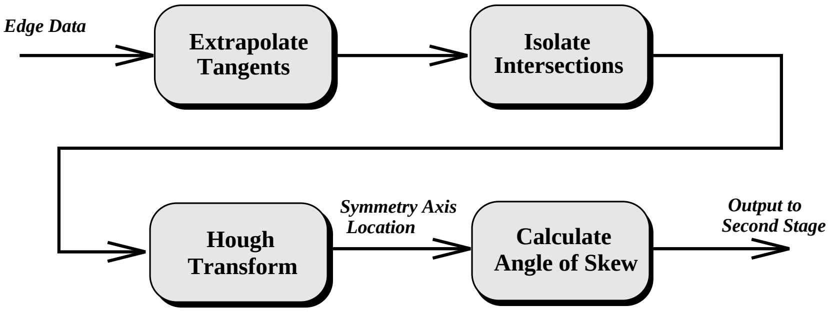


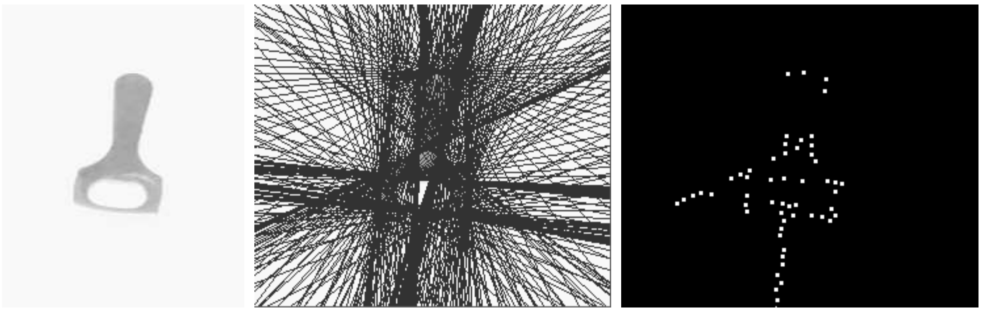
Figure 11: *Flow diagram for the initial symmetry detection algorithm*

1. **Extrapolate tangents of edgels in an array.** This is shown in figure 12. Each entry in the array contains data on the directions and numbers of tangents passing through it.

2. **Recover locations of strong intersections.** Entries with more than one registered direction are classified as intersection points. Each intersection point is given a strength corresponding to the number of votes for the *second strongest* direction, and only points above a specified intersection threshold are retained. By recovering only strong intersections, we are effectively considering only the tangents belonging to low-curvature segments. The locations of these intersections are more accurate than those obtained from high-curvature segments.

3. **Carry out a reduced Hough Transform on the intersection points.**

   The variation of the Hough Transform used here parameterise lines by the normal vector from the origin to the line, expressed in polar coordinates (d, θ), as



(a) (b) (c)

Figure 12: *Based on the edge map of the image in (a), the tangents for each edgel are extrapolated in (b). In (c), strong intersection points are recovered which represent intersections of low-curvature segments or straight lines.*

shown in figure 13.

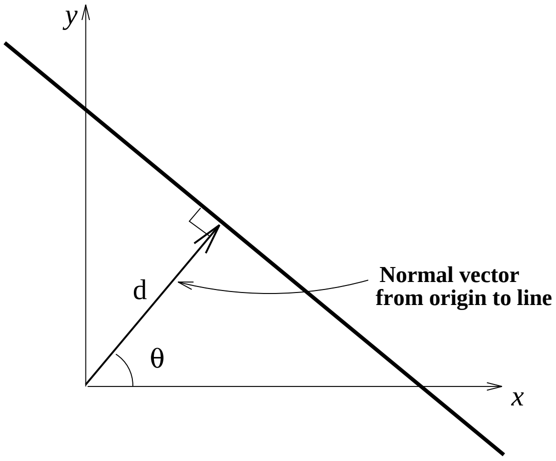
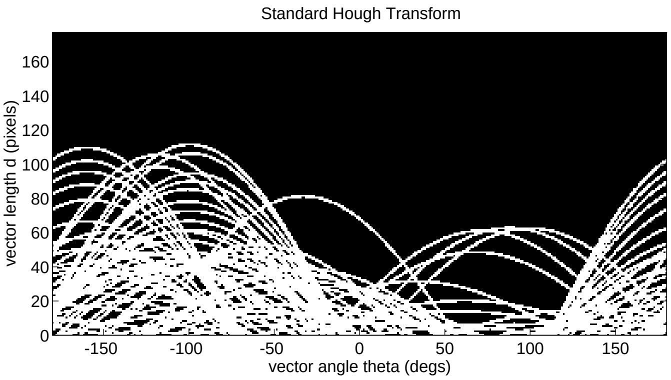


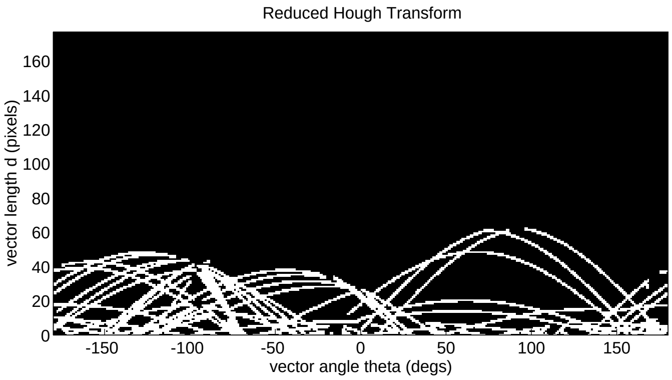
Figure 13: *A line in image space is parameterise by the normal vector from the origin to the line, expressed in polar coordinates.*

Each intersection point maps to a half-sinusoidal curve in Hough space, as may be seen in figure 14. The additional knowledge of property 2 allows us to reduce the Hough Transform by coding only the relevant portions of the sinusoids representing the angular sweep between the directions of the extrapolated tangents, since the property states that *the orientation of the axis of symmetry must lie between the extrapolation directions of tangents belonging to a skewed symmetric edgel pair*. This reduces the computation time as well as ignore false hypotheses such as those joining the tips of a serrated edge. The locations of multiple intersections in the Hough Space represent hypotheses for symmetry axis, which is based on property 1. Since there is no guarantee that the axis of symmetry has the largest number of intersections on it, we have to look at a number of strong hypotheses rather than just one.

4. **Estimate angle of skew and rank the hypotheses.** Ranking the hypotheses



(a)



(b)

Figure 14: *The Hough space in (a) shows the result of carrying out a standard Hough Transform on the intersection points shown in figure 12. Knowledge of the directions of tangents at the intersections allows reduction of the complexity of the Hough space (b), from property 2.*

is important since the more likely axes of symmetry should be passed onto the next stage first. This is achieved in a dual-purpose manner by predicting the angle of skew per hypothesis for the axis of symmetry. It is possible to calculate the angle of skew from just the directions of tangents at *one* intersection point and knowledge of the axis of symmetry, based on (1). Since we will have a number of intersection points contributing votes θ_k for the angle of skew with *different* degrees of accuracy, we will sum θ_k modified by an *optimal set of weights* a_k (obtained through sensitivity analysis and Lagrange multipliers) as expressed:

$$\hat{\theta} = \sum_{k=1}^{N} a_k \theta_k \tag{8}$$

where $\hat{\theta}$ is the optimal estimate. A detailed derivation is shown in appendix A.3. The *sample* variance may be obtained as well

$$\hat{\sigma}^2 = \frac{1}{N-1} \sum_{k=1}^{N} a_k^2 (\theta_k - \hat{\theta})^2 \tag{9}$$

This sample variance will be high for collinear intersection points which do not have similar predictions for the angle of skew, and hence can be used to reject false hypotheses. Therefore *the hypotheses are ranked in ascending order of their sample variances.*

This algorithm is a symmetry detection technique by itself since it recovers both the axis of symmetry and the angle of skew, and carries this out by making use of *local* data. The method is fast ($O(kN)$ where k is a linear dimension of the image and N is the number of edgels) but suffers from the disadvantage that near-parallel segments (which have intersection points outside the array) and high-curvature segments (which do not generate strong intersection points) do not vote.

5 B-Spline Curve Fitting

B-spline curve fitting is presently used to recover smoothed local curvature information, although any robust curvature estimation algorithms may be substituted.

Present techniques used in automatic fitting of B-splines by Medioni and Yasumoto [23] suffer from unstable estimated derivatives, since the method assumes that the control points of the spline are evenly spread out. In our experiments a partially automated technique is used. Chains of edgels are first automatically selected, followed by the manual specification of the control point locations *along the spline parameters*, and finally least squares solutions are obtained for the spline fits.

The least squares approach to spline-fitting is described in detail in a number of texts including [15], and assumes the model

$$\mathbf{X} = \mathbf{BP} + \mathbf{E} \tag{10}$$

where $\mathbf{X}$ is the matrix of observed boundary points, $\mathbf{B}$ is the standard spline blending matrix, $\mathbf{P}$ is the matrix of control points, and $\mathbf{E}$ is the error matrix with independent elements. Two constraints are also included to force the ends of each spline to coincide with the ends of the associated list of sampled edge points. The only difference between splines with an even spread of control points and those with individually-specified control point spacings is in the formulation of $\mathbf{B}$ — otherwise the spline models are the same.

The least squares estimation of the control points is found using Lagrange multipliers by solving

$$\begin{bmatrix} 2\mathbf{B}'\mathbf{B} & -\mathbf{b_1} & -\mathbf{b_n} \\ \mathbf{b}'_\mathbf{1} & 0 & 0 \\ \mathbf{b}'_\mathbf{n} & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{P} \\ \lambda' \\ \mu' \end{bmatrix} = \begin{bmatrix} 2\mathbf{B}'\mathbf{X} \\ \mathbf{x}'_\mathbf{1} \\ \mathbf{x}'_\mathbf{n} \end{bmatrix} \tag{11}$$

where $\mathbf{b}'_\mathbf{1}$,$\mathbf{b}'_\mathbf{n}$ are respectively the top and bottom rows of $\mathbf{B}$; $\mathbf{x}'_\mathbf{1}$,$\mathbf{x}'_\mathbf{n}$ are coordinates of the ends of the list of edge points (top and bottom rows of $\mathbf{X}$; and λ',μ' are vector forms of Lagrange multipliers.

Furthermore, the unbiased estimates for the overall x and y error variances, $\hat{\sigma}_x$ and $\hat{\sigma}_y$, for the boundary points may be obtained, which in turn provides estimates of the x and y covariance matrices for the control point positions. The full derivation may be found in a number of texts including [9]. We are also able calculate the variances and covariances between a point on the spline (as an estimate for the *actual* boundary point) and its derivatives with respect to the spline parameter t. The covariance matrix $\mathbf{K}$, mentioned in section 3.3, may be obtained after further manipulation. Details may be found in appendices A.4 and A.5.

This technique has the advantage that the spline fits are smoother and more accurate than the fully automated method, and it is also possible to obtain the covariances of estimated derivatives.

5.1 Calculating the Local Skewed Symmetry Field

A number of sample points per polynomial piece are then extracted with tangent and curvature information. Each sample point is compared with the rest in $\frac{1}{2}N(N-1)$ operations (where N is the number of sample points), to derive the LSSF in three arrays of the same size as the original image, containing separately the symmetricity values, and the two direction data. Although computationally expensive, this method has the advantage of not involving any occlusion-sensitive parameters.

5.2 Maximisation of Symmetry Measure in the Local Skewed Symmetry Field

Having obtained a ranked list of hypotheses for the axis of symmetry, we are able to calculate an overall symmetry measure for each of the hypothesis by averaging the **effective symmetricity** values of the points in the LSSF crossed by the axis of symmetry. Taking S as the global symmetry measure and P as the number of sample points, we have

$$S = \frac{1}{P} \sum_{i=1}^{P} \grave{\Psi}_{i\theta} \tag{12}$$

$$\grave{\Psi}_{i\theta} = \Psi_i |\cos(\theta - \alpha_i)| \tag{13}$$

where $\grave{\Psi}_{i\theta}$ is the *effective* symmetricity value at sample point i with respect to the *global* axis direction θ, Ψ_i is the *absolute* symmetricity value measured at sample point i and α_i is the direction of the *local* axis of symmetry in the LSSF at sample point i.

We then proceed to locate the position of maximum symmetry measure for the symmetry axis by initialising a *straight active contour model* [17] (a 'pole'). This 'pole', unlike normal 'snakes', is driven by effective symmetricity values which change according to the orientation of the pole. See figure 15.

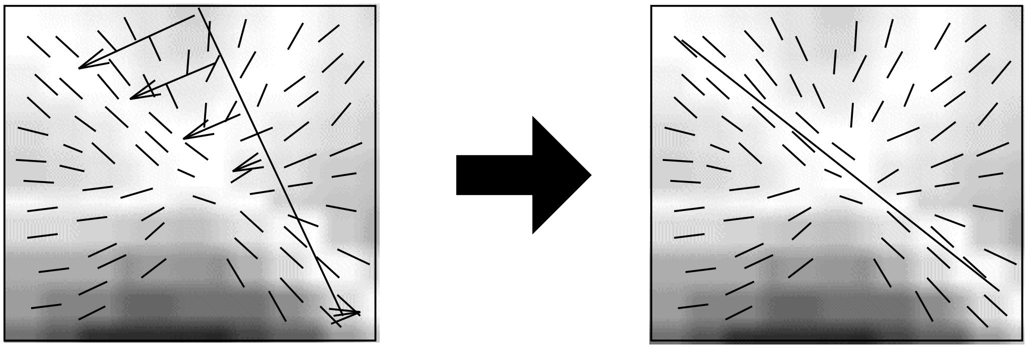


Figure 15: *The above figure diagrammatically shows a straight active contour model (a 'pole') being used to locate the position in the LSSF where the global symmetry measure is maximised. The 'forces' on the pole are based on effective symmetricity (defined in (13)) which depend on* **both** *the absolute symmetricity as well as the angle between the (first) direction of the field and the orientation of the pole. The pole is initialized at the location of the hypotheses obtained from the symmetry detection algorithm forming the first stage.*

Once the location of maximum symmetry measure has been found, the angle of skew may be estimated based on the weighted average of the second directions in the LSSF (local angles of skew) scaled by their effective symmetricity values. The new hypotheses for the axes of symmetry may then be ranked again by their global symmetry measure S. However this does not mean we reject all but one of the hypotheses, since different symmetrical objects, or separate symmetrical components, may be present.

6 Analysis of Results

6.1 Preliminary Results

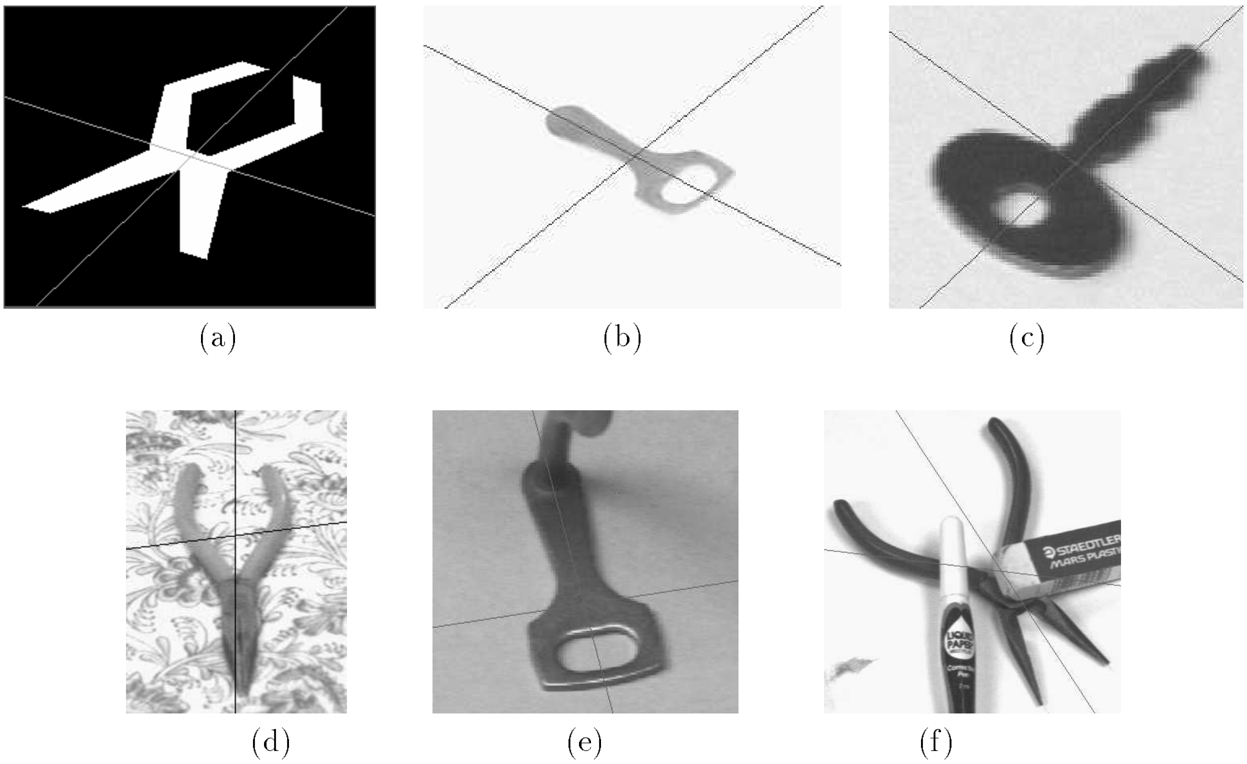


Figure 16: *Results obtained from the first symmetry detection stage indicate that the first stage is performing reasonably. However the accuracy is not robust enough to noise and occlusion, such as in the case of (d) and (f). Hence the need to use the LSSF.*

The initial symmetry detection algorithm was tested on a number of synthetic and real images, with straight and curved contours. The results show good performance by the initial technique, in most cases the location of the axis is found to a satisfactory degree, and often with an angle of skew of adequate accuracy. The images in figure 16 show the results obtained just after the first stage of the algorithm.

Figure 17 shows back-projection of the relatively more intact edge maps belonging to the images in the top row of figure 16, to a canonical frame with orthogonal symmetry axes.

However, there is a possibility of having extraneous intersection points from other structures located on the predicted axis of symmetry, and their votes are likely to badly distort the estimation for the angle of skew. This occurs in some of the lower

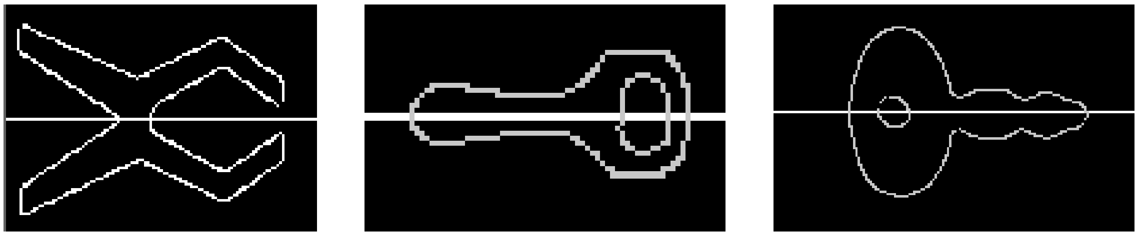


Figure 17: *Backprojection of the relatively intact edge maps of (a), (b) and (c) in figure 16 to a canonical frame where the symmetry axes are orthogonal.*

ranked hypotheses where slight displacement of the axis of symmetry resulted in large errors for the angle of skew. Ranking the hypotheses is also not always accurate, and therefore there is a need to use the LSSF to improve the robustness of such estimations. Some of these poorer hypotheses are used in the final stage of the algorithm to demonstrate the ability of the algorithm to handle such errors.

Besides the easily segmented objects shown in figure 8, we have also calculated the LSSF for two other scenes of objects for which mild occlusion is present, which results in fragmented contours and unwanted artifacts. These are shown in figure 18.

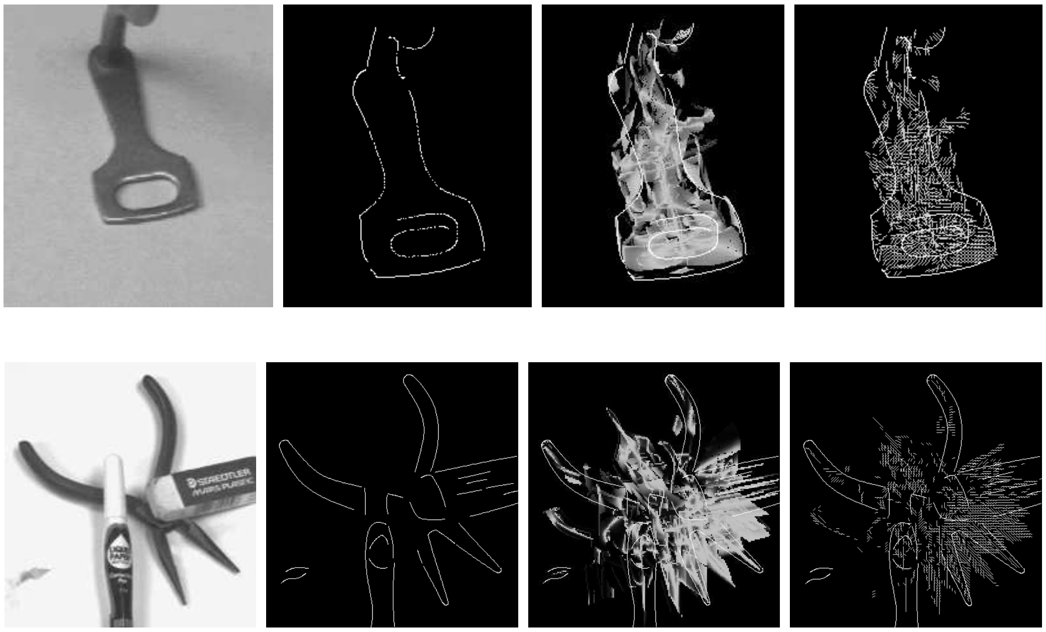


Figure 18: *The above figure shows the local skewed symmetry fields derived from the B-spline fitted edge maps of two objects, with the magnitude and direction maps shown respectively third and fourth from the left. These will be used in the last stage of the symmetry detection algorithm.*

Since accurate results are obtained in general from the initial algorithm, lower ranked hypotheses have been used to demonstrate the use of the 'pole' in the maximisation of the global symmetry measure in the LSSF, as mentioned earlier. The errors in both the positions of the axes of symmetry and the angles of skew are minimized accurately. These examples are shown in figure 19.

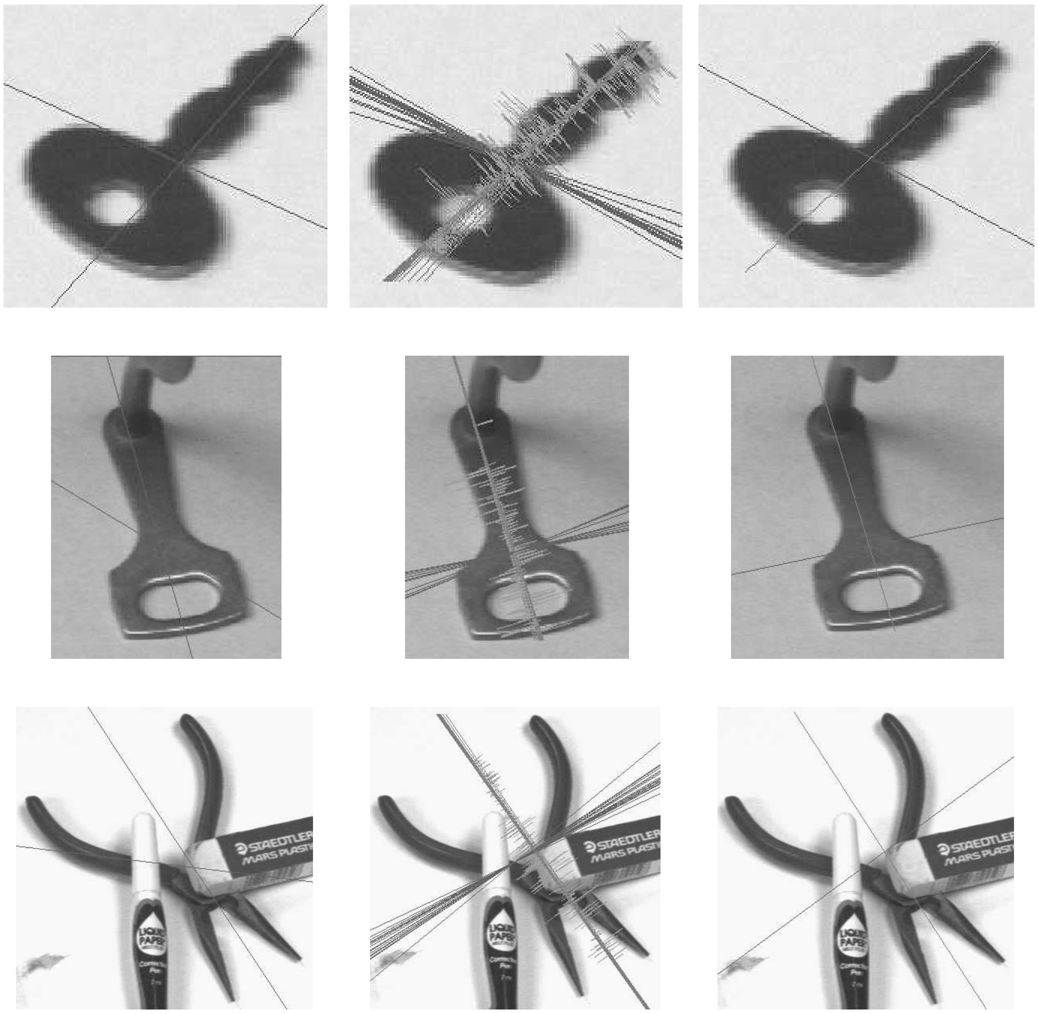


Figure 19: *The left column shows the initial hypotheses obtained from the first stage of the algorithm. Lower ranked hypotheses are used for images (a) and (b) in order to demonstrate this stage of the algorithm. The middle column shows the active 'poles' searching for the positions returning maximum global symmetry measures. The final results are shown in the right column. Notice that in both (b) and (c) where the objects are difficult to segment (because of shadows and occlusion) the algorithm still returned good estimates for the axis of symmetry and the angle of skew.*

Overall, the results obtained for the thinner and better segmented objects are particularly good. For objects which have significant thicknesses and those in scenes which include shadows, occlusion or noise, the results still return useful estimates.

6.2 Discussion

Both the initial symmetry detection algorithm and the later stages share similar advantages. The algorithm uses low-level data directly to extract symmetry. It is local in nature and does not depend on accurate grouping of edgels into complete contours, nor does it require segmentation of the object (as is necessary for a moment-based approach). Instead it allows individual pairs of points to vote for the symmetry axis and angle of skew, and hence it is resistant to occlusion and fragmentation. Noise and quantisation errors are handled by uncertainty analysis in the first stage and by adopting a continuous measure of local symmetry (symmetricity). The sample variance obtained for the angle of skew from the first stage and the global symmetry measure obtained in the final stage both provide means of measuring the overall symmetry. This mean that it is possible to gauge how symmetric the object is under the estimated set of axes and hence to be able to reject false hypotheses. The initial symmetry detection algorithm also has a speed advantage without having to adopt global measures, unlike other algorithms which are normally slow if they use local measures, or suffer from having to test a discretized set of orientations.

Some of the problems faced with our implementation are described below. The initial symmetry detection algorithm is unable to handle symmetric objects comprising solely of parallel symmetric edges, eg. a rectangle (in fact it interprets a rectangle as a skewed rhombus). The performance of this stage is also proportional to the number of unique intersection points of symmetric edgel pairs. Hence the algorithm favours gently curved objects or those with many non-parallel symmetric edges.

Two difficulties are encountered in the second stage of the algorithm. The first problem is speed, since every pair of sample points must be compared, which makes it slow. The other problem is that the curvature measurements are sensitive to errors, and the B-spline fits must be smooth in orientation to have stable curvature values. In some cases it may be better to ignore the symmetricity value and maximise the global symmetry measure using only the *first* directions in the LSSF. Finally, if a scene is over-cluttered with details, the large number of overlapping entries in the LSSF will reduce the coherency of the entries which contribute to the overall global symmetry.

7 Conclusions and Future Work

Although many methods have been presented for symmetry detection, they do not handle occlusion, fragmentation or noise well, which makes them unsuitable candidates to detect the symmetries needed in conditions where perceptual organisation is most useful. We have presented a new method which does not depend on segmentation of objects or any accurate determination of occlusion-sensitive global measures (eg. moments). Instead, independent support from local structures may be combined to achieve global symmetry detection in the presence of noise and even occlusion, via the local skewed symmetry field. Preliminary results obtained have been promising.

The next stage of development will be to allow *coherent* neighbouring elements in the LSSF to *reinforce* each other in order to suppress non-generic local skewed symmetries, ie. those which occur only instantaneously at two particular points on two contours, thereby relieving the problem faced when there are too many overlapping entries.

Given a method for detecting global skewed symmetry axes from an image, we will proceed on to investigate the recovery of the image contours from which the symmetry axes were formed, and to *improve the accuracy of the estimated contours based on symmetrical redundancy.* The eventual goal will be to accurately detect and segment symmetrical objects in an image, in *reverse* to the conventional method of segmenting the objects first followed by recovery of the symmetry axes. It is hope that this will be of particular use to computational perceptual organisation since it has been shown that symmetry is both a local and a global property, and is therefore useful in a grouping process.

Acknowledgements

The authors would like to thank Prof. Steve Zucker, Prof. Horace Barlow, Andrew Zisserman and Mark Wright for discussions and ideas. This research is supported by an Overseas Research Scholarship and a Cambridge Commonwealth Trust Bursary.

References

[1] M. J. Atallah. On symmetry detection. *IEEE Trans. on Computers*, 34(7):663–666, 1985.

[2] H. B. Barlow and B. C. Reeves. The versatility and absolute efficiency of detecting mirror symmetry in random dot displays. *Vision Research*, 19:783–793, 1979.

[3] H. Blum. A transformation for extracting new descriptors of shape. In W. Wathen-Dunn, editor, *Models for the Perception of Speech and Visual Form*, pages 362–380. MIT Press, 1967.

[4] M. Brady and H. Asada. Smoothed local symmetries and their implementation. *Int. Journal of Robotics Research*, 3(3):36 –61, 1984.

[5] M. Brady and A. Yuille. An extremum principle for shape from contour. *IEEE Trans. Pattern Analysis and Machine Intell.*, 6:288–301, 1984.

[6] J.F. Canny. A computational approach to edge detection. *IEEE Trans. Pattern Analysis and Machine Intell.*, 8:679–698, 1986.

[7] T. J. Cham. Visual Symmetry: Towards a Role in Perceptual Organisation. First Year PhD Research Report, June 1994. Department of Engineering, University of Cambridge.

[8] M.P. DoCarmo. *Differential Geometry of Curves and Surfaces*. Prentice-Hall, 1976.

[9] N. Draper and H. Smith. *Applied Regression Analysis*. Wiley Series in Probability and Mathematical Statistics. John Wiley, 1966.

[10] P. Eades and H. C. Ng. An algorithm for detecting symmetries in drawings. *Ars Combinatoria*, 23A:95–104, 1987.

[11] S.A. Friedberg. Symmetry evaluators. Technical Report TR 134, University of Rochester, 1984. (Revised 1986).

[12] S.A. Friedberg. Finding axes of skewed symmetry. *Computer Vision, Graphics and Image Processing*, 34:138–155, 1986.

[13] R. Glachet, J.T. Lapreste, and M. Dhome. Locating and modelling a flat symmetrical object from a single perspective image. *Computer Vision, Graphics and Image Processing – Image Understanding*, 57(2):219–226, 1993.

[14] A. Gross and T. Boult. Analyzing skewed symmetries. Technical Report CUCS-064-90, Columbia University, Dec 1990.

[15] A. K. Jain. *Fundamentals of Digital Image Processing*. Prentice Hall Information and System Science Series. Prentice Hall, 1989.

[16] T. Kanade. Recovery of the three-dimensional shape of an object from a single view. *Artificial Intelligence*, 17:409–460, 1981.

[17] M. Kass, A. Witkin, and D. Terzopoulos. Snakes:active contour models. In *Proc. 1st Int. Conf. on Computer Vision*, pages 259–268, 1987.

[18] F. Labonté, Y. Shapira, and P. Cohen. A perceptually plausible model for global symmetry detection. In *Proc. 4th Int. Conf. on Computer Vision*, pages 258–263, 1993.

[19] D. G. Lowe. *Perceptual Organization and Visual Recognition*. Kluwer Academic Publishers, 1985.

[20] G. Marola. On the detection of the axes of symmetry of symmetric and almost symmetric planar images. *IEEE Trans. Pattern Analysis and Machine Intell.*, 11(1):104–108, 1989.

[21] T. Masuda, K. Yamamoto, and H. Yamada. Detection of partial symmetry using correlation with rotated-reflected images. *Pattern Recognition*, 26(8):1245–1253, 1993.

[22] J. D. McCafferty. *Human and Machine Vision: Computing Perceptual Organisation.* Ellis Horwood Series in Digital and Signal Processing. Ellis Horwood, 1990.

[23] G. Medioni and Y. Yasumoto. Corner detection and curve representation using curve B-splines. In *Proc. CVPR, Miami*, pages 764–769, 1986.

[24] P. Miller and S. Astley. Automated detection of breast asymmetries. In *Proc. British Machine Vision Conference 1993, Guildford*, pages 519–528, 1993.

[25] D.P. Mukherjee, A. Zisserman, and M. Brady. Shape from symmetry – detecting and exploiting symmetry. To appear in Philosophical Trans. of the Royal Society. (Tech. Rep. OUEL 1988/93).

[26] J Ponce. On characterizing ribbons and finding skewed symmetries. *Computer Vision, Graphics and Image Processing*, 52:328–340, 1990.

[27] D. Reisfeld, H. Wolfson, and Y. Yeshurun. Detection of interest points using symmetry. In *Proc. 3rd Int. Conf. on Computer Vision, Osaka*, pages 62–65, 1990.

[28] D. Reisfeld, H. Wolfson, and Y. Yeshurun. Context free attentional operators: the generalised symmetry transform. To appear in *Int. Journal of Computer Vision*, 1994.

[29] L.G. Roberts. Machine perception of three - dimensional solids. In J.T. Tippet, editor, *Optical and Electro-optical Information Processing*. MIT Press, 1965.

[30] C. W. Therrien. *Decision, Estimation and Classification: An Introduction to Pattern Recognition and Related Topics.* John Wiley, 1989.

[31] L. Van Gool, T. Moons, D. Ungureanu, and A. Oosterlinck. The characterisation and detection of skewed symmetry. To appear in *Computer Vision, Graphics and Image Processing*, 1994.

[32] L. Van Gool, T. Moons, D. Ungureanu, and E. Pauwels. Symmetry from shape and shape-from-symmetry. Technical report, Katholieke Universiteit Leuven, Sep 1994. To appear in *Int. Journal of Robotics Research.*

[33] S. K. Yuen. Shape from contour using symmetries. In O. Faugeras, editor, *Lecture Notes in Computer Science*, volume 427, pages 437–453. Springer-Verlag, 1990. Proc. 1st Euro. Conf. on Computer Vision.

Appendices

A.1 Local Symmetry Relation of Curvatures to Tangent Angles

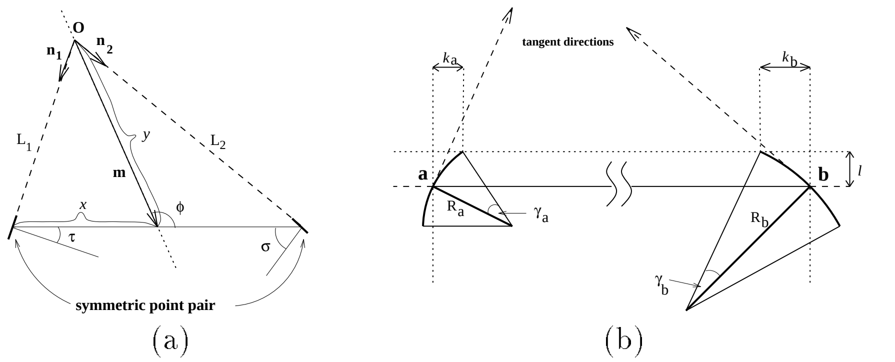


Figure 20: *Notation for appendix A.1 (a) symmetry base system; (b) close-up of symmetric point pair.*

Referring to figure 20, we consider *small* angles γ_a and γ_b. From simple trigonometry,

$$\begin{aligned} k_a &= R_a \gamma_a \sin(\tau + \gamma_a) \\ k_b &= R_b \gamma_b \sin(\sigma + \gamma_b) \end{aligned}$$

$$l = R_a \gamma_a \cos(\tau + \gamma_a) = R_b \gamma_b \cos(\sigma + \gamma_b) \tag{14}$$

Now let

$$\begin{aligned} m &= k_a - l \tan \tau \\ n &= k_b - l \tan \sigma \end{aligned}$$

From the property of skewed symmetry, we have

$$\begin{aligned} m = n \quad &\Rightarrow \quad k_b - k_a = l(\tan \sigma - \tan \tau) \\ &\Rightarrow \quad \tan \sigma - \tan \tau = \tan(\sigma + \gamma_b) - \tan(\tau + \gamma_a) \\ &\Rightarrow \quad \tan \sigma - \tan(\sigma + \gamma_b) = \tan \tau - \tan(\tau + \gamma_a) \end{aligned}$$

Using the simple trigonometric formula $\tan(A - B) = \frac{\tan A - \tan B}{1 + \tan A \tan B}$, we have

$$[1 + \tan \sigma \tan(\sigma + \gamma_b)] \tan \gamma_b = [1 + \tan \tau \tan(\tau + \gamma_a)] \tan \gamma_a$$

This approximates to

$$
\begin{aligned}
(1+\tan^2\sigma)\gamma_b &= (1+\tan^2\tau)\gamma_a \\
\sec^2\sigma\gamma_b &= \sec^2\tau\gamma_a
\end{aligned}
$$

Substituting into (14), we get

$$\frac{R_b}{R_a} = \frac{\sec^2\sigma}{\sec^2\tau}\cdot\frac{\cos\tau}{\cos\sigma} = \frac{\cos^3\tau}{\cos^3\sigma} \tag{15}$$

A.2 Minimization of Mahalanobis Distance

The multivariate Gaussian probability density function for the state-space shown in figure 7 is given by

$$p(\hat{\mathbf{x}}) = \frac{1}{(2\pi)|\mathbf{K}|^{\frac{1}{2}}} e^{[-\frac{1}{2}(\hat{\mathbf{x}}-\mathbf{m})'\mathbf{K}^{-1}(\hat{\mathbf{x}}-\mathbf{m})]} \tag{16}$$

where $\mathbf{m}$ is the true state vector of a pair of contour-points. The Mahalanobis distance is given by the power term of the exponential

$$d^2 = (\hat{\mathbf{x}}-\mathbf{m})'\,\mathbf{K}^{-1}\,(\hat{\mathbf{x}}-\mathbf{m}) \tag{17}$$

Since this is a proper distance function it is possible to use the function to measure the distance between any two points in the state-space, and in this application we consider the minimum Mahalanobis distance from the observed state to the local skewed symmetry subspace $v = u$.

For

$$\mathbf{K}^{-1} = \begin{pmatrix} a & b \\ b & c \end{pmatrix}, \qquad x_o = \left(\frac{\cos\sigma_a}{\cos\sigma_b}\right)^3, \quad y_o = \frac{\kappa_a}{\kappa_b}$$

$$d^2 = a(x_o - x)^2 + 2b(x_o - x)(y_o - y) + c(y_o - y)^2$$

substituting $v = u$ and equating $\frac{d}{dx}(d^2) = 0$,

$$
\begin{aligned}
x_{min} &= \frac{(b+a)x_o + (b+c)y_o}{a+2b+c} && (18) \\
\min_x d^2 &= \left[a - \frac{(a+b)^2}{a+2b+c}\right]x_o^2 + \left[c - \frac{(b+c)^2}{a+2b+c}\right]y_o^2 + \\
&\quad \left[b - \frac{(a+b)(b+c)}{a+2b+c}\right]x_o\,y_o && (19)
\end{aligned}
$$

A.3 Sensitivity Analysis in Angle of Skew Calculations

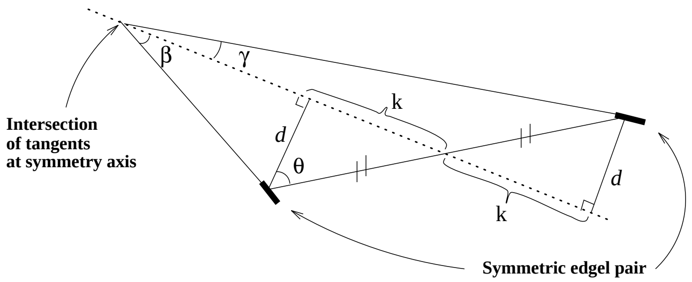


Figure 21: Notation for Sensitivity Analysis

Referring to figure 21, we have from (1),

$$\tan\theta = \frac{1}{2}(\cot\gamma - \cot\beta)$$

or as a function,

$$\theta = g(\beta, \gamma) \tag{20}$$

Expanding and differentiating,

$$\frac{\partial g}{\partial \beta} = \frac{1}{1 + \frac{1}{4}(\cot\gamma - \cot\beta)^2}\,\frac{1}{2}\mathrm{cosec}^2\beta$$

This applies to γ as well, and simplifying gives

$$\frac{\partial \theta}{\partial \beta} = \frac{2}{\sin^2\beta[4 + (\cot\gamma - \cot\beta)^2]} \tag{21}$$

$$\frac{\partial \theta}{\partial \gamma} = -\frac{2}{\sin^2\gamma[4 + (\cot\gamma - \cot\beta)^2]} \tag{22}$$

From N intersection points lying on the axis of symmetry, we derive observations of the angle of skew θ_k, $k = 1, \ldots N$ using (1). These estimates have an error

$$\theta_k = \theta + \delta\theta_k \tag{23}$$

where θ is the exact value of the angle of skew, $\delta\theta_k$ is the error introduced by $\delta\beta_k$ and $\delta\gamma_k$.
Each estimate and error are assumed independent.

We seek a set of parameters a_k such that the estimate for the angle of skew is optimal:

$$\hat{\theta} = \sum_{k=1}^{N} a_k \theta_k \tag{24}$$

This can be expressed as a minimisation problem, ie. minimise $Var\{\hat{\theta}\}$ subject to $E\{\hat{\theta}\} = \theta$, and traditionally solved with Lagrange multipliers. Treating a_k's as a vector $\mathbf{a}$, the system of equations are

$$\begin{aligned} &\nabla_{\mathbf{a}} Var\{\hat{\theta}\} - \lambda \nabla_{\mathbf{a}} \left(\sum_{k=1}^{N} a_k \right) = 0 \\ &\sum_{k=1}^{N} a_k = 1 \end{aligned} \tag{25}$$

which are $N+1$ linear equations in the same number of unknowns.

Solving this will give

$$a_i = \frac{1}{\sigma_i^2 \sum_{k=1}^{N} \frac{1}{\sigma_k^2}} \tag{26}$$

where $\sigma_i^2 = E\{\delta\theta_j^2\}$ is the variance of the errors, with $E\{\delta\theta_j \delta\theta_k\} = 0$ for $j \neq k$.

We can combine this with perturbation analysis:

$$\begin{aligned} \delta\theta &= \frac{\partial g}{\partial \beta}\delta\beta + \frac{\partial g}{\partial \gamma}\delta\gamma \\ &= m_\beta \, \delta\beta + m_\gamma \, \delta\gamma \end{aligned} \tag{27}$$

with substitutions to m's for convenience.

Hence

$$\begin{aligned} E\{\delta\theta_i^2\} &= E\left\{(m_{\beta i}\delta\beta_i + m_{\gamma i}\delta\gamma_i)^2\right\} \\ &= (m_{\beta i}^2 + m_{\gamma i}^2)\, \sigma^2 \end{aligned} \tag{28}$$

where $\sigma^2 = E\{\delta\beta_i^2\} = E\{\delta\gamma_i^2\}$ assuming that $\delta\beta_i$, $\delta\gamma_i$ are independent with zero means and have statistically the same distribution.

Putting this into (26) gives

$$a_i = \frac{1}{(m_{\beta i}^2 + m_{\gamma i}^2) \sum_{k=1}^{N} \frac{1}{m_{\beta k}^2 + m_{\gamma k}^2}} \tag{29}$$

which are the weights needed to obtain the optimal estimate for the angle of skew.

A.4 Error Analysis in Spline Fitting

The calculation of the spline coordinates at parameter t may be formulated as

$$\hat{\mathbf{x}}(t)' = \mathbf{b}'(t)\,\mathbf{P} \tag{30}$$

where

$$\mathbf{b}'(t) = [\ t^{k-1} \quad t^{k-2} \quad \cdots \quad 1\ ]\,[\ \mathbf{f}_1 \quad \mathbf{f}_2 \quad \cdots \quad \mathbf{f}_k\ ]\,\mathbf{G} \tag{31}$$

$\mathbf{f}$'s are column vectors of polynomial coefficients for each polynomial piece in the basis function, k is the order of the spline, and $\mathbf{G}$ is a matrix dependent on the integer quotient of t which is used to select the relevant control points for the calculation. The other variables are as described in section 5.

Let

$$C = \begin{bmatrix} 2\mathbf{B}'\mathbf{B} & -\mathbf{b_1} & -\mathbf{b_n}t \\ \mathbf{b}'_\mathbf{1} & 0 & 0 \\ \mathbf{b}'_\mathbf{n} & 0 & 0 \end{bmatrix} \tag{32}$$

which is the non-singular square matrix shown in equation 11. For simplicity, we define $\mathbf{M}$ as the upper portion of $\mathbf{C^{-1}}$ and $\mathbf{W}$ as shown.

$$\begin{bmatrix} \mathbf{M} \\ \mathbf{r}'_{\mathbf{n+1}} \\ \mathbf{r}'_{\mathbf{n+2}} \end{bmatrix} = \mathbf{C^{-1}} \quad , \quad \mathbf{W} = \begin{bmatrix} 2\mathbf{B}'\mathbf{X} \\ \mathbf{x}'_\mathbf{1} \\ \mathbf{x}'_\mathbf{n} \end{bmatrix} \tag{33}$$

The least squares solution for the control points is $\hat{\mathbf{P}} = \mathbf{MW}$ leaving an error matrix $\mathbf{E}$. Since x and y coordinates are assumed independent, there is no loss in generality by only considering the spline fit for the x ordinate of the boundary point positions. The squared-error magnitude of the least squares solution is given by

$$\epsilon'_x \epsilon_x = (\mathbf{B}\hat{\mathbf{p}}_\mathbf{x} - \mathbf{x})'(\mathbf{B}\hat{\mathbf{p}}_\mathbf{x} - \mathbf{x}) \tag{34}$$

The unbiased estimate of the variance in the x values is

$$\hat{\sigma}^2 = \frac{\epsilon'_x \epsilon_x}{b - n + 2} \tag{35}$$

where b is the number of boundary points, n is the number of control points, and $n - b - 2$ represents the number of degrees of freedom in the regression (including the two additional constraints on the end points of the spline.

The covariance matrix of the positions of control points may be derived as follows:

$$\begin{aligned} Cov(\hat{\mathbf{p}}_x) &= E\{\delta\mathbf{p}\,\delta\mathbf{p}'\} = E\{(\mathbf{M}\,\delta\mathbf{w}_\mathbf{x})(\delta\mathbf{w}'_\mathbf{x}\,\mathbf{M}')\} \\ &= \mathbf{Q}\,\hat{\sigma}^2_x \end{aligned} \tag{36}$$

where

$$\delta \mathbf{w_x} = \begin{bmatrix} 2\mathbf{B}'\delta\mathbf{x_x} \\ \delta x_1 \\ \delta x_n \end{bmatrix} \quad , \quad \mathbf{Q} = \mathbf{M} \begin{bmatrix} 4\mathbf{B}'\mathbf{B} & 0 \\ 0 & 0 \end{bmatrix} \mathbf{M}' \tag{37}$$

which is based on the assumption that the end sample points $\mathbf{x_1}$ and $\mathbf{x_n}$ are exact (ie. δx_1 and δx_n are zero), since the spline is forced to lie on these.

Furthermore the variances and covariances of the estimated boundary position and its derivatives at t may be obtained. Two of these are shown below as examples:

$$Var(\hat{x}(t)) = \mathbf{b}'\mathbf{Q}\mathbf{b}\hat{\sigma}_x^2 \tag{38}$$

$$Cov(\hat{x}(t), \hat{\dot{x}}(t)) = \mathbf{b}'\mathbf{Q}\dot{\mathbf{b}}\hat{\sigma}_x^2 \tag{39}$$

Derivation of the variance and covariance of various terms may then be carried out in a similar manner.

A.5 Covariance Matrix of Symmetricity State-Space

In order to obtain the covariance matrix $\mathbf{K}$ of the symmetricity state-space in section 3.3, we need to consider the variances of $v = \kappa_a cos^3\sigma_b$ and $u = \kappa_b cos^3\sigma_a$, which may be derived from the results shown in appendix A.4.

The variance of the curvature term may be found by

$$\kappa = \frac{\ddot{y}\dot{x} - \ddot{x}\dot{y}}{(\dot{x}^2 + \dot{y}^2)^{3/2}}, \qquad \delta\kappa = (\nabla_{\mathbf{p}_x}\kappa)'\delta\mathbf{p}_x + (\nabla_{\mathbf{p}_y}\kappa)'\delta\mathbf{p}_y$$

$$\begin{aligned} (\nabla_{\mathbf{p}_x}\kappa)' &= \frac{1}{(\dot{x}^2 + \dot{y}^2)^{3/2}} \left((\dot{y}\ddot{\mathbf{b}}' - \ddot{y}\dot{\mathbf{b}}') - 3\dot{x}\frac{\ddot{y}\dot{x} - \ddot{x}\dot{y}}{\dot{x}^2 + \dot{y}^2}\dot{\mathbf{b}}' \right) \\ (\nabla_{\mathbf{p}_y}\kappa)' &= \frac{1}{(\dot{x}^2 + \dot{y}^2)^{3/2}} \left((\dot{x}\ddot{\mathbf{b}}' - \ddot{x}\dot{\mathbf{b}}') - 3\dot{y}\frac{\ddot{y}\dot{x} - \ddot{x}\dot{y}}{\dot{x}^2 + \dot{y}^2}\dot{\mathbf{b}}' \right) \end{aligned} \tag{40}$$

and

$$Var(\kappa) = (\nabla_{\mathbf{p}_x}\kappa)'\mathbf{Q}(\nabla_{\mathbf{p}_x}\kappa)\hat{\sigma}_x^2 + (\nabla_{\mathbf{p}_y}\kappa)'\mathbf{Q}(\nabla_{\mathbf{p}_y}\kappa)\hat{\sigma}_y^2 \tag{41}$$

The derivation of the variances for the angles σ_a and σ_b may be carried out similarly. For clarity, a change in notation $\alpha = \sigma_a$ will be used.

$$\alpha = \tan^{-1}\frac{\dot{y}_a}{\dot{x}_a} - \tan^{-1}\frac{y_b - y_a}{x_b - x_a}, \qquad \delta\alpha = (\nabla_{\mathbf{p}_x}\alpha)'\delta\mathbf{p}_x + (\nabla_{\mathbf{p}_y}\alpha)'\delta\mathbf{p}_y$$

where (x_a, y_a) are the coordinates of the point a and similarly for point b.

$$
\begin{aligned}
\nabla_{\mathbf{p}_x}\alpha &= -\frac{\dot{y_a}}{\dot{x_a}^2+\dot{y_a}^2}\,\dot{\mathbf{b}}_{\mathbf{a}} + \frac{y_b-y_a}{(x_b-x_a)^2+(y_b-y_a)^2}\,(\mathbf{b}_b-\mathbf{b}_a) \\
\nabla_{\mathbf{p}_y}\alpha &= \frac{\dot{x_a}}{\dot{x_a}^2+\dot{y_a}^2}\,\dot{\mathbf{b}}_{\mathbf{a}} + \frac{x_b-x_a}{(x_b-x_a)^2+(y_b-y_a)^2}\,(\mathbf{b}_b-\mathbf{b}_a) \qquad (42)
\end{aligned}
$$

and

$$Var(\alpha) = (\nabla_{\mathbf{p}_x}\alpha)'\mathbf{Q}(\nabla_{\mathbf{p}_x}\alpha)\hat{\sigma}_x^2 + (\nabla_{\mathbf{p}_y}\alpha)'\mathbf{Q}(\nabla_{\mathbf{p}_y}\alpha)\hat{\sigma}_y^2 \qquad (43)$$

Using (40) and (42), the covariance between κ and α may also be obtained:

$$Cov(\kappa,\alpha) = (\nabla_{\mathbf{p}_x}\kappa)'\mathbf{Q}(\nabla_{\mathbf{p}_x}\alpha)\hat{\sigma}_x^2 + (\nabla_{\mathbf{p}_y}\kappa)'\mathbf{Q}(\nabla_{\mathbf{p}_y}\alpha)\hat{\sigma}_y^2 \qquad (44)$$

In order to obtain variances for u and v, we consider the differentials:

$$
\begin{aligned}
\delta u &= \delta(\kappa_b \cos^3\alpha) = \cos^3\alpha\,\delta\kappa_b - 3\kappa_b\cos^2\alpha\sin\alpha\,\delta\alpha \qquad (45) \\
\delta v &= \delta(\kappa_a \cos^3\beta) = \cos^3\beta\,\delta\kappa_a - 3\kappa_a\cos^2\beta\sin\beta\,\delta\beta \qquad (46)
\end{aligned}
$$

where β is only a notation change (like α) and equivalent to the angle σ_b. Assuming that the points a and b are on different splines, the variances and covariances of u and v are as given below:

$$
\begin{aligned}
Var(u) &= \cos^6\alpha\,Var(\kappa_b) + 9\kappa_b^2\cos^4\alpha\sin^2\alpha\,Var(\alpha) \qquad (47) \\
Var(v) &= \cos^6\beta\,Var(\kappa_a) + 9\kappa_a^2\cos^4\beta\sin^2\beta\,Var(\beta) \qquad (48) \\
Cov(u,v) &= -3\kappa_a\cos^3\alpha\cos^2\beta\sin\beta\,Cov(\kappa_b,\beta) - \\
&\quad 3\kappa_b\cos^3\beta\cos^2\alpha\sin\alpha\,Cov(\kappa_a,\alpha) \qquad (49)
\end{aligned}
$$

Finally, the covariance matrix $\mathbf{K}$ is given by

$$\mathbf{K} = \begin{pmatrix} Var(v) & Cov(u,v) \\ Cov(u,v) & Var(u) \end{pmatrix} \qquad (50)$$

which by substitution from previous equations may be evaluated for a particular state (v, u).