A Modal Approach to Feature–based Correspondence

Larry S. Shapiro and J. Michael Brady

Robotics Research Group
Department of Engineering Science
Oxford University OX1 3PJ
United Kingdom

Abstract

We propose a novel method for performing point–feature correspondence based on a modal shape description. Introducing shape information into a low–level matching process allows our algorithm to cope easily with rotations and translations in the image plane, yet still give a dense correspondence. We also show positive results for scale changes and small skews, and demonstrate how reflectional symmetries can be detected.

1 Introduction

The correspondence problem is a generic one in computer vision, arising whenever a match must be found between the features of two related patterns. It emerges in paradigms as diverse as stereo fusion, structure–from–motion, model–based recognition and navigation. In this paper, we propose a novel token–based technique for matching two sets of point features, using a modal shape description.

Our starting point is the work of Scott and Longuet–Higgins, reviewed in Section 2. After identifying two shortcomings of their algorithm, we outline our modal approach (Section 3) and give some experimental results (Section 4). We conclude in Section 5 with a set of open problems and directions for future research.

2 Previous work

The correspondence problem has long intrigued computer vision researchers, and a vast literature is testimony to the numerous techniques proposed to solve it. There are two main classes of algorithm, *intensity–based* and *feature–based*. The former operate directly on the raw grey–level data in images, while the latter employ preprocessing stages to extract features such as corners, edges or blobs. We adopt the feature–based approach for various reasons, among them the undesirability of dense temporal sampling in our application (mobile robot navigation).

In a recent paper [1], Scott and Longuet–Higgins proposed an algorithm to match 2D point–features across a pair of images. Consonant with the "minimal mapping" philosophy of Ullman [2], they incorporated an affinity measure between features (based on inter–element distances) and a competition scheme allowing candidate features to contest matches. These criteria were formulated as a *principle of proximity*

BMVC 1991 doi:10.5244/C.5.11

(favouring matches across shorter distances) and a *principle of exclusion* (favouring one–to–one matches). The resulting mapping then effectively minimised the overall sum of the squared distances travelled by the features, subject to the uniqueness constraint.

A remarkable feature of the algorithm was its elegant implementation, founded on a well–conditioned eigenvector solution which involved no iteration. As input, the algorithm received a set of m features ($\mathbf{x}_{i,1}$) in image I_1 and n features ($\mathbf{x}_{j,2}$) in image I_2. The computation then consisted of three stages. The first step was to enumerate all possible pairwise matches and store their affinities in a *proximity matrix* $\mathbf{G}$. Each element G_{ij} recorded the attraction between the i^{th} feature in I_1 and the j^{th} feature in I_2 via a Gaussian–weighted distance metric

$$G_{ij} = e^{-d_{ij}^2/2\sigma^2}, \quad i = 1 \ldots m, \; j = 1 \ldots n$$

where $d_{ij}^2 = \| \mathbf{x}_{i,1} - \mathbf{x}_{j,2} \|^2$ was the squared Euclidean distance between the two features. Thus G_{ij} ranged from 0 for widely–separated features ($d_{ij} = \infty$) to 1 for coincident features ($d_{ij} = 0$). The parameter σ controlled the degree of interaction between the two sets of features. This was roughly like placing a small circle around a feature in I_1 and only allowing it to interact with those features in I_2 lying within the circumference. Hence, a small value of σ enforced local interactions, while a larger value permitted more global interactions.

The second step was to perform a *singular value decomposition* (SVD) of $\mathbf{G}$, i.e., express $\mathbf{G}$ as

$$\mathbf{G} = \mathbf{TDU}.$$

The $\mathbf{T}$ and $\mathbf{U}$ matrices were orthogonal, i.e., their rows (and columns) were mutually perpendicular and had unit length. The $\mathbf{D}$ matrix contained the (positive) singular values along its diagonal in descending numerical order.

The final step was to compute the correlation (in a scalar product sense) between $\mathbf{T}$'s rows and $\mathbf{U}$'s columns, giving an *association matrix* $\mathbf{P}$,

$$\mathbf{P} = \mathbf{TEU},$$

where $\mathbf{E}$ was obtained by replacing each diagonal element in $\mathbf{D}$ by a 1. The element P_{ij} then indicated the strength of attraction between features $\mathbf{x}_{i,1}$ and $\mathbf{x}_{j,2}$, where 1 indicated a perfect match and 0, no match at all. The correspondence between the two features was "strong" only if P_{ij} was largest in both its row and its column, implying a mutual consent to the match. When P_{ij} was largest in its column but not its row (or vice versa), a "weak" correspondence was implied, with multiple features competing for the match.

This algorithm was shown [1] to maximise the trace of $\mathbf{P}^\mathsf{T}\mathbf{G}$. In other words, the $\mathbf{P}$ matrix was effectively a "mask" which slotted over $\mathbf{G}$ and selected the biggest elements. Since G_{ij} was large precisely when d_{ij}^2 was small, an overall minimum squared distance mapping was ensured. This can be understood intuitively by imagining pieces of string tied between matching features with the aim of minimising the total amount of string used. At the same time, however, $\mathbf{P}$ was orthogonal, so there could only be *one* large element per row or column. Hence, no feature in I_1 could be strongly coupled with more than one feature in I_2, neatly giving the exclusion principle.

Empirical results in [1] showed that the algorithm successfully matched random scatter diagrams undergoing 2D translations, expansions and shears. A theoretical proof confirmed that the algorithm would always recover these mappings, provided that σ was sufficiently large. Our investigation of the algorithm's performance on real-world objects confirmed this claim, but we draw attention to two weaknesses.

Firstly, the algorithm does not cope with large rotations in the image plane. An example is given in Figure 1(a), where features on the outline of a bottle have been rotated about the optic axis by 80°. Clearly, the algorithm fails to generate the correct correspondence; the correct matches (obtained using our algorithm) appear in Figure 1(b). This failure is a consequence of treating all features as equal, with no regard for the structure *within* the image. For instance, Figure 1(a) contains violations of the continuity constraint (whereby neighbouring features in I_1 should remain neighbours in I_2).

Secondly, implementation considerations can make the assumption of large σ unreasonable, since this forces the smaller singular values towards zero. Consequently, some columns in $\mathbf{T}$ (and rows in $\mathbf{U}$) become unstable and the association matrix becomes incorrect. This phenomenon is so marked that sometimes even a simple 1D translation cannot be properly matched [3], although the exact effect obviously varies from computer to computer (depending on the maths processor).

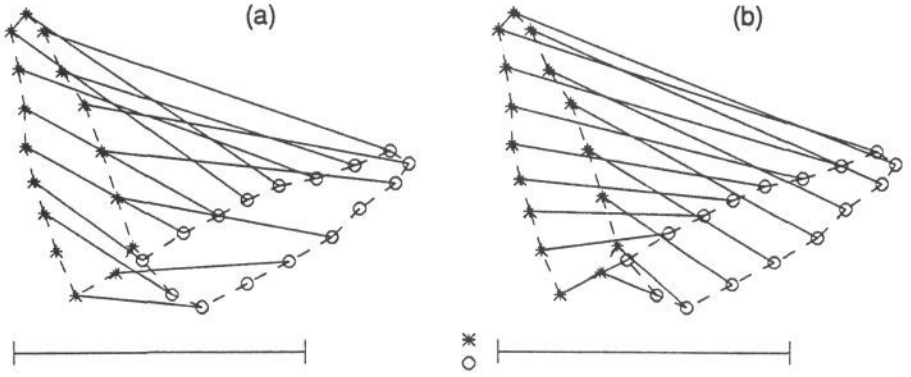

Figure 1: *A bottle rotated about the optic axis by* 80°. *Asterisks indicate image* I_1, *circles indicate image* I_2, *and the bar indicates* σ: *(a) output of the algorithm in Section 2 (only strong attractions are shown, which is why certain features remain unmatched); (b) output of our algorithm (correct solution).*

3 The modal approach

In devising our solution, we sought to retain the clear strengths of the above algorithm while overcoming its weaknesses. It was apparent that in order to deal with rotations, the structure inherent within the individual images would have to be accounted for, requiring a *shape representation*. The algorithm presented here first analyses each image separately to extract its *modes*, and then uses these modes to establish correspondence. The modes essentially encode the "shape" of the scene based on the inter-feature distances, and so constitute a shape description.

To gain an intuitive understanding of the approach, consider an image with m features, and suppose that we define a set of m axes to act as a coordinate system in an m–dimensional space. Each image–feature is then assigned a coordinate in this higher space, i.e., each feature is mapped from its 2D image–plane into a hyperspace with m axes. We perform such a mapping independently for each image, and when the "shapes" of the images are similar, features which belong together will coincide in the hyperspace. We outline the algorithm below and then demonstrate its operation with a worked example.

3.1 The algorithm

Consider first how to form the modes of a single image having m features $\mathbf{x}_i$. A square *proximity matrix* $\mathbf{H}$ is created, recording the distances between features *within* the image. In other words, here we use *intra–image* distances, rather than *inter–image* distances. We employ the same Gaussian–weighted metric as before,

$$H_{ij} = e^{-r_{ij}^2/2\sigma_x^2} ,$$

where $r_{ij}^2 = \| \mathbf{x}_i - \mathbf{x}_j \|^2$. Evidently, $\mathbf{H}$ is symmetric ($r_{ij} = r_{ji}$) and its diagonal elements are unity ($r_{ii} = 0$). The parameter σ_x again controls the interaction between features, with the subscript x emphasising that interaction occurs between features in the *same* image rather than features in two different images. For small σ_x, then, each feature only has knowledge of its local surround, while for large σ_x, each feature is more globally aware. Since the two analyses proceed independently, the value of σ can be different for the two images, say σ_1 and σ_2 respectively.

Next, we compute the eigenvalues λ_i and eigenvectors E_i of $\mathbf{H}$, i.e., solve

$$\mathbf{H}E_i = \lambda_i E_i, \quad i = 1 \ldots m,$$

The eigenvectors are of unit length and are mutually orthogonal, and hence form an orthonormal basis. They are termed *modes* by analogy with mechanical oscillating systems, whose eigenvectors describe the modes of vibration[1]. In matrix form,

$$\mathbf{H} = \mathbf{V}\mathbf{D}\mathbf{V}^\top.$$

The diagonal matrix $\mathbf{D}$ contains the (positive) eigenvalues along its diagonal in decreasing size. The *modal matrix* $\mathbf{V}$ is orthogonal and has the eigenvectors as its column vectors (i.e., $\mathbf{V} = [E_1 \mid \cdots \mid E_m]$). Each row of $\mathbf{V}$ can then be thought of as a *feature vector* F_i, containing the m modal coordinates of feature i, i.e., the expansion of the i^{th} image feature along the m modal axes:

$$\mathbf{V} = \begin{bmatrix} F_1 \\ \vdots \\ F_m \end{bmatrix}.$$

This computation proceeds simultaneously for the two images. That is, for image I_1 (m features) we obtain $\mathbf{H}_1 = \mathbf{V}_1\mathbf{D}_1\mathbf{V}_1^\top$ while for image I_2 (n features) we obtain $\mathbf{H}_2 = \mathbf{V}_2\mathbf{D}_2\mathbf{V}_2^\top$. The associated feature vectors are written $F_{i,1}$ and $F_{j,2}$.

[1]The mechanism for generating these modes was described in [4], where they were used to group image features into "natural clusters". We have employed these modes as a shape description and have extended them to allow matching between images.

The final stage is to correlate the two sets of feature vectors, yielding the association matrix $\mathbf{Z}$. As before, Z_{ij} reflects the confidence in the match between $\mathbf{x}_{i,1}$ and $\mathbf{x}_{j,2}$. Here, three points should be noted. Firstly, because the images have different numbers of features, their number of modes will differ. We therefore truncate the $|\, m - n \,|$ least significant modes from the image with more features, where "least significant" is indicated by smallest eigenvalues. Thus both modal matrices will have k columns (or modes)[2], where $k = \min\{m, n\}$. Effectively, we have discarded components of the feature vectors along the least important axes.

Secondly, the sign of each eigenvector is not unique, since switching its direction does not violate the orthonormality of the basis. However, it is vital that both sets of axes have *consistent* directions since we wish to directly compare the feature vectors. A sign correction stage is thus necessary. Briefly, we treat $\mathbf{V}_1$ as the reference basis and proceed to orient the axes in $\mathbf{V}_2$ one at a time, choosing for each that direction which maximally aligns the two sets of feature vectors (expanded only along the axes which have already been untangled). Further details are given in [3].

Thirdly, the association matrix $\mathbf{Z}$ differs from $\mathbf{P}$ in that a perfect match is now indicated by the value 0, while a value of 2 indicates no match at all. Best matches are thus given by elements in $\mathbf{Z}$ which are *smallest* in their row and column. The values Z_{ij} are obtained by taking the Euclidean distance between feature vectors

$$Z_{ij} = \| F_{i,1} - F_{j,2} \|^2,$$

rather than their scalar product. The advantages of this approach are robustness to truncation of inessential modes, improved sensitivity (due to an increased range of values) and a convenient interface to the sign correction algorithm.

3.2 An example

Two sets of features are shown in Figure 2 and the aim is to discover the correct correspondence. For the first image, we choose $\sigma_1 = 4$ and obtain

$$\mathbf{H}_1 = \begin{bmatrix} 1.00 & 0.86 & 0.78 & 0.57 \\ 0.86 & 1.00 & 0.97 & 0.40 \\ 0.78 & 0.97 & 1.00 & 0.44 \\ 0.57 & 0.40 & 0.44 & 1.00 \end{bmatrix}, \quad \mathbf{V}_1 = \begin{bmatrix} 0.53 & 0.02 & 0.81 & -0.24 \\ 0.54 & -0.34 & -0.12 & 0.76 \\ 0.54 & -0.29 & -0.52 & -0.60 \\ 0.37 & 0.90 & -0.23 & 0.10 \end{bmatrix}.$$

Similarly, for the second image we set $\sigma_2 = 4$ and obtain (after sign correction)

$$\mathbf{H}_2 = \begin{bmatrix} 1.00 & 0.78 & 0.94 & 0.73 \\ 0.78 & 1.00 & 0.94 & 0.44 \\ 0.94 & 0.94 & 1.00 & 0.61 \\ 0.73 & 0.44 & 0.61 & 1.00 \end{bmatrix}, \quad \mathbf{V}_2 = \begin{bmatrix} 0.53 & 0.09 & 0.71 & -0.45 \\ 0.49 & -0.51 & -0.57 & -0.41 \\ 0.54 & -0.25 & 0.12 & 0.79 \\ 0.42 & 0.82 & -0.39 & 0.03 \end{bmatrix}.$$

Each row in $\mathbf{V}_1$ and $\mathbf{V}_2$ refers to an image feature, and thus if the (arbitrary) numbering of two features in an image is changed, their feature vectors simply change positions in $\mathbf{V}$. Finally, the relative similarities between the two sets of features are

[2] In fact, fewer than k modes can also be used.

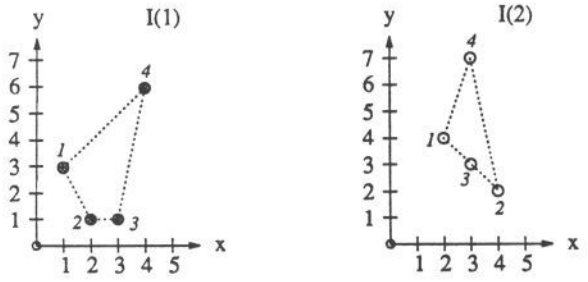

Figure 2: *Two images to be matched, each having four features. Image I_2 (open dots) is a distorted version of I_1 (solid dots), and has moved upwards and to the right. The features have been labelled arbitrarily, and the correct solution is* 1–1, 2–3, 3–2 *and* 4–4.

summarised in the association matrix $\mathbf{Z}$

$$\mathbf{Z} = \begin{bmatrix} \underline{0.06} & 2.22 & 1.62 & 2.18 \\ 2.33 & 1.60 & \underline{0.07} & 1.96 \\ 1.68 & \underline{0.09} & 2.35 & 1.65 \\ 1.87 & 2.37 & 1.94 & \underline{0.04} \end{bmatrix}.$$

The rows in $\mathbf{Z}$ index features in I_1 while the columns index features in I_2. The fourth row, for instance, derives from comparing the fourth row in $\mathbf{V}_1$ (feature vector $F_{4,1}$) with the four rows in $\mathbf{V}_2$ (feature vectors $F_{j,2}$). The smallest elements are underlined and give the correct answer (see Figure 2 caption). Thus, the second feature in I_1 matches the third feature in I_2 (element Z_{23}), and so on.

4 Results

Because the modes are based solely on the distances between features, the shape description is unaffected by transformations which preserve these distances (such as rotations, translations and reflections in the image plane). This is demonstrated clearly in Figures 1(b) and 3(a) ($\sigma_1 = \sigma_2$ in both cases). The program also handles uniform scale well (Figure 3(b)), since scaling the image by S is equivalent to replacing σ by σ/S. Information about the relative sizes of σ_1 and σ_2 is contained in the eigenvalues of $\mathbf{H}_1$ and $\mathbf{H}_2$.

Furthermore, our shape description provides information about reflectional symmetries by identifying all possible matches (Figures 4(a)–(d)). These symmetries are detected when switching the sign of the axis has no detrimental effect on the match. In such cases, the number of arbitrary sign choices equals the number of (orthogonal) reflectional symmetries. Finally, the modes have proven to be fairly robust to distortions in the image, so long as the global shape doesn't change substantially. Figure 4(e) shows the performance of the algorithm when faced with small perspective distortion. However, for large slant (Figure 4(f)), the "shape" of the figure changes too much and the match is unsuccessful.

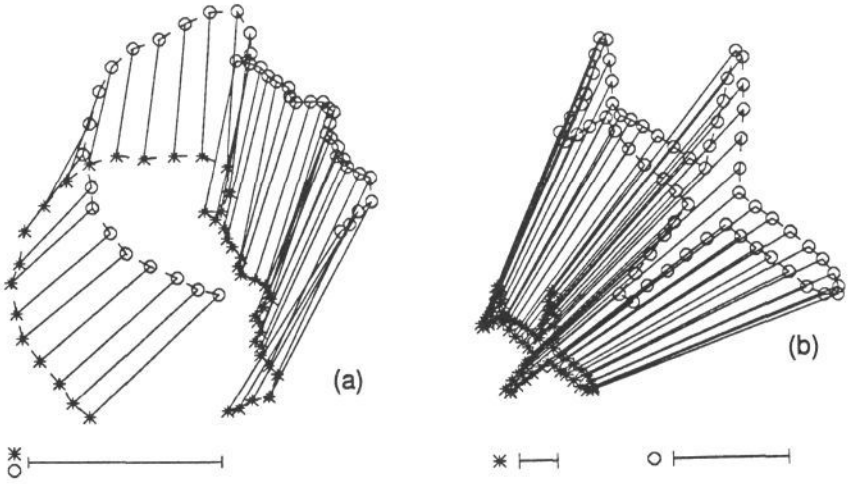

Figure 3: *The modal algorithm copes with rotation, translation and scale: (a) outline of a head translated and rotated in the image plane ($\Delta X = 5$, $\Delta Y = 3$, $\theta = 25^0$); (b) an aeroplane scaled by a factor of $2\frac{1}{2}$. Here, σ_1 and σ_2 are unequal ($\sigma_2 = 3\sigma_1$).*

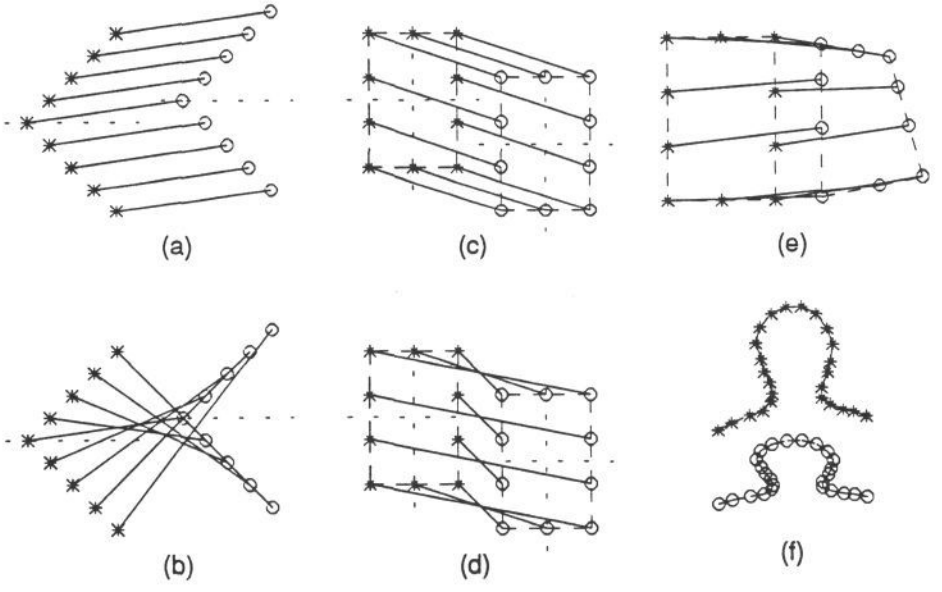

Figure 4: *The modal algorithm (dotted lines show symmetry axes): (a)(b) two possible mappings for a roof-like shape (one reflectional symmetry); (c)(d) two of four possible mappings for a rectangle (two reflectional symmetries); (e) a rectangle skewed by small perspective distortion; (f) two skewed shapes whose modes are too dissimilar to match.*

5 Conclusion

We have presented a novel correspondence algorithm which generates a modal description of an image based on the spatial distribution of its features. This shape representation allows rotations and translations in the image plane to be handled trivially. The algorithm also copes with uniform scaling and small skews, and moreover detects multiple valid matches arising from reflectional symmetries. From an implementation viewpoint, the algorithm is well–conditioned, linear in the number of features, and potentially parallelisable; moreover, it gives a one–shot solution. However, as with any global shape representation, the presence of occlusion and multiple moving objects present difficulties which have not yet been resolved.

Future research will address the feasibility of incorporating inter–image feature distances (d_{ij} in Section 2) into the algorithm; presently, only intra–image distances (r_{ij} in Section 3) are used. Rigorous tests on noise tolerance have still to be conducted, and we also intend to investigate rotational symmetries and automatic selection of the σ's.

Acknowledgments

We have benefitted greatly from discussions with Guy Scott and Christopher Longuet–Higgins. LSS is supported by an ORS Award (UK) and by a Foundation for Research Development Scholarship (RSA).

References

[1] G.L. Scott and H.C. Longuet–Higgins, "An algorithm for associating the features of two patterns", *Proceedings of the Royal Society of London,* Vol. B244, 1991, pp. 21–26.

[2] S. Ullman, *The Interpretation of Visual Motion,* MIT Press, USA, 1979.

[3] L.S. Shapiro. "Towards a Vision–Based Motion Framework", First Year Report, Department of Engineering Science, Oxford University, May 1991.

[4] G.L. Scott and H.C. Longuet–Higgins, "Feature grouping by 'relocalisation' of eigenvectors of the proximity matrix", *Proceedings of the British Machine Vision Conference (BMVC90),* Oxford University, Sept. 1990, pp. 103–108.