PARTIAL SHAPE RECOGNITION: A LANDMARK-BASED APPROACH

Nirwan Ansari
Center for Communications and Signal Processing
Electrical and Computer Engineering Department
New Jersey Institute of Technology
Newark, New Jersey 07102
(201) 596-5739.

Edward J. Delp
Computer Vision and Image Processing Laboratory
School of Electrical Engineering
Purdue University
West Lafayette, Indiana 47907
(317) 494-1740.

Abstract

When objects are occluded, many shape recognition methods that use global information will fail. To recognize partially occluded objects, we represent each object by a set of "landmarks." The landmarks of an object are points of interest relative to the object that have important shape attributes. Given a scene consisting of partially occluded objects, a model object in the scene is hypothesized by matching landmarks of the model to those in the scene. A measure of similarity between two landmarks, known as the sphericity, is used to perform the matching.

A technique, known as hopping dynamic programming, is described to guide the landmark matching. The location of the model in the scene is estimated with a least squares fit. A heuristic measure is then computed to decide if the model is in the scene.

1. Introduction

Shape recognition is a important area in pattern recognition and computer vision. We use the term *shape* to refer to the invariant geometrical properties of the relative distances among a set of static spatial features of an object. These static spatial features are known as the *shape features* of the object. After extracting the shape features from a model and a scene, a similarity measure must be used to compare the shape features. The similarity measure is referred to as a *shape measure*. The shape measure should be invariant when the object is viewed at a different scale or orientation. This does not suggest that size and orientation are not important for the shape recognition task. They are in fact important attributes that will be estimated either as a part of the shape recognition system, or as a separate task. Shape measures should thus be invariant to translation, rotation, and scaling.

The problem we address in this paper is that of recognizing and locating planar objects that may be occluded or touching each other. The literature on partial shape recognition is extensive. A review can be found in [1]. In our approach, the shape features of an object are the landmarks associated with the object. We define the *landmarks* of an object as the points of interest of the object that have important shape attributes. Examples of landmarks are corners, holes, protrusions, and high curvature points. They can be problem specific based on *a priori* knowledge. One contribution of our approach is the introduction of a local shape measure, sphericity. It can be shown [1] that all invariant functions under similarity transformations are functions of sphericity. Instead of evaluating many feature values in order to characterize the similarity between two features, we use sphericity to discriminate the dissimilarity between two landmarks. In contrast to some partial shape recognition methods [2-6], our approach is not sensitive to scale variation.

2. Landmark Extraction

It is important to note that the entire contour of an object is not needed to use landmarks to achieve recognition. The approach only requires knowledge of the positions of the landmarks of the object.

Among the extreme points, points with high curvature along the object contour are features that are most attractive. The contour, as in the case of a model, usually represents one object. However, in a general scene, when occlusion is allowed, the contour could represent merged boundaries of several objects. In this paper, we will only consider landmarks as points of high curvature along an object contour. Other problem specific types of landmarks will not be considered. Note that erroneous landmarks of objects in a scene may occur due to occlusion or noise in the scene.

Figure 1 shows the landmarks of various objects. They correspond to the extreme curvature points of the Gaussian smoothed object boundaries. Details of detecting landmarks can be found in [7].

3. Sphericity

The sphericity of a triangular transformation which maps a triangle to another triangle is a measure of similarity between the two triangles. Under the triangular transformation, the inscribed circle of one triangle is mapped onto an inscribed ellipse of the other triangle. As shown in Figure 2, the sphericity is defined as the ratio of the geometric mean to the arithmetic mean of the lengths of the principal axes of the inscribed ellipse; i.e., sphericity is $(2\sqrt{d_1 d_2})/(d_1+d_2)$. If the two triangles are similar, the sphericity is one. The less similar the two triangles, the smaller is the value of the sphericity. If the vertices of one triangle are taken to be the coordinates of

CH2809-2/89/0000-0831 $1.00 © 1989 IEEE

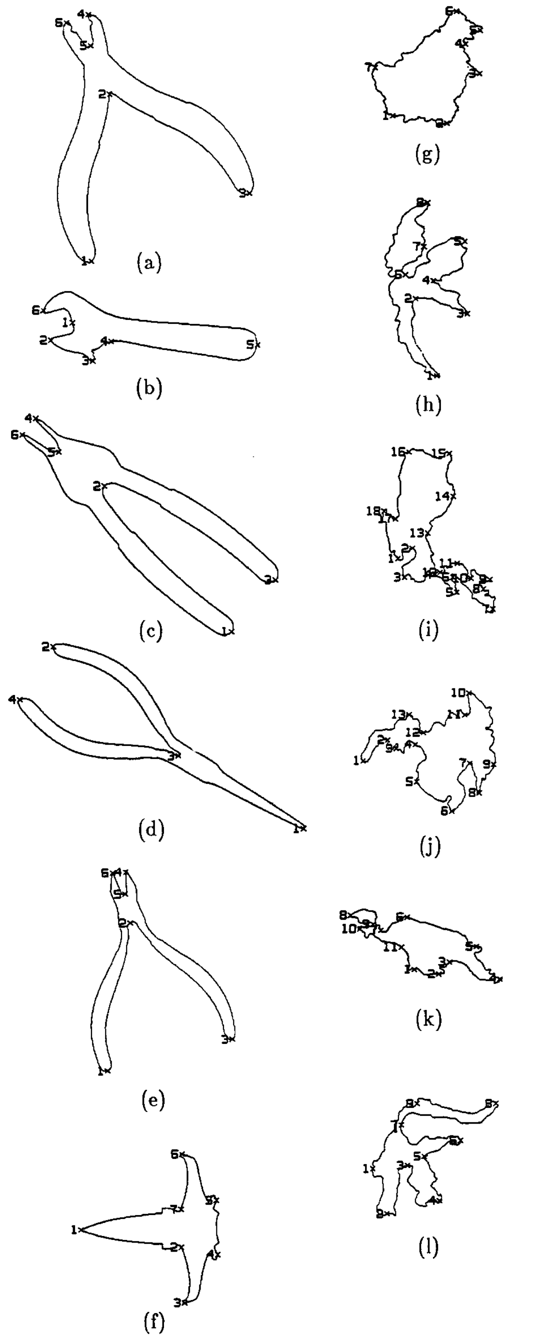

Figure 1. The landmarks of a library of objects. (a) wire stripper. (b) wrench. (c) specialty plier. (d) needle-nose plier. (e) wire cutter. (f) spacecraft. (g) Island of Borneo. (h) Island of Halmahera. (i) Island of Luzon. (j) Island of Mindanao. (k) Island of New Guinea. (l) Island of Sulawesi

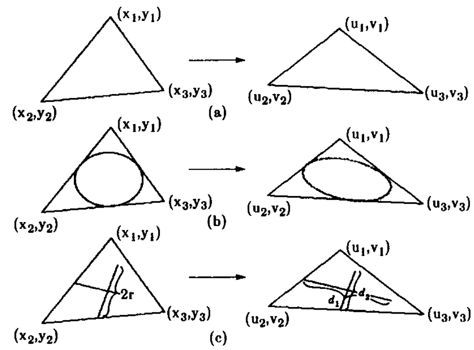

Figure 2. A mapping from a triangle to another triangle. (a) Original triangles. (b) Mapping from the inscribed circle to an inscribed ellipse. (c) Mapping of the principal axes. Sphericity is defined as $(2\sqrt{d_1 d_2})/(d_1+d_2)$.

three consecutive landmarks belonging to the model, and the vertices of the other triangle as those belonging to the scene, the sphericity is thus a measure of similarity between the model and scene landmarks.

The triangular transformation is uniquely defined by an affine transform which is a mapping of $\mathbf{x}$ to $\mathbf{u}$, where $\mathbf{x},\ \mathbf{u} \in \mathbb{R}^2$, such that

$$\mathbf{u} = \mathbf{A}\mathbf{x} + \mathrm{bt}\ ,$$

where

$$\mathbf{x} = \begin{bmatrix} x \\ y \end{bmatrix},\ \mathbf{u} = \begin{bmatrix} u \\ v \end{bmatrix},\ \mathrm{bt} = \begin{bmatrix} e \\ f \end{bmatrix},\ \mathbf{A} = \begin{bmatrix} a & b \\ c & d \end{bmatrix},\ \text{and}\ \det(\mathbf{A}) \neq \mathbf{0}\ .$$

Many properties of the sphericity such as maximal invariance under similarity transformations are discussed in [1].

The hypothesis of a model object in the scene is made by matching the model landmarks to the scene landmarks. The location of the object in the scene is then estimated by a least squares fit.

4. Hopping Dynamic Programming

Let $\{(x_1, y_1), (x_2, y_2), \cdots, (x_n, y_n)\}$ be the coordinates of a sequence of landmarks associated with a model, and $\{(u_1, v_1), (u_2, v_2), \cdots, (u_m, v_m)\}$ be those associated with the scene. The goodness of match between the ith model landmark and the jth scene landmark is given by the sphericity derived from the triangular transformation mapping

$$\{(x_{i-1}, y_{i-1}), (x_i, y_i), (x_{i+1}, y_{i+1})\}$$

to

$$\{(u_{j-1}, v_{j-1}), (u_j, v_j), (u_{j+1}, v_{j+1})\}.$$

A mapping is said to be *sense reversing* [8] if the Jacobian of the mapping is negative. To account for the sense of a mapping, sphericity is negative if the triangular transformation is sense reversing.

A table of compatibility is constructed between the model and scene landmarks. The row index i corresponds to a model landmark while the column index j corresponds to a scene landmark. The (i, j) entry of the table is the sphericity value of the triangular transformation mapping the ith model landmark and its two adjacent landmarks to the jth scene landmark and its two respective adjacent landmarks. Consider a simple example of a scene where there are two objects overlapping each other as shown in Figure 3. A table of compatibility between the wire stripper and the scene is shown in Table I. Since the landmarks of an object are obtained by tracing sequentially along the object boundary, it is likely that matches between the model and scene landmarks correspond to a sequence of high-valued entries that are diagonal to each other in the table. A brute-force approach to finding such a sequence is impractical. We will instead formulate a dynamic programming procedure to achieve the matching.

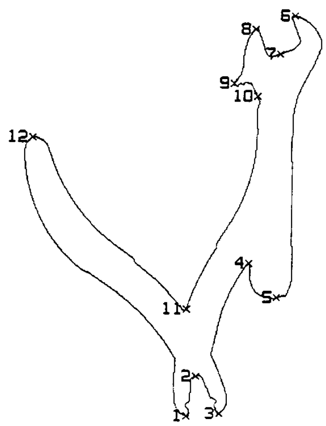

Figure 3. A scene which consists of an overlapping wire stripper and wrench. Each landmark is labeled and indicated by an "×."

Our matching procedure is somewhat similar to the feature matching algorithm of Gorman and Mitchell [6]. They used backward dynamic programming to find a minimum distance path from the first column to the last column of their augmented inter-segment distance table. Their assumption that the path must make use of all the scene features is inadequate because the scene may have extraneous or missing features due to occlusion. Instead of this assumption, we will only require that our path covers the range of either all the model landmarks or all the scene landmarks. Neither the starting point nor the destination point of a path which corresponds to a sequence of matches between the scene and model landmarks is known. Instead, a *support entry*, which is an entry in the table that provides strong evidence of a true match between a model and a scene landmark, is used to guide the matching. The evidence is strong if the entry as well as its diagonal neighboring entries have sphericity values close to one. That is, the model landmark and its neighboring landmarks match well locally with the scene landmark and its neighboring landmarks. Denote s(i, j) as the sphericity value at the (i, j) entry of the table. The (i, j) entry of the table is said to be the support entry of the table if the sum $s(i-1, j-1)+s(i, j)+s(i+1, j+1)$ is maximum. In the example shown in Table I, the support entry can either be entry (3, 12) or (4, 1). Since the sphericity is a local similarity measure, the overall goodness of the match between the model and the scene is determined by the sum of the sphericity values of those landmarks that match each other. The sequence of matches should correspond to a path in the table that passes through the support entry and maximizes the sum of the sphericity values of the path with the following two constraints:

(1) A model landmark cannot match to more than one scene landmark, and

(2) A scene landmark cannot match to more than one model landmark.

By the above two constraints, a vertical or a horizontal transition of the path should not be considered as a match between the model and the scene landmark.

Unlike the classical shortest path problem [9], we desire to find a path that passes through the support entry, rather than from a starting point to a destination point, or vice versa. Here, the support entry is treated as both a starting and a destination point. That is, we work both forward and backward from the support entry. Let (k, l) be the support entry, $a_b(i, j)$ is the accumulated sum of sphericity values from the (k, l) to (i, j) entry in the backward procedure, and $a_f(i, j)$ is the accumulated sum of sphericity values from the (k, l) to (i, j) entry in the forward procedure.

Treating the support entry as the destination point, we have the following set of transition rules for the backward procedure:

(1) $a_b(i-1,j-1)=\max\{a_b(i,j)+s(i-1,j-1), a_b(i-1,j), a_b(i,j-1)\}$

(2) $a_b(i-1,l)=\max\{s(i,l), s(i-1,l)\}$

(3) $a_b(k,j-1)=\max\{s(k,j), s(k,j-1)\}$

(4) $a_b(k,l)=s(k,l)$.

A diagonal transition according to Rule (1) implies a possible match between the (i−1)th model landmarks and the (j−1)th scene landmark, and hence the sphericity value at (i−1,j−1) is added to the accumulated sum of the sphericity at (i, j) to produce the accumulated sum of the sphericity at (i−1,j−1). Since a horizontal or vertical transition does not constitute a match, the accumulated sum of sphericity remains the same as before the transition. Rules (2) and (3) are the boundary conditions. Rule (4) is the initial condition.

Treating the support entry as the starting point, we similarly have the following set of transition rules for the forward procedure:

(1) $a_f(i+1,j+1)=\max\{a_f(i,j)+s(i+1,j+1), a_f(i+1,j), a_f(i,j+1)\}$

(2) $a_f(i+1,l)=\max\{s(i,l), s(i+1,l)\}$

(3) $a_f(k,j+1)=\max\{s(k,j), s(k,j+1)\}$

(4) $a_f(k,l)=s(k,l)$.

We need to address how to switch between the forward and backward procedure. Let $(\tilde{i}, \tilde{j})$ be the entry which the backward procedure has reached at the present stage, and $(\hat{i}, \hat{j})$ be the entry which the forward procedure has reached at the present stage. We define the backward average sphericity value at entry $(\tilde{i}, \tilde{j})$ as $a_b(\tilde{i}, \tilde{j})$ divided by the number of transitions made by the backward procedure traversing from entry (k, l) to entry $(\tilde{i}, \tilde{j})$ of the

Table I

An example of the landmark matching task between the wire stripper and the scene shown in Figure 3. (a) The table of compatibility. (b) The result of performing hopping dynamic programming using (3, 12) as the support entry. (c) The resulting path indicated by 1's is the maximum value path.

Model \ Scene	1	2	3	4	5	6	7	8	9	10	11	12
1	0.07	-0.37	0.08	-0.20	0.04	0.04	-0.31	0.19	0.17	-0.03	-0.31	0.16
2	-0.16	0.94	-0.40	0.41	-0.20	-0.08	0.98	-0.86	-0.62	0.20	1.00	-0.34
3	0.03	-0.44	0.21	-0.09	0.09	0.02	-0.28	0.32	0.12	-0.07	-0.34	1.00
4	1.00	-0.12	0.03	-0.58	0.02	0.72	-0.18	0.09	0.30	-0.02	-0.15	0.03
5	-0.12	1.00	-0.37	0.33	-0.17	-0.06	0.86	-0.76	-0.45	0.16	0.93	-0.48
6	0.02	-0.15	0.60	-0.04	0.99	0.01	-0.15	0.30	0.08	-0.54	-0.17	0.07

(a)

Model \ Scene	10	11	12	1	2	3
1	2.00	2.00	1.00	0.00	0.00	0.00
2	2.00	2.00	1.00	0.00	0.00	0.00
3	1.00	1.00	1.00	1.00	1.00	1.00
4	0.00	0.00	1.00	1.99	1.99	1.99
5	0.00	0.00	1.00	1.99	2.99	2.99
6	0.00	0.00	1.00	1.99	2.99	3.60

(b)

Model \ Scene	10	11	12	1	2	3
1	1	0	0	0	0	0
2	1	1	0	0	0	0
3	0	0	1	0	0	0
4	0	0	0	1	0	0
5	0	0	0	0	1	0
6	0	0	0	0	0	1

(c)

table. The forward average sphericity value is similarly defined. The procedure which has a larger average sphericity proceeds one stage. The algorithm continues in this fashion until the combined path of both the forward and backward procedures covers the range of either all the model or all the scene landmarks. The combined path is known as the *maximum value path*. We call this procedure hopping dynamic programming (*HDP*). Using the earlier example, and using entry (3, 12) as the support entry, HDP yields the result shown in Table I.

After determining the path, several heuristics are used to further refine the match between the model and scene landmarks along the path. From the two constraints mentioned earlier, entries along the path that result from horizontal or vertical transitions cannot be considered as matches. Only entries along the path that result from diagonal transitions are considered as possible matches. We also require that the entries along the path must be above a certain threshold before they can be considered as possible matches. A threshold value of 0.7 has been empirically developed. In the above example shown in Table I, entries (2, 11), (3, 12), (4, 1), (5, 2) are considered as possible matches. Isolated entries that have been considered as possible matches so far are then eliminated because they are not locally supported by their neighbors. The example shown in Table I does not have any isolated entry, and hence entries considered as matches remain the same.

The final step is to check the values of the entries that are considered as matches along the path. If the entry has a value that is greater than 0.95, its adjacent diagonal entries will also be considered as matches. In Table I, since all entries that are considered as matches between the model and scene landmarks have sphericity value greater than 0.95, their respective adjacent diagonal entries are considered as matches. Thus, entries (1, 10) and (6, 3) are also considered as matches. In this example, model landmarks 1, 2, 3, 4, 5, and 6 match to scene landmarks 10, 11, 12, 1, 2, and 3, respectively.

5. Location Estimation and Match Verification

Location of the object in the scene is estimated by finding a coordinate transformation consisting of translation, rotation, and scaling that maps the matched landmarks of the model to the corresponding scene landmarks in a least squares sense. A score based on the least squared error of the mapping is used to quantify the overall goodness of the match between the model and scene.

The least squared error only quantify how well a portion of the model landmarks match to the corresponding scene landmarks. It does not, however, account for the overall goodness of match. Let ϵ be the least squared error derived from the matched pairs of landmarks between the model and the scene. To account for the overall goodness of the match between the model and the scene, we use the following heuristic measure which penalizes incomplete matching of the landmarks of the model:

$$\epsilon' = \begin{cases} (1.0+(\frac{n-2}{k-2})\log_2(\frac{n-2}{k-2}))\overline{\epsilon} & \text{for } k \geq 3, \\ \infty & \text{for } k = 0,1,2. \end{cases} \quad (9)$$

where n is the total number of landmarks of the model, k is the number of model landmarks that match the scene landmarks, and $\overline{\epsilon} = \epsilon/(k(\text{scale factor}))$, i.e., $\overline{\epsilon}$ is the normalized least squared error. The scale factor is derived from the coordinate transformation. The heuristic measure, ϵ', is referred to as the *match error.* Note that when k=n, $\epsilon' = \bar{\epsilon}$; i.e., no penalty is added to the normalized least squared error when all model landmarks match those in the scene. The penalty is greater if k is smaller. In the earlier example, since all the model landmarks match those in the scene, the match error value of 0.62 is the same as the least squared error. The hypothesis of the model in the scene is finally determined by the value of the match error with a small error we accept the hypothesis while with a large error we nullify the hypothesis. The decision strategy is thus a thresholding operation. If a match error is above a threshold, the match is considered correct; otherwise, the match is considered incorrect. In our study, this threshold is set empirically.

6. Experimental Results

Consider again the scene shown in Figure 3, the results of performing the landmark matching task between the scene and each of the tool models shown in Figure 1 are summarized in Table II. Models that match well to the objects in the scene are those with the smallest match errors. Though the wire cutter is not in the scene, the match error between the wire cutter and the scene is quite small. The reason for this is that the relative positions of the landmarks of the wire cutter are similar to those of the wire stripper.

Table II
The summary of the results of matching a library of objects with the scene shown in Figure 3.

Models	Total Number of Model landmarks	Number of matched model landmarks	Match Error
wrench	6	6	1.98
needle-nose plier	4	2	∞
wire cutter	6	5	7.39
specialty plier	6	2	∞
wire stripper	6	6	0.62

Figure 4 shows a more complicated scene which consists of six overlapping objects. Compared to their respective models, the objects in the scene have been rotated and/or scaled. Compared to their respective model landmarks, some of the object landmarks in the scene are missing. With respect to each model, those landmarks in the scene not belonging to the model are considered as extraneous landmarks. The results of matching each model object of the library to the scene are summarized in Table III. Figure 5 shows the results of mapping the island of Luzon into the scene. Additional experiments involving noisy landmarks are described in [1].

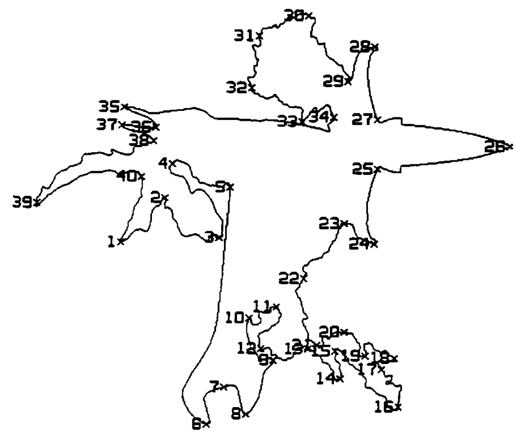

Figure 4. A scene which consists of six overlapping objects. Each landmark is labeled and indicated by an "X."

Table III
The summary of the results of matching a library of objects with the scene shown in Figure 4.

Models	Total number of model landmarks	Number of the model landmarks that match with the scene	Match Error
wrench	6	4	0.74
needle-nose plier	4	0	∞
wire cutter	6	2	∞
specialty plier	6	3	7.89
wire stripper	6	2	∞
Borneo	7	5	11.75
Halmahera	8	6	0.57
Luzon	18	14	0.78
Mindanao	13	3	54.59
New Guinea	11	4	77.81
Sulawesi	9	4	18.08
spacecraft	7	5	0.55

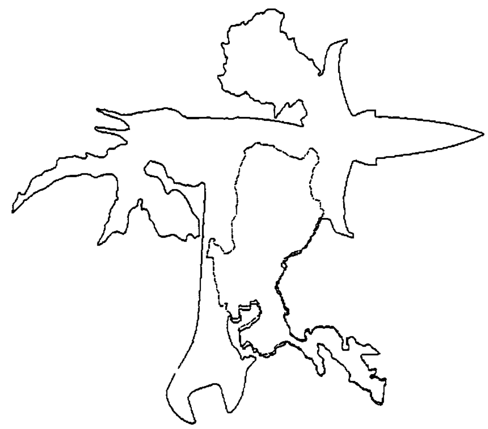

Figure 5. The result of mapping Luzon into the scene shown in Figure 4.

7. Conclusions

The experimental results have demonstrated that the landmark matching task can handle occlusion reasonably well. The performance depends on the quality of the extracted scene landmarks, and the number of correct landmarks in a scene that are detectable. When matching landmarks of a model to those in a scene, at least three landmarks in a scene that correspond to the model must be detectable. In addition, part of the sequential order of the detectable landmarks must be preserved.

References

[1] N. Ansari, *Shape Recognition: A Landmark-Based Approach*, Ph.D. Thesis, Purdue University, School of Electrical Engineering, West Lafayette, IN, August 1988. Also issued as Purdue University, School of Electrical Engineering, Technical Report, TR-EE-88-31, July 1988.

[2] R. C. Bolles and and R. A. Cain, "Recognizing and locating partially visible objects: the local-feature-focus method," *International Journal of Robotics Research*, vol. 1, no. 3, pp. 57-82, Fall 1982.

[3] T. F. Knoll and R. C. Jain, "Recognizing partially visible objects using feature indexed hypotheses," *IEEE Transactions on Robotics and Automation*, vol. RA-2, no.1, pp.3-13, March 1986.

[4] M. W. Koch and R. L. Kashyap, "Using polygons to recognize and locate partially occluded objects," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-9, no. 4, pp. 483-494, July 1987.

[5] K. E. Price, "Matching closed contours," *Proceedings of the Seventh International Conference on Pattern Recognition*, Montreal, Canada, Jul 30 - Aug 2, 1984, pp. 990-992.

[6] J. L. Turney, T. N. Mudge and R. A. Volz, "Recognizing partially occluded parts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-7, no. 4, pp. 410-421, July 1985.

[7] N. Ansari and E. J. Delp, "A note on detecting dominant points," *Proceedings of the SPIE 1989 Conference on Visual Communications and Image Processing*, Philadelphia, Nov 5-10, 1989.

[8] B. O'Neill, *Elementary Differential Geometry.* New York: Academic Press, 1966.

[9] S. E. Dreyfus and A. M. Law, *The Art and Theory of Dynamic Programming*, New York: Academic Press, 1977.