OCCLUDED OBJECT DISCRIMINATION BY A MODIFIED HOPFIELD NEURAL NETWORK

DEXIANG LUO

OCCLUDED OBJECT DISCRIMINATION BY A MODIFIED HOPFIELD NEURAL NETWORK

By

©Dexiang Luo, B.Eng

A Thesis
Submitted to the School of Graduate Studies
in Partial Fulfillment
of the Requirements of the Degree of
Master of Engineering
In
Electrical Engineering

Faculty of Engineering and Applied Sciences
Memorial University of Newfoundland
January, 2000

St. John's, Newfoundland, Canada

Abstract

This thesis presents a new method of the 2-D partially occluded object discrimination for the computer vision application. A binary modified Hopfield neural network was applied to perform the global feature matching. To obtain the feature points of the object, a Gaussian function was implemented to smooth the object boundary curve and a curvature estimation method was used to extract the dominant points. A 3-point matching method was used to perform the initial comparison and to build the disparity matrix. Finally, the coordinate transformation was used to eliminate the false matched points. Two image banks, a model object image bank and an occluded object image bank, were built for the discrimination test. The result showed that the discrimination algorithm was successful.

Keyword: *occluded object discrimination, disparity matrix, dominant points, modified Hopfield neural network, image bank, global optimization*

Acknowledgement

I am grateful to my supervisor, Dr. Ray Gosine, for his help and guidance in the past two years. I also would like to thank Faculty of Engineering & Applied Sciences and the Graduate School of Memorial University of Newfoundland for providing the financial support. Finally, I appreciate all the colleagues of the Intelligent System Group of C-CORE for their discussion and help in writing this thesis.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Problem Statement and Application

2-D object discrimination is an important issue in the pattern recognition and computer vision domain. In the past two decades, numerous work on isolated and occluded object recognition has been done in medical diagnosis, industrial automation, character recognition and military applications (Brzakovic, D. *et al.*, 1990) (Gamage, L. B. *et al.*, 1996) (Heinemann, P. H. *et al.*, 1996) (Kim, J.H. *et al.*, 1996) (McClure, J.E. and Morrow, C. T., 1987) (Meulders, S. *et al.*, 1980) (Shen, D.G. and Ip, H.H.S., 1999) (Wen, W. and Lozzi, A., 1992). Isolation means the object in the scene is complete, with no shading or overlapping, while occlusion occurs when two or more objects in a scene touch or overlap with each other.

The discrimination of an object requires an accurate shape description. The shape descriptors (Loncaric, S., 1998) can be classified into two types: *global descriptors*

and *local descriptors*. Global shape descriptors such as object contour, area, 2-D moments, Fourier descriptors and Hough transforms are robust with the isolated object recognition, but are not suitable in case of occlusion due to the difficulty in extracting the complete features of the object. Local shape descriptors, however, are usually applied to occluded object recognition. Such descriptors include corners, holes, protrusions, lines and curve-segments on the object.

Before performing the discrimination task, some image processing operations are required to extract the object characteristics such as contours and texture. Figure 1.1 shows a general approach to perform the recognition task in a computer vision system. The sample objects are compared with the models in the model image database and the matching result is evaluated.

For local shape representation, a possible solution (Ansari, N. and Delp. E.J., 1990) is to describe an object shape with a set of characteristics each defining a small part of the object. The extracted features should be well distributed along the entire object such that under partial occlusion, the remaining representation can still provide sufficient information for the accurate discrimination. It has been suggested from the viewpoint of the human visual system (Gavrila. D.M., 1998) that some dominant points along the object contour are rich in information content and are sufficient to characterize the shape of the object. A ***dominant point*** (Ansari, N. and Delp. E.J., 1991) is defined as a point in a curve at which its curvature is local maximum or minimum. In the proposed algorithm, dominant points extracted from the object contour are used as the shape attributes. The discrimination of an object in a partially occluded scene is determined by how well the dominant points of the test object and

the model object are matched to each other. The matching is usually based on the local features of the dominant points, such as the distance and the angle between each other. However, such a simple comparison has two deficiencies. First, the geographical relationships of a dominant point and other dominant points are not fully explored, which leads to the ambiguous recognition of the object identity. Secondly, a contour may be separated into a set of small isolated clusters such that none are long enough to form a significant part of the object. Identification of an unknown object based only on a small cluster of dominant points is obviously unreliable. How to achieve a global matching only based on the local attributes is a challenging problem in the domain of partially occluded object discrimination.

1.2 Approach Description

The proposed research is mainly devoted to the partially occluded object recognition. In the experiment, a model hand-tool image bank was built, the test scene images including lightly, moderately and heavily occluded model objects were tested and successful results have been achieved.

This thesis offers a new algorithm for the occluded object recognition. Dominant points along the object boundary are extracted as the shape features. Local attributes of the dominant points, such as *distance ratio* and *interior angle* between the neighboring pairs are extracted as features for the initial matching. The initial matching result is then imposed as the initial states of the *modified Hopfield neural network* (Haykin, S., 1994), and the geographical characteristics of the dominant points are used as the converging constraints. The final convergent state of the

modified Hopfield neural network is regarded as the optimal global matching result between the sample and model object. The flow chart of the discrimination algorithm is illustrated in Figure 1.2.

1.3 Thesis Structure

Chapter 2 gives the related background and comments. Chapter 3 shows the image processing methods that proceed the object discrimination. Chapter 4 details the proposed discrimination algorithm. Chapter 5 analyzes the discrimination results based on different objects, discusses the parameters and offers some suggestions for the future research. Chapter 6 concludes the work done in the thesis.

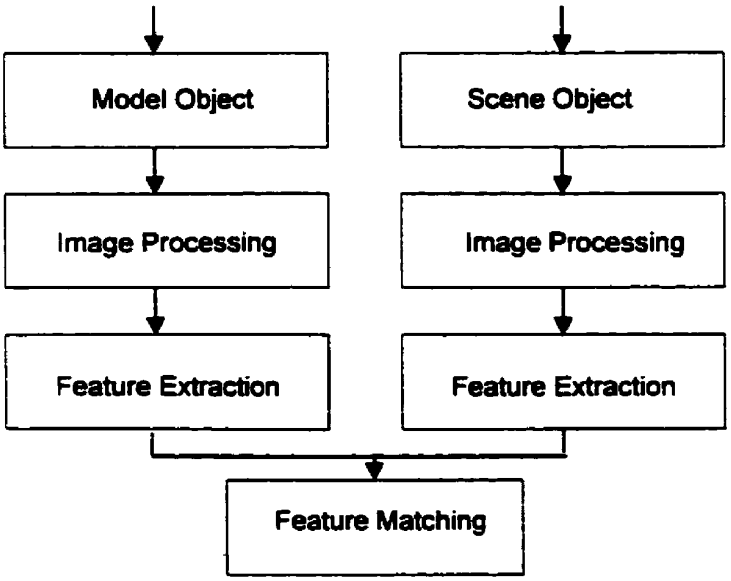


Figure 1.1 A general flow chart for object discrimination

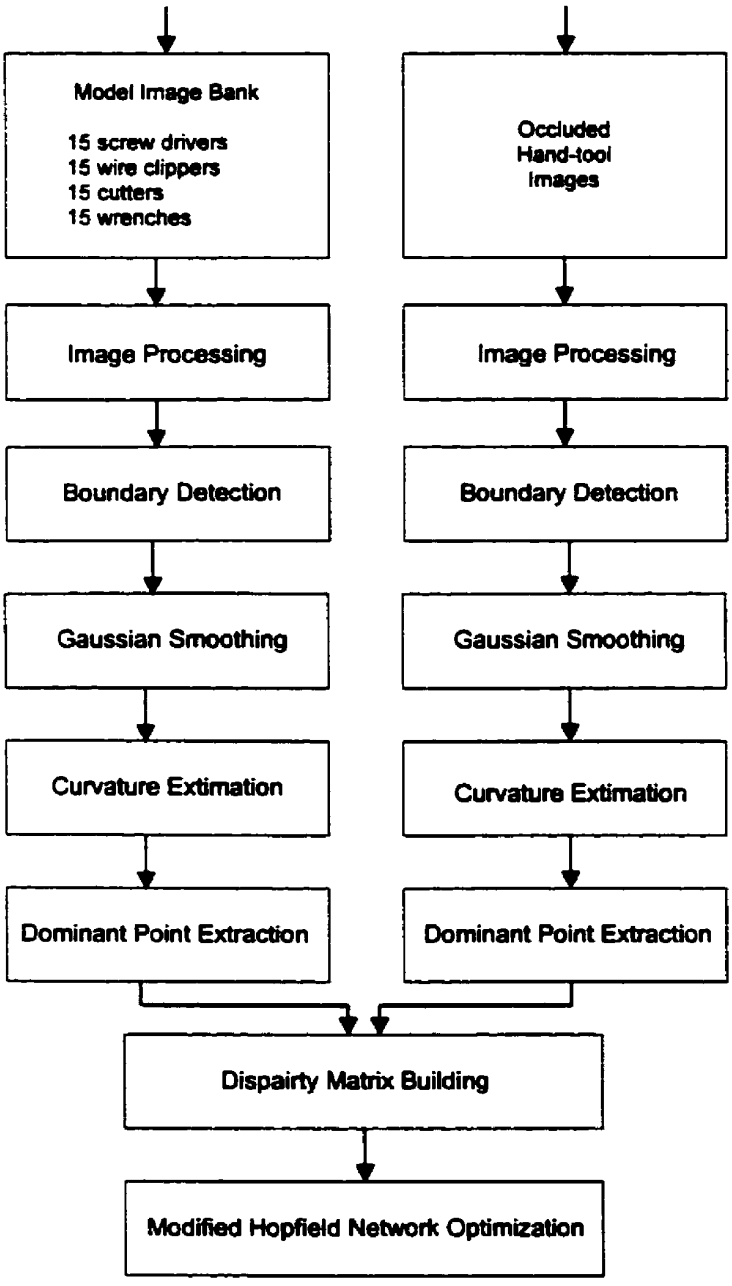


Figure 1.2 Flow chart of the thesis discrimination algorithm

Chapter 2

Background

Shape is an object property that has been carefully investigated in the past thirty years in computer vision domain. Numerous applications are involved with shape description and examples include alphanumeric optical character recognition (OCR) (Shen, D.G. and Ip, H.H.S., 1999) (Wang, S.S. *et al.*, 1994), planar shape classification (Adler, S.L. and Krishnan, B., 1998) (Cantoni, V. et al., 1998), industrial object discrimination (Ayache, N. and Faugeras, O.D., 1986) (Lamdan, Y. *et al.*, 1990), automatic inspection (Howarth, M.S. and Searcy, S.W., 1989) (Tao, Y. *et al.*, 1996), medical diagnosis (Brzakovic, D. *et al.*, 1990), biological analysis (Meulders, S. *et al.*, 1980), etc. As mentioned in Chapter 1, the first step in object discrimination is shape description. Loncaric (Loncaric, S., 1998) and Marshall (Marshall, S., 1989) summarized the shape description methods which have been developed in the past thirty years. According to their summation, despite the variety of the approaches, shape description

can be classified into different categories with different criteria, such as global and local descriptors based on representing the global or local features, region and boundary descriptors based on representing only the outside silhouette or the whole shape region. A problem in object discrimination research is how to choose an appropriate shape description method, because the choice of the method depends on the property of the shape to be discriminated and the particular application. Since the proposed topic is dealing with partially occluded object recognition, only global and local shape description methods will be discussed.

2.1 Global Shape Description

Global shape description has a long history and is comparatively mature. It is usually applied in the isolated shape representation and object discrimination. The ordinary global descriptors include: 2-D moments (Hu, M.K., 1962), Fourier descriptors (Person, E. and Fu, K, 1977) (Zahn, C.T. and Roskies, R.Z., 1972), Hough transform (Ballard, D.H., 1981), medial axis transform (Mott-Smith, J, 1970) (Beatty, D.A., 1993), shape numbers (Sonka, M. *et al.*, 1998), polygonal approximation (Pavalidis, T. and Horowitz, S.L., 1974) (Ramer, U., 1972) and morphological representation (Pitas, I. and Venetsanopoulos, A.N., 1992). Many descriptors are robust and invariant under rotation, orientation and scaling. 2-D moments and Fourier descriptors may be the most popular and classical global shape descriptors, for an overall understanding, these two methods are discussed in the following examples.

2.1.1 2-D Moments

The concept of 2-D moments was first presented by Hu in 1962 (Hu, M.K., 1962) and since then has been applied as an effective approach for shape description. It is a region-based algorithm which interprets a normalized gray-level or binary image in a density distribution function $\rho(x,y)$. Hu defined the $(p+q)th$ order moments in terms of Riemann integrals as:

$$m_{pq} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x^p y^q \rho(x,y)dxdy \qquad p,q = 0,1,2,\cdots \qquad \text{(Equation 2.1)}$$

Where x, y are the region pixel coordinates, the above function can be denoted in the digitized form as:

$$m_{pq} = \sum_{i=-\infty}^{\infty}\sum_{j=-\infty}^{\infty} i^p j^q f(i,j) \qquad \text{(Equation 2.2)}$$

Where i, j are the pixel coordinates in the digitized form. Translation invariance can be achieved when central moments are used here:

$$m_{pq} = \sum_{i=-\infty}^{\infty}\sum_{j=-\infty}^{\infty} (i-x_c)^p (j-y_c)^q f(i,j) \qquad \text{(Equation 2.3)}$$

Where (x_c, y_c) is the centroid of the region and can be obtained by:

$$x_c = \frac{m_{10}}{m_{00}}, \qquad y_c = \frac{m_{01}}{m_{00}} \qquad \text{(Equation 2.4)}$$

The scale-invariant features can also be found in the scaled central moments η_{pq}. Suppose the scale factor is α, that is, $x' = \alpha x$ and $y' = \alpha y$:

$$\eta_{pq} = \frac{\mu'_{pq}}{(\mu'_{00})^{\gamma}}$$

$$where \quad \gamma = \frac{p+q}{2} + 1, \mu'_{pq} = \frac{\mu_{pq}}{\alpha^{(p+q+2)}} \qquad \text{(Equation 2.5)}$$

Then a rotation, translation and scale invariant moment to describe the object shape can be characterized in the following heuristic forms:

$$\begin{aligned}
\phi_1 &= \eta_{20} + \eta_{02} \\
\phi_2 &= (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \\
\phi_3 &= (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \\
\phi_4 &= (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \\
&\cdots
\end{aligned} \qquad \text{(Equation 2.6)}$$

Generally, the first four terms are enough to describe the object shape. 2-D moments are frequently implemented in the shape description domain. However, since it is a region-based method, it is computationally extensive. Several papers presented fast methods for computing moments (Jian, X.Y. and Bunke, H., 1991) (Li, B. C. and Shen, J., 1991) (Li, B.C. and Shen, J., 1994), such methods just compute the moments from the object boundary, greatly improve the computation efficiency.

2.1.2 Fourier Descriptors

The Fourier descriptor used in shape description, which was first offered by Zahn and Roskies in 1972 (Zahn, C.T. and Roskies, R.Z., 1972), is a boundary-based method. Fourier descriptors require that the shape boundary is a closed curve and can be denoted by a complex function *z(t)*, usually *t* is the index variable of the pixel sequence along the

boundary, so actually $z(t)$ is a parameter-form function. If the boundary curve has N points, then $z(t)$ can be expressed as:

$$z(t) = \sum_n T_n e^{jnt} \qquad \text{(Equation 2.7)}$$

Where T_n is regarded as the Fourier descriptor. In a $T_n=a_n+b_n$ form, the Fourier descriptors can be written as following:

$$a_n = \frac{1}{N-1}\sum_{m=1}^{N-1} x_m e^{-i\frac{2\pi}{N-1}nm}$$
$$b_n = \frac{1}{N-1}\sum_{m=1}^{N-1} y_m e^{-i\frac{2\pi}{N-1}nm} \qquad \text{(Equation 2.8)}$$

Since the Fourier transform requires that the number of the boundary points is 2^n, the points have to be normalized before applying the transform. Though a_n and b_n are not invariant after the transform, their normalized magnitude:

$$r_n = (|a_n|^2 + |b_n|^2)^{\frac{1}{2}} \qquad \text{(Equation 2.9)}$$

is rotation and translation invariant.

An object shape can be described by a set of Fourier coefficients, the following table indicates the relation between the order of Fourier coefficient and the corresponding property of the object shape (Tao, Y. et al., 1996).

Table 2.1 Fourier coefficients and the implying shape properties

Fourier Coefficient	Implied Shape Property
F(0)	Average radius
F(1)	Bendingness
F(2)	Elongation
F(3)	Triangle
F(4)	Square

Fourier representation is a usual approach for shape description, it is robust, accurate and easy for shape classification (Persoon, E. and Fu, K.S., 1977) (Tao, Y *et al.*, 1995). Since the coefficients are a combination of a set of sine and cosine heuristics, it can also be viewed as a kind of regression.

2.2 Shape Discrimination with Global Description

Almost all the global shape discrimination methods can be concluded as "distance measure". The usual discrimination process is the comparison between the corresponding coefficients or the segments matching. In such a process, the square of Euclidean distance between the corresponding coefficients (same coefficient number can be achieved after element normalization) and segments (can be aligned after coordinates transformation, to be discussed later) may be used to evaluate the resemblance of the model and the test object. The zero distance means a perfect match, and a quality score of the matching can be granted based on the distance value.

The most convenient approach for the global shape discrimination is using a classifier. As stated above, usually the shape can be represented by a set of feature vectors (such as heuristic items). Some pattern recognition methods, such as statistical, syntactical, neural network or fuzzy classifiers, may be implemented to discriminate different shapes.

Wang (Wang, S.S. *et al.*, 1994) presented a shape descriptor called a *moment Fourier descriptor (MFD)*, which combines 2-D moments and Fourier descriptors, to describe complex objects. A set of *MFD* heuristic elements can uniquely represent the object shape. After normalization, similarity between the model and the test object is determined by the mean square value of the distance between the heuristic elements. This algorithm is invariant under rotation, translation and scaling of the object, and can recognize Chinese characters and keys accurately. This paper is a very good example for the object recognition by a global shape representation.

2.2.1 Chamfer ¾ Distance Transformation

Chamfer ¾ distance transformation (Borgefors. G., 1986) (Liu, H.C. and Srinath, M.D., 1990) is a popular approach for object recognition and image comparison. It converts the boundary pixels of a shape into a gray-level image where all pixels have a value corresponding to the distance to the nearest boundary pixel. In order for two shapes to be compared, one object boundary is aligned and imposed onto another. The model image is created with the boundary pixels set to zero and non-boundary pixels set to infinity. The transformation consists of two passes: a forward pass and a backward pass.

Suppose *(i, j)* is an index point of the image, the forward pass modifies the distance image as follows:

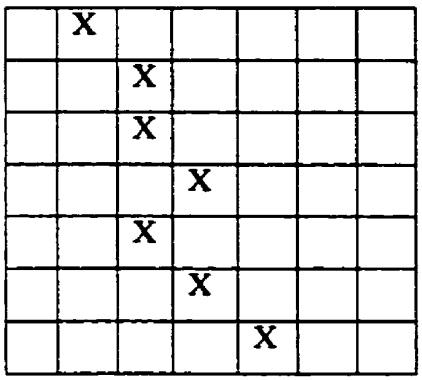
$$\begin{aligned} &for \quad i = 2,\cdots,rows \quad do \\ &\quad for \quad j = 2,\cdots,columns \quad do \\ &\qquad v_{i,j} = \min(v_{i-1,j-1} + 4, v_{i-1,j} + 3, v_{i-1,j+1} + 4, v_{i,j-1} + 3, v_{i,j}) \end{aligned}$$

Similarly, the backward pass operates as follows:

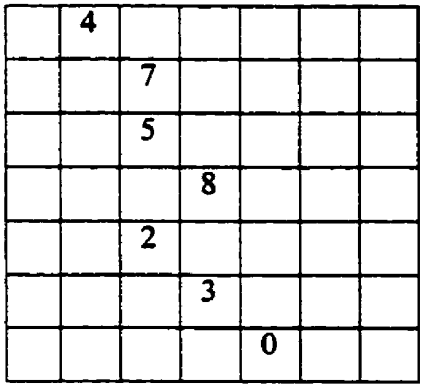
$$\begin{aligned} &for \quad i = rows - 1,\cdots,1 \quad do \\ &\quad for \quad j = columns - 1,\cdots,1 \quad do \\ &\qquad v_{i,j} = \min(v_{i,j}, v_{i,j+1} + 3, v_{i+1,j-1} + 4, v_{i+1,j} + 3, v_{i+1,j+1} + 4) \end{aligned}$$

Where v_{ij} is the value of the pixel in position *(i, j)*. As shown in Figure 2.1, the value in Figure 2.1(b) is the distance between two corresponding boundary pixels.

Computing the boundary value in the above way, the distance between the two objects can be obtained and the average distance value is used to evaluate the goodness of the match.



(a) Model boundary pixel



(b) Distance between the model and the test object boundary. Obtained by computing the Euclidean distance between the model boundary pixels and their nearest distances from the test object boundary pixels.

Figure 2.1. Compute the distance. The normalized boundary pixel is placed over the distance images. The average distance of the normalized boundary pixels on the distance image is used to measure the goodness of fit.

2.2.2 Curve and Line Segment Matching

Curve and line segment matching is also a common method for shape discrimination. Both methods are contour-based. For curve matching (Lamdan, Y. *et al.*, 1990) (Wolfson, H.J., 1990), the general approach is to first use a coordinate transformation to align the two curves and then compute the distance between the corresponding pixels. For line segment matching, a polygonal approximation is usually used to represent the object boundary, and the length and the angle of each line segment to the x or y axis are regarded as the features. The feature distance between the objects is used to evaluate the resemblance.

2.2.3 Deficiencies of Global Matching Method

Object overlapping often happens in industrial applications. For example, in a fish inspection conveyor, two fish maybe overlap under the digital camera, thus cause difficulty for recognition and failure of automatic grading. How to deal with such a problem is an important topic for pattern recognition and computer vision research.

A common deficiency of global shape descriptors under object occlusion lies in that distortion of a local area in the object will affect the overall shape representation as a whole. For example, in Fourier descriptors, all the elements of the object are involved in normalization and Fourier transformation. If a few elements are changed or lost, all of the final coefficients will be consequently changed. Obviously, such a global representation is not suitable for the occluded object recognition. Also, for the curve or line segment

matching, if some parts of the object are occluded, it is difficult to find the corresponding sub-curve or line segments, thus only calculating the distance between the curves or line segments cannot reflect the goodness of the matching between two objects. Under such a circumstance, the object matching needs a local shape description.

2.3 Local Shape Description

Local shape descriptors have been applied to the partially occluded object discrimination domain since early 1980s (Loncaric, S., 1998). Many early methods attempted to find some salient attributes of the object, such as holes, right angles, and protrusions, then search for common attributes between the sample and the model object. However, such methods are very primitive and limited. They fail when these attributes themselves are occluded, hence are discarded by more recent local shape descriptors.

Local shape descriptors are a set of representations extracted from the object region or boundary, and the representations are independent. Thus the distortion or lost of a few representations does not affect the rest ones. The development of the local shape descriptors is based on the global ones and many local shape descriptors borrow ideas from the global descriptors. Many global descriptors, after some modification, can be used as local shape descriptors, such as Fourier descriptors and 2-D moments (to be discussed later). Some popular local representations include auto-regression, B-Spline, dominant points, polygonal approximation, etc. Almost all of these descriptors are boundary-based. For discrimination, they can be classified into two fields: sub-curve isomorphism and sub-polygon isomorphism.

2.3.1 Sub-curve Isomorphism

Wolfson (Wolfson, H.J., 1990) proposed a curve representation algorithm to find the longest matching sub-curve that appears in the test and the reference curves. This algorithm does not depend on special features of the curves such as the critical points that require a polygonal approximation. Wolfson represented both curves by characteristic strings of the real numbers. He first built the *arc-length versus total turning angle* graph of the curve by accumulating the total turning angle of the curve at each vertex (pixel on the curve) and denoted it as $\phi(s)$. Then he sampled $\phi(s)$ at equally spaced points and computed the mean difference for each equally spaced segment. Finally, the longest matching sub-curve is determined by comparing the set of mean differences. This algorithm shares some ideas with many *auto-regression* methods (Dubois, S.R. and Glanz, F.H., 1986) (Kurita, T. *et al.*, 1994) used to describe complex and isolated objects with the *accumulated angle along the curve* as the features. Wolfson's experiment shows his algorithm achieves a good result for the isolated object recognition, and can deal with partially occluded objects. However, it also suffers shortcomings. First, the representation based on *arc-length versus total turning angle* is not unique. Secondly, how to deal with the sporadic short matching curves is not explored.

2.3.2 Sub-polygon Isomorphism

As stated above, many global representation methods can be applied to the local shape representation after some modification. Among them, sub-polygon isomorphism

(line segment matching) is a popular one. Line segments matching is based on the *piecewise linear approximation* of the object boundary, it is usually categorized into two approaches (Ansari, N. and Delp, E.J., 1991):

1) *Polygonal approximation* of the digital curve subject to certain constraints on the goodness of fit.

2) *Dominant points* extraction through angle or corner detection schemes.

Both the polygonal approximation and the dominant point extraction are popular representations for the global and local shape description and are discussed in detail in this chapter (dominant point extraction will be discussed in next chapter).

Polygonal approximation is to represent object boundary by a polygon, the line segments of the polygon are usually used as the shape features. A common characteristic shared by most polygonal approximation methods is the *collinearity* test. Collinearity is usually determined by the maximum perpendicular distance from a point on the boundary portion to the straight line connecting the two terminal points of that portion. The basic idea of many current advanced polygonal approximation algorithms is offered by Ramer and Pavlidis (Ramer, U., 1972) (Pavlidis, T. and Horowitz, S.L., 1974). Ramer suggested the top left most point and the bottom right most points of the boundary be used as the two initial break points, and sought the new break points by using the *collinearity* criterion repeatedly. Pavdilis and Horowitz approximated boundary points by interpolating straight line segments. The procedure was first to assign an arbitrary number of points along the boundary as the initial set break points, then to combine the *collinearity criterion* and a *split-and-merge* process to achieve the final set of break

points. However, these two methods suffer from the same problem: the break points generated are not stable under rotation, scaling and shifting. Based on these two methods, Ansari and Delp (Ansari, N. and Delp, E., 1991) proposed a more robust algorithm: *curvature guided polygonal approximation*. This method first uses the *Gaussian function* to smooth the boundary curve to remove the false local concavities and convexities, then computes the curvature for every point along the boundary. The points with the positive maximum and negative minimum curvature are selected as the initial break points. Finally, a set of break points are detected by Pavlidis' algorithm. Satisfying results are achieved by this method.

The above *piecewise linear approximation* methods are frequently applied in the occluded object recognition because it is easy to extract local features from the polygonal segments, such as segment length, orientation, length ratio or interior angle between the adjacent line segments. These features are appropriate for matching after affine transformation. Such methods are used in (Ansari, N. and Delp, E.J., 1990) (Ayache, N. and Faugeras, O.D., 1986) (Gorman, J.W. and Mitchell, O.R., 1988) (Han, M.H. and Jang, D.S., 1988) (Lamdan, Y. *et al.*, 1990) (Liu, H.C. and Srinath, M.D, 1990) (Loch, M.W. and Kashyap, R.L., 1987) (Tsang, P.W. *et al.*, 1994) (Wen, W. and Lozzi, A., 1992) to discriminate the partially occluded objects.

2.3.3 B-Spline, Auto-regression and Other Local Descriptors

Besides the sub-curve and sub-polygon isomorphism, some other global shape descriptors can be used in the local shape description after modification. Such as B-Spline, auto-regression, Fourier descriptors and hidden Markov model.

B-Spline (Salari, E. and Balaji, S., 1991) is a method derived from ship and aircraft design, it is attractive because it exhibits local control and can be used in the presence of incomplete boundary information. *B-Spline* uses the piecewise polynomial curves that are guided by a set of control points (CP) combined with blending functions. The tangents and curvatures at the CPs are viewed as local features. This method is effective for some occluded object discrimination, but it is computationally extensive since for every CP the *B-Spline* equation, which is formulated in a matrix form, has to be solved.

Gorman (Gorman, J.W. *et al.*, 1988) uses *Fourier descriptors* to describe a set of polygonal segments represented by chain codes. Since the Fourier representation requires that the code be a close or periodical form, the chain codes are formed by searching from the first break point, travelling through the line to the second and third break points, then searching through the old route back to the first break point. When the method works for partially occluded object recognition, every line segment needs to be represented by a set of Fourier coefficients, it is very computationally extensive if the number of the break points increases.

Auto-regression (AR) (Dubois, S.R. and Glanz, F., 1986) is a popular method for the local shape representation. The AR parameters as a feature vector can be used to characterize shapes in a very systematic manner. The major disadvantage of AR is that it is very sensitive to the shape occlusion since AR models the whole shape with only one set of predictive parameters. If the shape contains a large number of sample points and the contour varies radically, the shape becomes unpredictable. However, *hidden Markov model* (HMD) (Aas, K. *et al.*, 1999) (He, Y. and Kundu, A., 1991) does not model the whole shape as a single feature vector, it explores the relationship between the consecutive segments of the shape to be classified in a probabilistic domain. The combination of AR and HMD was developed to discriminate different maps and achieve a good result. But at the same time, this algorithm is complicated and very computationally extensive.

2.4 Local Matching Problem

A difficult problem often encountered in the occluded object recognition is matching the local representations after the feature extraction. In isolated object discrimination, the feature matching between the test and the reference object is simple and straightforward. For the heuristic representation, the usual approach is to normalize the number of heuristic items to the same and compute the Euclidean distance between the corresponding features (Tai, Y. et al., 1996) (Zahn, C.T. and Roskies, R.Z., 1972). For the line or curve segment representation, the usual method is first to apply the

coordinate transformation to align the segments, then match the corresponding segments by comparing their length and orientation, or just calculating the coordinate difference between each other. Another effective approach is to apply a statistical (Tao, Y. *et al.*, 1996), fuzzy (Hu, B.G. *et al.*, 1998) (Liu, X.F. *et al.*, 1994) or neural classifier (Mitzias, D.A. and Mertzios, B.G., 1994) to discriminate the object shapes based on the shape feature vectors.

However, for local shape discrimination, though some ideas are borrowed from the global methods, such straightforward algorithms do not function well under occlusion. The problem lies in the criterion for evaluation of the resemblance between two objects. A uniform standard such as the total square sum of the distance or the average distance between two comparing objects may be regarded as the criterion. But under occlusion, some unmatched representations between the model and the scene object may be almost totally different, such a uniform standard has no meaning for the discrimination. Borgefors, Liu and Tsang (Borgefors, G., 1988), (Liu, H.C. and Srinath, M.D, 1990) (Tsang, P.W.M, et al., 1994) all used Chamfer ¾ distance transformation and a partial distance measurement scheme to perform the local matching for the occluded object discrimination. Liu and Srinath (Liu, H.C. and Srinath, M.D, 1990) first extracted the dominant points of the object boundary, then approximated the boundary by a set of line segments. They designed a two-step segment matching approach. The first step was the segment matching based on *length ratio* and *interior angle* between the adjacent segments of the reference and the test object. The second step used a group matching to remove the segments for which fewer than three consecutive segments in one shape

match consecutive segments in the other shape. They then normalized the unknown shape by the scaling, rotation and shifting parameters. The evaluation of similarity was based on the distance transformation and partial distance measurement. The goodness of the match was computed from the average of the boundary distance computed by the Chamfer ¾ Distance Transformation. To deal with the occlusion, the boundary was broken into N equal length segments with an arbitrary starting, the local distance between each segment and the test object is defined from the boundary distance:

$$\frac{1}{3}[\frac{1}{n}\sum_{i=1}^{n} v_i^2]^{1/2} \qquad \text{(Equation 2.10)}$$

Where n is the number of boundary points in the segment and v_i is their superimposed distance value. Then the minimum distance between the $N/2$ consecutive segments is regarded as the partial distance between two shapes. The simulation results show the algorithm is satisfactory for discriminating map and airplane images. But from a "global" view, this method is not good enough because the relationships between the remote points on the model and the test boundary are not explored. That is also the common deficiency for many local comparison algorithms. Tsang (Tsang, P.W.M. *et al.*, 1994) almost shares the same idea as Borgefors (Borgefors, G., 1988) in the local comparison and suffers the same shortcoming.

Dynamic programming (Appendix II) is another popular algorithm applied in the occlude object recognition (Ansari, N. and Delp, E.J., 1990) (Gorman, J.W. *et al.*, 1988). A fundamental property of dynamic programming is the "*principle of optimality*" which implies that the minimum or maximum distance path found by the dynamic programming

will be of the minimum or maximum distance in the global sense. Ansari and Delp first extracted the dominant points from the smoothed object boundary, then arranged the dominant points in an ordered form. The feature of every dominant point is achieved through the *sphericity* (Appendix III) (Ansari, N. and Delp, E. J., 1990) of the ordered triplets. Sphericity is also viewed as a kind of local property. Dominant triplets (three consecutive dominant points) always form a triangle in which an ellipse can be inscribed. The *sphericity* is defined as the ratio of the *geometric mean* to the *arithmetic mean* of the lengths of the principal axis of the inscribed ellipse. An entry support table is built to denote the resemblance between the reference and the test object. The row index indicates the dominant point number of the reference object, while the column index corresponds to those of the test object. *Hopping dynamic programming* (HDP) is used by Ansari and Delp (Ansari, N. and Delp, E.J., 1990) to retrieve the maximum value path penetrating through the entry support table. Unlike Gorman's method (Gorman, J.W. *et al.*, 1988), both forward and backward transition rules are applied. The final maximum value path corresponds to the best match of the entry index, and HDP is used as an optimization approach. However, HDP cannot always guarantee the "global optimization". Like the Hopfield neural network coping with the *Travelling Salesperson Problem* (See Appendix IV), HDP suffers the same "local minima" problem.

Wen and Lozzi (Wei, W. and Lozzi, A., 1992) presented a sub-polygon matching algorithm for the partially occluded object recognition. They first approximated the object boundary by a polygon, then divided the polygon into a set of sub-polygons. Each sub-polygon was then represented by the 2-D moments. The moment distance between

the sub-polygons within the "mother" polygon and the moment distance of each sub-polygon in the reference object with all the sub-polygons in the test and the reference object are all explored. Finally all the parameters are used for matching. Since in this algorithm, the relations between remote segments of both the model and the test object are explored from a "global view", the matching may be viewed as "global matching". Actually, the idea is also reflected in this thesis by use of a modified Hopfield neural network.

In general, most algorithms applied up to now for the local matching are not mature and they are still based on the "local distance measure". Hopping dynamic programming and Hopfield networks can deal with such issues because both methods cope with global optimization problems. While the physical meaning for optimization is quite clear, they suffer from the "local minima" problem.

2.5 Hopfield Neural Network

2.5.1 Neural Network Application In General

Since middle 1980s, neural networks (Schalkoff, R., 1992) (Taoukalas, L.H. and Uhrig, R.E, 1997) have been applied to shape classification (Gupta, L and Sayeh, M.R., 1988) (Neocognitron, K.F, 1988) (Tsang, P.W.M. *et al.*, 1992). Unlike the traditional statistical and syntactical classifiers which process data sequentially, neural network classifiers compute data in parallel to provide a high computation rate. Moreover, neural network classifiers can learn in the training process and they are robust because the loss

or distortion of a few computing units does not significantly affect the overall performance. The most popular network used in shape discrimination is the *backpropagation* (BP) network. The BP network is also widely used in digital signal processing, character and speech recognition and it comprises 70% of all the neural network applications (Tsoukalas, L.H. *et al.*, 1997).

Gupta (Gupta, L. *et al.*, 1989) adopted a *three layer perceptron* (backpropagation) network to classify a set of closed planar shapes. The input data was the one dimensional radius boundary signature after normalization. The result shows that this BP classifier is more robust and powerful compared with conventional classifiers under the noise environment.

However, for the local shape classification, the BP network is not useful since it was not designed for optimization. No global and local concepts are involved in this network and it uses a set of feature vectors for training and testing. The first paper exploring Hopfield neural network (HNN) for solving partially occluded object discrimination was proposed by Nasrabadi and Li in 1991 (Nasrabadi, N.M. and Li, W., 1991). The HNN used by Nasrabadi and Li is actually a modified version compared with the standard Hopfield network. In the following several years, many papers (Ansari, N. and Li, K.W., 1992) (Chung, P.C. *et al.*, 1994) (Kim, J.H. *et al.*, 1996) (Lee, J.S. *et al.*, 1996) improved the algorithm based on Nasrabadi and Li's work.

2.5.2 Hopfield Neural Network

2.5.2.1 Concept of Hopfield Neural Network

The Hopfield neural network may be viewed as a nonlinear associative memory or content–addressable memory (CAM) (Haykin, S., 1994) (Schalkoff, R.J., 1992) that was presented first by Hopfield in 1982 (Hopfield, J.J., 1982). The primary function of this network is to retrieve a pattern stored in memory in case a noisy or incomplete version presented. An important property of a CAM is to retrieve a stored pattern provided with a reasonable subset of the information content of that pattern. It is robust and error-correcting because it can recover the inconsistent pattern information. To perform the retrieving task, a set of stable patterns must first be putted into the network which are regarded as fundamental memories. If the noise or inconsistent pattern is close to the stable memory, the system should evolve with time and finally converge to the fundamental memory. Therefore, the Hopfield network may be described as a dynamically convergent system.

Neurons are the basic computing units in the Hopfield model, they have two states: on and off, which can be denoted by $s_i=+1$ *or* $s_i=-1$. For a network made up of N neurons, the state of the network is defined by the vector:

$$s=[s_1,s_2,\cdots,s_N]^T \qquad \text{(Equation 2.11)}$$

The diagram of the Hopfield network structure is shown in Figure 2.2

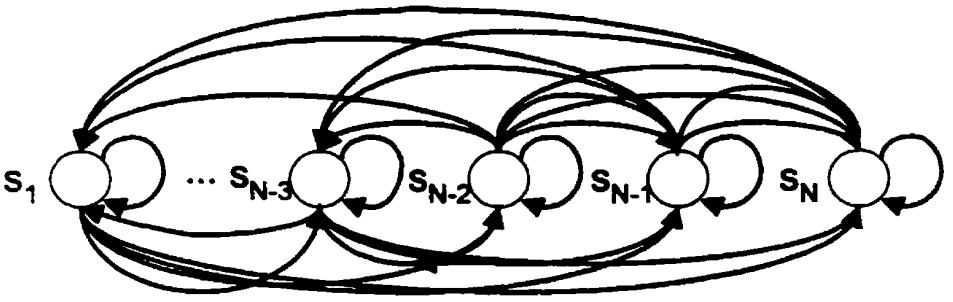


Figure 2.2 The diagram of Hopfield neural network

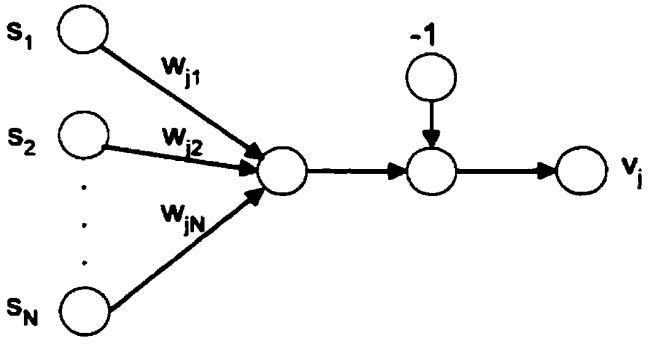


Figure 2.3 Signal-flow graph of the net activation potential v_j of neuron j

As shown in Figure 2.2, neurons are connected with each other by synaptic weights. For example, w_{ji} is used to denote the synaptic weight between neuron i and j. Since the rest of the neurons are coupled with neuron j by the weights, as shown in the signal-flow graph of Figure 2.3 (Haykin, S., 1994), the contribution of all the net neurons acting on neuron j may be denoted in a following summation form:

$$v_j = \sum_{i=1}^{N} w_{ji} s_i - \theta_j \qquad \text{(Equation 2.12)}$$

where θ_j is a fixed threshold applied externally to neuron j. Neuron j changes its state according to the following deterministic rule:

$$s_j = \begin{cases} +1 & if \quad v_j > 0 \\ -1 & if \quad v_j < 0 \end{cases} \qquad \text{(Equation 2.13)}$$

2.5.2.2 Hopfield Network Operation Procedure

Haykin (Haykin, S., 1994) concludes the operation procedure for the Hopfield network may be summarized as follows:

1) Storage (learning). Let ε_μ denote a known set of N-dimensional memories. Construct the network by using the outer product rule (Hebb's postulate of learning) to compute the symmetric synaptic weights of the network as

$$w_{ji} = \begin{cases} \frac{1}{N} \sum_{\mu=1}^{p} \xi_{\mu,j} \xi_{\mu,i}, & j \neq i \\ 0, & j = i \end{cases} \qquad \text{(Equation 2.14)}$$

where w_{ji} is the weight from neuron i to neuron j. The elements of the vector ε_μ equal $+1$ and -1. Once they are computed, the synaptic weights are fixed.

2) Initialization. Let x denote an unknown N-dimensional input vector presented to the network. The algorithm is initialized by setting

$$s_j(0) = x_j, \qquad j = 1, \cdots, N \qquad \text{(Equation 2.15)}$$

where $s_j(0)$ is the state of neuron j at time $n=0$, and x_j is the jth element of the vector x.

3) Iterate on until convergence. Update the elements of state vector $s(n)$ asynchronously (i.e., randomly and one at a time) according to the rule:

$$s_j(n+1) = \begin{cases} +1, & \sum_{i=1}^{N} w_{ji} s_i(n) > 0 \\ -1, & \sum_{i=1}^{N} w_{ji} s_i(n) < 0 \end{cases} \qquad \text{(Equation 2.16)}$$

Repeat the iteration until the state vectors remain unchanged.

4) Outputting. Let s_{final} denote the stable state computed at the end of step 3. The resulting output vector y of the network is

$$y = s_{final} \qquad \text{(Equation 2.17)}$$

2.5.2.3 Convergence Discussion

Since the property of the Hopfield network as a dynamically convergent system is the key to the retrieving of the incomplete or noise version pattern, it functions well only when the network is convergent. Consider a Hopfield network with symmetric synaptic weights $w_{ji}=w_{ij}$ $(w_{ii}=0)$, let s_i denote the state of neuron i, where $i=1,2,....,N$. The Liapunov (energy) function of the Hopfield network considered here is proved by Haykin

(Haykin, S., 1994) to be a convergent system. For example, the energy function is described as:

$$E = -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} w_{ji}s_i s_j \quad (i \neq j), \quad when \quad \theta_j = 0 \qquad \text{(Equation 2.18)}$$

The energy change ΔE due to the change ΔS_j in the state of neuron j is denoted as:

$$\Delta E = -\Delta s_j \sum_{\substack{i=1 \\ i \neq j}}^{N} w_{ji}s_i \qquad \text{(Equation 2.19)}$$

Since the network is binary, when the state of neuron j changes

$$\left\{ \begin{array}{l} when \quad S_j = +1 \rightarrow S_j = -1, \quad \Delta S_j < 0, \quad \sum_{\substack{i=1 \\ i \neq j}}^{N} w_{ji}s_i = S_j = -1 < 0 \\ when \quad S_j = -1 \rightarrow S_j = +1, \quad \Delta S_j > 0, \quad \sum_{\substack{i=1 \\ i \neq j}}^{N} w_{ji}s_i = S_j = +1 > 0 \end{array} \right\} \qquad \text{(Equation 2.20)}$$

Thus the change of the state of the neuron always causes the energy function E to be a monotonically decreasing function. State changes will continue until a local minimum of the energy landscape is reached.

Like other network structures, Hopfield network suffers from the "local minima" problem, but it does work under most applications. In this thesis, some algorithms dealing with occluded object discrimination using Hopfield network are implemented and modified to fit the specific problem. The network is more complicated than the standard Hopfield network since it involves with a 2-D matrix, so the network applied in this thesis is actually a modified one, it is used to deal with the global optimization of the matching.

Chapter 3

Image Processing

Image processing is an important part of a computer vision system. Succeeding the image acquisition, the functions of image processing generally include visual enhancement and feature presentation. Since the object features, such as edge or color, are regarded as the input information for the later processing, it is important that these features can accurately represent the original object. For example, in a computer integrated manufacturing system, robotic hands may handle the industrial objects in the conveyor by its 2-D or 3-D contour generated by the image processing operations. If the high precision is required, the contour should match the original object very accurately. Hence, the selection of appropriate image operations is important for the feature representation of the objects.

In the following discrimination algorithm, the dominant points are extracted from the object boundary, so the purpose for the image processing operations is to achieve an accurate object boundary from the scene. The cascading operations performed on the input images are Laplacian operation, image binarization, morphological operation and edge detection. The flow chart for the image processing algorithm is shown as Figure 3.1.

3.1 Laplacian Operation

Laplacian operator (Russ, J.C., 1995) is a local or neighborhood operator, which increases the local contrast of the image. In practical applications, sometimes the objects cannot be directly extracted from the background due to no crisp gray-level differences between the object boundary and the background. In image processing, a low-pass filter is used to sharpen the image by locally increasing the contrast at discontinuities and making the edge easy to see. For this purpose, Laplacian operator was first applied. Experimentation showed that an object thresholded after a Laplacian operation was smoother than directly applied a thresholding on it. While there are many different approaches for the image enhancement, what is applied here is a 3×3 kernel written as an array of integers as following:

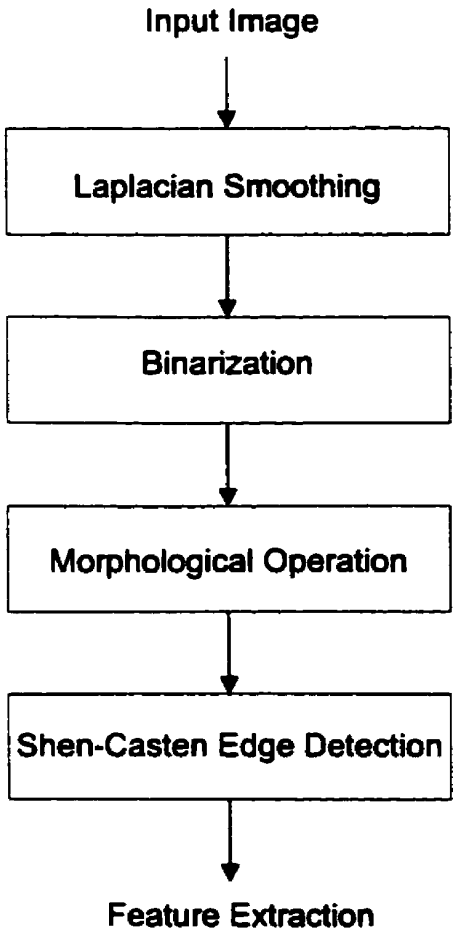


Figure 3.1 Image operations flow chart

1	2	1
2	4	2
1	2	1

Table 3.1 A standard Laplacian kernel

This kernel is convolved with the whole image as follows: First a pixel in the image is regarded as the central pixel. This central pixel and its eight neighbors form a mask and their gray-level values are multiplied with the corresponding values of the Laplacian mask and the summation is obtained. The new gray-level value of this central pixel is then obtained by dividing this summation by 16 (the sum of the 9 weights of the Laplician mask). This process is performed from the top-left to the bottom-right of the image.

3.2 Morphological operation

The binary image is achieved by thresholding the Laplacian filtered image. However, sometimes the complete binary object in the scene can not always be achieved by simply applying a uniform gray-level threshold (see Figure 3.2 (c)), morphological operation is a good approach to repair them. Morphological operations are actually the most extensive operations for binary image processing. Erosion, dilation and their cascaded combinations are frequently used in such applications.

Binary erosion and dilation are logical decision-making operations (Burdick, H.E., 1997). Like the convolution, binary erosion and dilation combine the local neighboring pixels with a mask to achieve a result. For simplicity, the pixels in the binary image are denoted as 1 and 0. Binary erosion uses a 3×3 mask as following:

1	1	1
1	1	1
1	1	1

Table 3.2 The mask used for erosion

The pixels in the mask have an "AND" relationship, that means every pixel in the mask must be 1 in order for the output of the central pixel to be 1, otherwise the central pixel will be 0. It can be imagined that the effect of erosion on a binary image is to diminish the edge of a white area of pixels and can be used to separate edge-overlapped objects. On the contrary, the effect of dilation on a binary image is to increase the black edge area, it has a following convolution mask:

0	0	0
0	0	0
0	0	0

Table 3.3 The mask used for dilation

The logical relationship between the neighborhoods is "OR", which means only one pixel being 1 can result in the output of the central pixel to be 1. Dilation is the opposite operation of erosion and can be used to repair some "cracks" in an object to achieve a complete one.

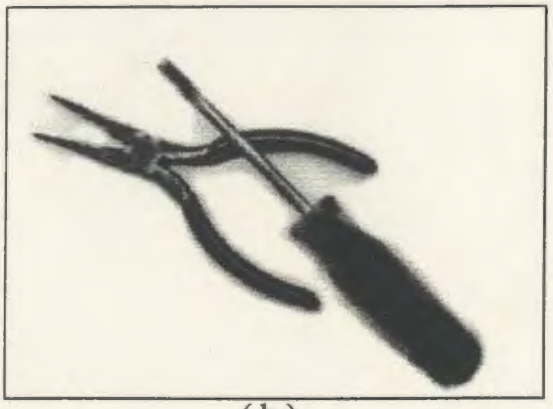
The combination of an erosion followed by a dilation is called an opening, while a dilation followed by a erosion is called a closing (Bassmann, H. and Basslich, P.W., 1992). Such morphological operations are used to separate or repair "cracks" in objects in the scene. In the proposed experiment, closing was applied to the binary image because after threshold filtering, some parts within the object are missing due to uneven gray-level of the object.

3.3 Boundary Detection

The Shen-Castan algorithm (Parker, J.R, 1997) was used for boundary extraction. Compared with the template-based edge detection methods such as the Sobel edge detector, Kirsch edge detector, Marr-Hildreth edge detector (Parker, J.R, 1997) (Russ, J.C., 1995) and Canny edge detector (Canny, J.F., 1986), the Shen-Castan detector is regarded as superior (Parker, J.R., 1997). The comparison was achieved by Parker through testing several typical square template images under different signal-to-noise ratio (SNR). Shen-Castan and Canny edge detector are now the most popular edge detectors used in image processing domain, they are insensitive to noise and the optimal result may be achieved via adjusting the involved parameters.

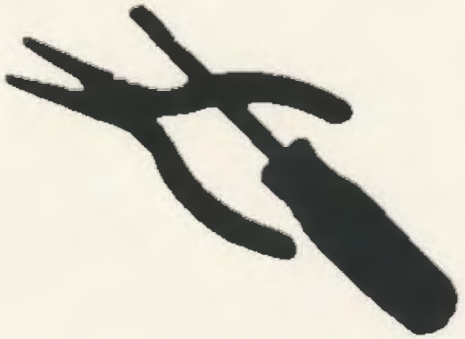
More than 100 images were tested in the experiment by the Shen-Castan edge detector and several examples of the extracted boundaries are shown in Figure 3.2-3.4. By adjusting the thinning and smoothing parameters, the complete boundaries could be achieved even with oblique portions.

(a)

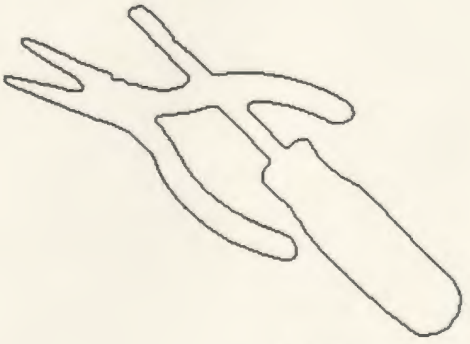


(b)

(c)



(d)



(e)

Figure 3.2 (a) Original overlapped image (b) After Laplacian smoothing
(c) After binarization (d) After morphological operation (e) After edge detection

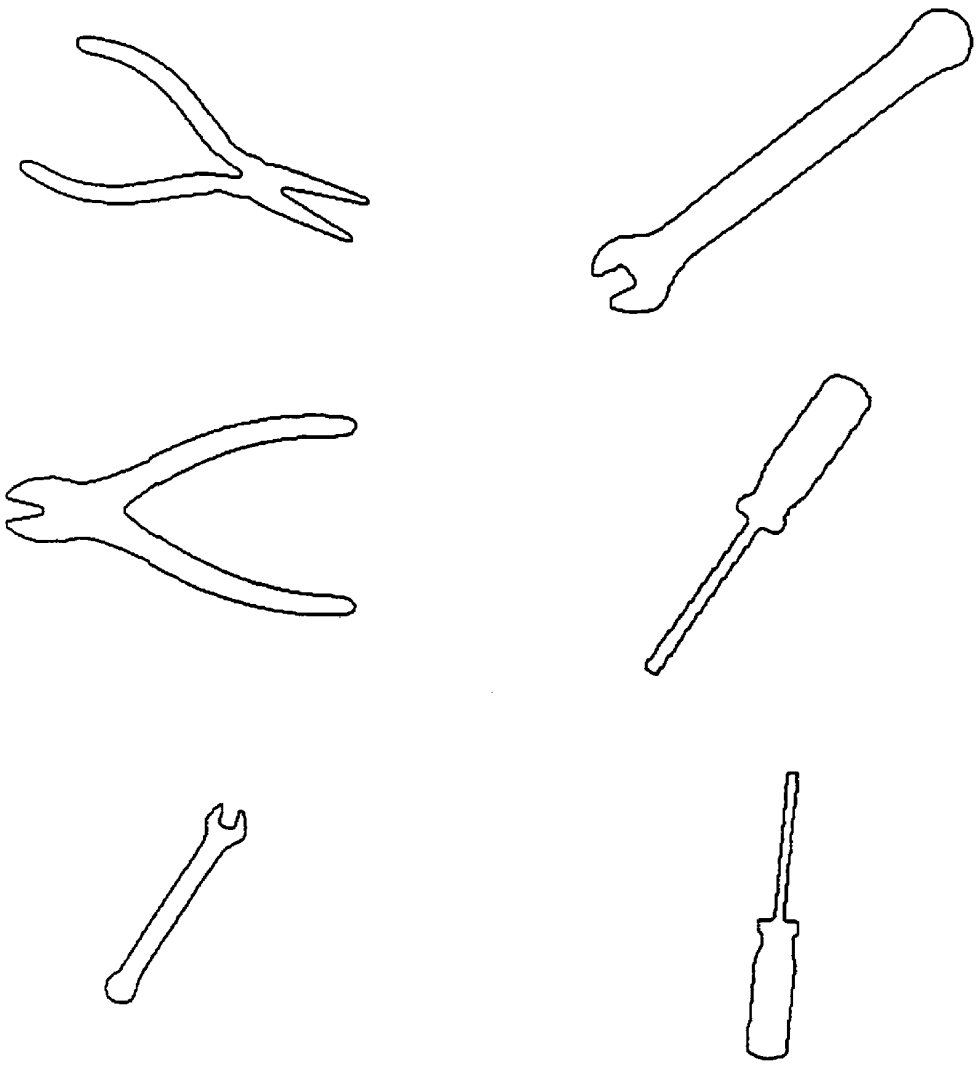


Figure 3.3 Different model hand-tool images after boundary detection

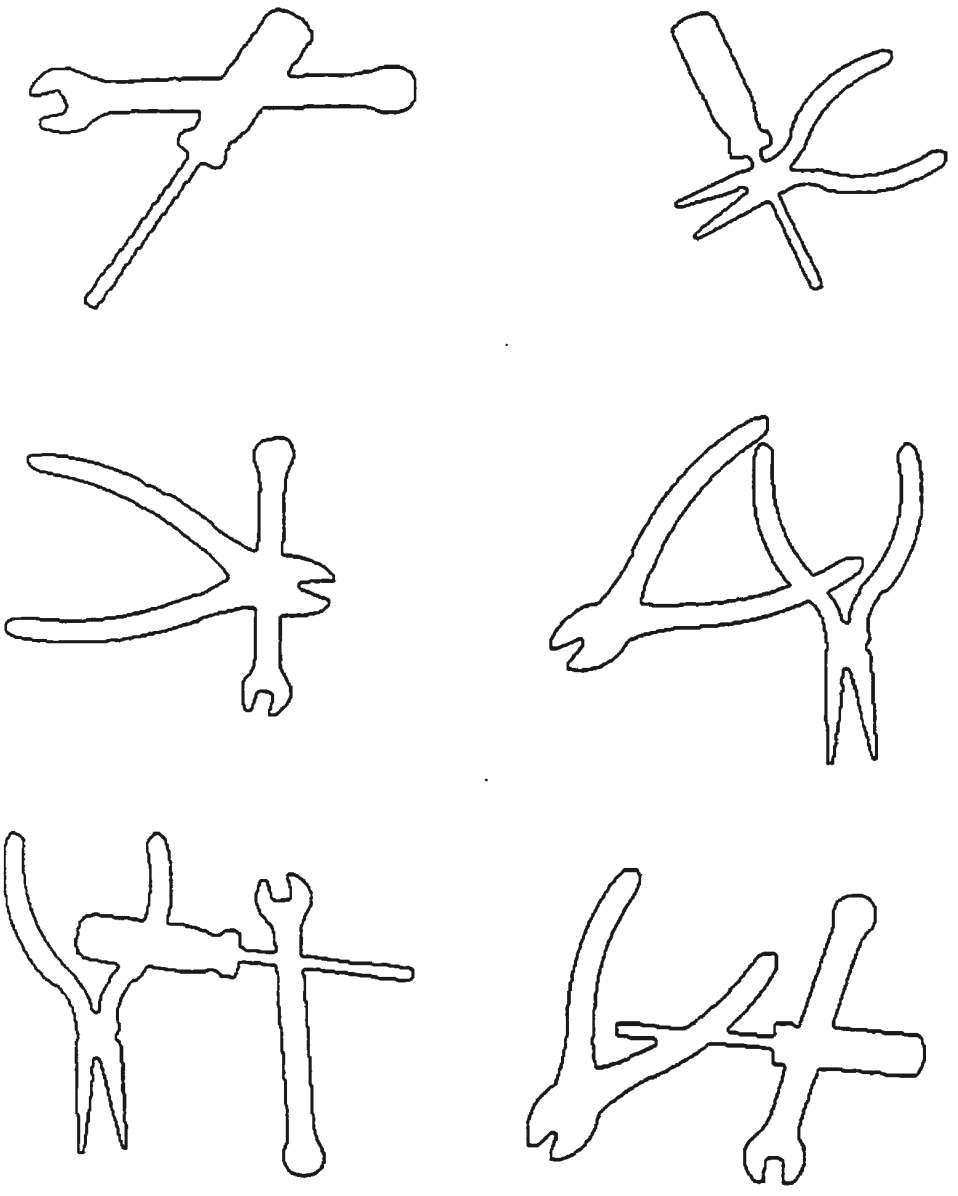


Figure 3.4 Overlapped hand-tool images after boundary detection

Chapter 4

Shape Discrimination Method

As mentioned in Chapter 2, the first step to object discrimination should be based on the feature extraction. For the proposed algorithm, *dominant points* on the object boundary are regarded as the shape features. The advantage for choosing dominant points rather than polygons with Ramer and Pavlidis' method as shape features has been discussed in Chapter 2. Simply speaking, dominant points are more stable under image rotation, scaling and shifting (Ansari, N. and Delp, E.J., 1991). The dominant points are obtained by searching the pixels along the boundary whose curvature values are local extreme (maximum or minimum). The initial comparison between the test and the model object is performed by building a *disparity matrix* whose

elements denote the measure between the local properties of the dominant points of the test and the model object. However, as discussed earlier, a matching only based on the local features is not reliable. Therefore, a modified Hopfield neural network is implemented to perform the global optimization based on the initial local comparison. The algorithm flow chart is shown in Figure 4.1.

4.1 Boundary Pixel Linking

Boundary pixel linking is the first step toward indexing of the dominant points. Because the detection of dominant points is achieved by searching for the local extreme curvatures along the object boundary, the coordinates and curvature value of each point must be indexed from its identity (sequence along the boundary). There should exist a correspondence between the pixel's sequence, its curvature value and coordinates. Such a mapping can be obtained by building a linked list of all of the boundary pixels. In this linked list exists a correspondence of the coordinates, sequence and curvature value for each boundary pixel. Therefore, only by indexing the identities of the dominant points, their corresponding coordinates and curvature values can be retrieved.

The linking algorithm is a Freeman chain coding method (Freeman, H., 1961). It is performed by first choosing a boundary pixel, then searching its neighborhood along the boundary in a counterclockwise way as illustrated in Figure 4.2.

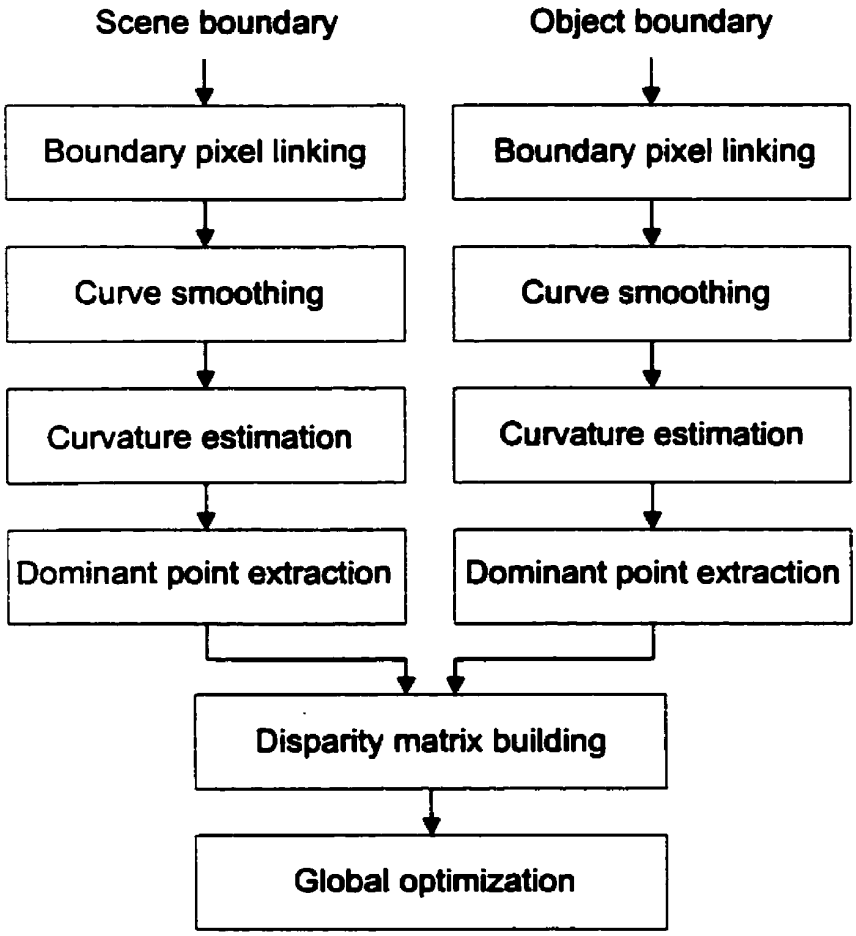


Figure 4.1 Algorithm flow chart for object discrimination

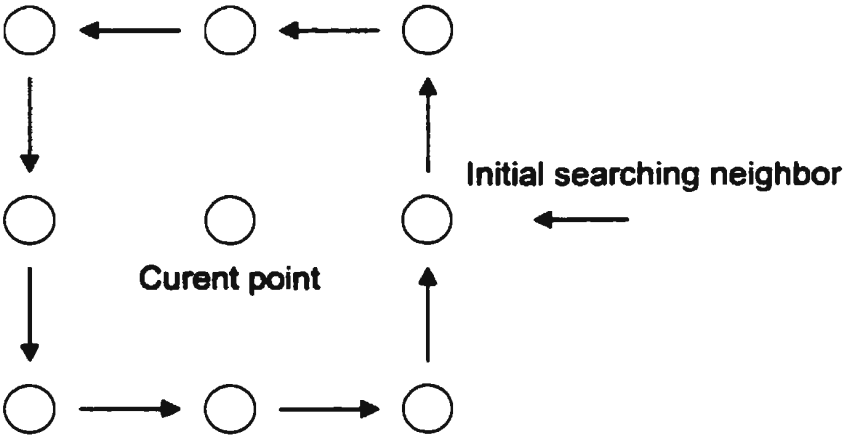


Figure 4.2 Boundary linking algorithm by a 8-neighbor Freeman coding

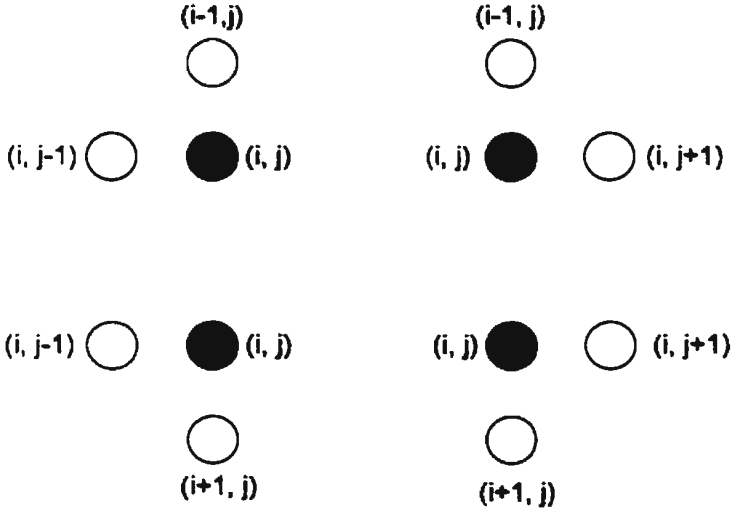


Figure 4.3 Four cases of spurious pixels (spurious pixels are marked as shadow)

As shown in Figure 4.2, a pixel and its 8 neighbors form a 3×3 mask. The central pixel in the mask is always regarded as the "current pixel" along the boundary and the searching is performed from its initial searching neighbor. Once a neighbor is identified as a boundary pixel, the current pixel is marked as "found" and its properties (i.e., its coordinates and sequence number along the boundary) are recorded in a linked list. The identified neighbor takes the "current pixel" position and the searching continues until all the pixels along the boundary are marked as "found".

It can be seen that to achieve an acceptable linked list, the boundary should be exactly one-pixel wide, otherwise, the searching will not function. However, to date no edge detectors, including the Shen-Casten (Parker, J.R., 1997) and Canny edge detector (Canny, J.F., 1986), can achieve such a perfect boundary for an arbitrary object. There are always some spurious pixels along the boundary after the boundary thinning which may locate as shown in Figure 4.3.

Under such circumstances, the spurious pixels have to be removed before linking. Before the linking, the boundary pixels are marked as 0 and the non-boundary pixels are marked as 255. Then the spurious pixel removing can be performed using the following algorithm:

```
if [i, j] = 0
    if [i, j-1] = 0
        if [i-1, j] = 0 or [i+1, j] = 0
            remove [i, j]; [i, j] = 255
    elseif [i, j+1] = 0
        if [i-1, j] = 0 or [i+1, j] = 0
            remove [i, j]; [i, j] = 255
```

The choice of the initial searching pixel does not matter since all of the pixel's sequences on the boundary are relative. So just by scanning, the first boundary pixel encountered may be regarded as the initial searching pixel.

4.2 Dominant Point Extraction

When objects are occluded, shape discrimination methods using global features usually fail due to the lack of reliable information. Hence, to discriminate the occluded object, the feature set should be robust enough so that when some features are missing, the rest can still provide some discrimination information. In the proposed research, a set of dominant points are extracted as the object shape features. The physiological research on the human visual system has shown that some dominant points of the boundary are rich in information content and are sufficient to characterize the shape of the object (Attneave, F., 1954). Here the dominant points are extracted as the points which possess the highest curvature magnitudes among a local boundary portion. As discussed in Chapter 2, there are many approaches for dominant point detection. Polygonal approximation is the most common approach for solving such a problem

(Pavlidis, R. and Horowitz, S.L., 1974) (Ramer, U., 1972). However, the extraction of the dominant points only based on the polygonal approximation is not stable and robust since a unique representation usually cannot be achieved under rotation, shifting and scaling for the same object. As a result, methods combining curvature estimation and polygonal approximation were developed. *Curvature guided polygonal approximation* and *cardinal curvature points method* had been discussed in detail by Ansari and Delp (Ansari, N. and Delp, E.J., 1991) which achieved a robust and stable representation under rotation, scaling and shifting of the object.

Since 1993, multiresolution wavelet transform (Lee, J.S. *et al.*, 1993) (Yoon, S.H. *et al.*, 1998) has been applied to detect the dominant points. Wavelet transform combines the multiresolution approximation and curvature scale-space representation to detect the dominant points. If a point can be detected under different scale parameters, it will be regarded as a dominant point. This method, however, involves extensive computation.

Chung (Chung, P.C., *et al.*, 1994) presented a competitive Hopfield neural network for polygonal approximation which can be regarded as a method for the dominant point extraction. Based on this parallel algorithm, the extraction is then performed as a minimization of a criterion function defined as the arc-to-chord deviation between the curve and the imposed polygon. This method is novel for parallel implementation in shape representation and therefore offers the potential of real-time computation. But it obtains valid results only when the network convergence is

guaranteed due to the "*local minima*" problem, and it is not rotation, shifting and scaling independent.

The dominant point extraction in the proposed algorithm is using the curvature estimation. The points with local extreme curvature values are regarded as the dominant point candidates, and the final dominant points are chosen from such candidates by a magnitude thresholding.

4.2.1 Curvature Estimation

The *curvature* is generally defined as the *derivative of the tangent angle to the curve* (Asada, H and Brady, M., 1986). As described earlier, after the boundary pixel linking, the pixels along the boundary are put into a linked list in which the sequence and coordinates of each pixel can be indexed. To achieve such a data structure, the boundary pixels are expressed in a parameterized way. If the indexed variable of the boundary pixels is t, in the Cartesian coordinate system, the boundary can be described by the following function:

$$C = \{x(t), y(t)\} \qquad \text{(Equation 4.1)}$$

and the curvature function can be expressed as:

$$k(t) = \frac{\dot{x}(t)\ddot{y}(t) - \ddot{x}(t)\dot{y}(t)}{[\dot{x}(t)^2 + \dot{y}(t)^2]^{3/2}} \qquad \text{(Equation 4.2)}$$

Where $t = 0,1,2,\cdots N, N$ *is the boundary pixel number*

However, if the curvature estimation is directly performed on the original boundary, the points detected usually cannot optimally represent the object shape because some local concavities and convexities may be introduced during the boundary extraction. Therefore, to optimally represent the object shape based on the dominant points, the boundary curve smoothing is necessary for removing the local concavities and convexities. A Gaussian filter (Kim, J.H, *et al.*, 1996) (Mokhtarian, F. and Mackworth, A., 1986) (Nasari, N. and Li, K.W., 1993) (Tsang, P.W.M. *et al.*, 1992), which has been shown to be an ideal smoothing filter, is used here. The extreme (positive maximum and negative minimum) curvature points along the boundary smoothed by the Gaussian filter with an approximate smoothing width are stable with respect to rotation, shifting and scaling (Mokhtarian, F. and Mackworth, A., 1986). Consequently, provided the smoothing width being correctly chosen, the dominant points on the smoothed curve will be the most salient points to represent the object shape.

4.2.2 Gaussian Smoothing

The Gaussian smoothing filter ((Mokhtarian, F. and Mackworth, 1986) is a low-pass filter which can be described as following:

$$g(t,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{t^2}{2\sigma^2}} \qquad \text{(Equation 4.3)}$$

Where t is the indexed sequence of the boundary pixels, $t=0,1,2,\ldots,N$. σ is the smoothing width. The selection of σ is very important to achieve a satisfying smoothed curve. If σ is too big, some salient points may be removed and the feature information will be lost. On the contrary, when σ is too small, some small local perturbations cannot be removed thus spurious features may remain.

The smoothed curvature function is obtained by convolving the original curvature function with the Gaussian function and may be denoted as:

$$k(t,\sigma) = k(t) \otimes g(t,\sigma)$$

$$k(t,\sigma) = \frac{\dot{x}(t,\sigma)\,\ddot{y}(t,\sigma) - \ddot{x}(t,\sigma)\,\dot{y}(t,\sigma)}{[\dot{x}(t,\sigma)^2 + \dot{y}(t,\sigma)^2]^{3/2}} \qquad \text{(Equation 4.4)}$$

Where $t = 0,1,2,\cdots N, N$ *is the boundary pixel number*

The convolution forms of $x(t)$, $y(t)$ and their corresponding one-order and two-order derivatives are described as (Mokhtarian, F. and Mackworth, A., 1986):

$$\begin{aligned} x(t,\sigma) &= x(t) \otimes g(t,\sigma), & y(t,\sigma) &= y(t) \otimes g(t,\sigma) \\ \dot{x}(t,\sigma) &= x(t) \otimes \dot{g}(t,\sigma), & \dot{y}(t,\sigma) &= y(t) \otimes \dot{g}(t,\sigma) \\ \ddot{x}(t,\sigma) &= x(t) \otimes \ddot{g}(t,\sigma), & \ddot{y}(t,\sigma) &= y(t) \otimes \ddot{g}(t,\sigma) \end{aligned} \qquad \text{(Equation 4.5)}$$

Since

$$\dot{g}(t,\sigma) = \frac{\partial}{\partial t} g(t,\sigma) = \frac{\partial}{\partial t}\left(\frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-t^2}{2\sigma^2}}\right) = -\frac{t}{\sigma^3\sqrt{2\pi}} e^{\frac{-t^2}{2\sigma^2}}$$

$$\ddot{g}(t,\sigma) = \frac{\partial^2}{\partial t} g(t,\sigma) = \frac{\partial}{\partial t}\left(-\frac{t}{\sigma^3\sqrt{2\pi}} e^{\frac{-t^2}{2\sigma^2}}\right) = \frac{1}{\sqrt{2\pi}}\left(\frac{t^2}{\sigma^5} - \frac{1}{\sigma^3}\right) e^{\frac{-t^2}{2\sigma^2}}$$

(Equation 4.6)

the discrete convolution forms are expressed as:

$$x(t,\sigma) = x(t) \otimes g(t,\sigma) = \int_{-\infty}^{\infty} x(u) g(t-u,\sigma)\,du = \int_{-\infty}^{\infty} x(u) \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(t-u)^2}{2\sigma^2}}$$

$$= \sum_{u=0}^{N-1} x(u) \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(t-u)^2}{2\sigma^2}}$$

$$\dot{x}(t,\sigma) = x(t) \otimes \dot{g}(t,\sigma) = \int_{-\infty}^{\infty} x(u)\, \dot{g}(t-u,\sigma)\,du = \int_{-\infty}^{\infty} x(u)\left(-\frac{t-u}{\sigma^3\sqrt{2\pi}} e^{\frac{-(t-u)^2}{2\sigma^2}}\right)du$$

$$= \sum_{u=0}^{N-1} x(u)\left(-\frac{t-u}{\sigma^3\sqrt{2\pi}} e^{\frac{-(t-u)^2}{2\sigma^2}}\right)du$$

$$\ddot{x}(t,\sigma) = x(t) \otimes \ddot{g}(t,\sigma) = \int_{-\infty}^{\infty} x(u)\, \ddot{g}(t-u,\sigma)\,du = \int_{-\infty}^{\infty} x(u) \frac{1}{\sqrt{2\pi}}\left(\frac{t^2}{\sigma^5} - \frac{1}{\sigma^3}\right) e^{\frac{-(t-u)^2}{2\sigma^2}}$$

$$= \sum_{u=0}^{N-1} x(u) \frac{1}{\sqrt{2\pi}}\left(\frac{t^2}{\sigma^5} - \frac{1}{\sigma^3}\right) e^{\frac{-(t-u)^2}{2\sigma^2}}$$

where N is the total boundary pixel number

(Equation 4.7)

Similarly

$$y(t,\sigma) = y(t) \otimes g(t,\sigma) = \int_{-\infty}^{\infty} y(u) g(t-u,\sigma) du = \int_{-\infty}^{\infty} y(u) \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(t-u)^2}{2\sigma^2}} = \sum_{u=0}^{N-1} y(u) \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(t-u)^2}{2\sigma^2}}$$

$$\dot{y}(t,\sigma) = y(t) \otimes \dot{g}(t,\sigma) = \int_{-\infty}^{\infty} y(u)\, \dot{g}(t-u,\sigma)\, du = \int_{-\infty}^{\infty} y(u)(-\frac{t-u}{\sigma^3\sqrt{2\pi}} e^{\frac{-(t-u)^2}{2\sigma^2}})du$$

$$= \sum_{u=0}^{N-1} y(u)(-\frac{t-u}{\sigma^3\sqrt{2\pi}} e^{\frac{-(t-u)^2}{2\sigma^2}})du$$

$$\ddot{y}(t,\sigma) = y(t) \otimes \ddot{g}(t,\sigma) = \int_{-\infty}^{\infty} y(u)\, \ddot{g}(t-u,\sigma)\, du = \int_{-\infty}^{\infty} y(u) \frac{1}{\sqrt{2\pi}} (\frac{t^2}{\sigma^5} - \frac{1}{\sigma^3}) e^{\frac{-(t-u)^2}{2\sigma^2}}$$

$$= \sum_{u=0}^{N-1} y(u) \frac{1}{\sqrt{2\pi}} (\frac{t^2}{\sigma^5} - \frac{1}{\sigma^3}) e^{\frac{-(t-u)^2}{2\sigma^2}}$$

where N is the total boundary pixel number

(Equation 4.8)

Figure 4.5 (a) (b) illustrate an original screw driver boundary and an overlapped wrench and cutter boundary. Figure 4.6 and Figure 4.7 show their smoothed forms under different smoothing widths. Figure 4.8 and Figure 4.9 plot the corresponding smoothed curvature functions. It can be seen that an appropriate smoothing parameter is important for the final smoothing effect. For example, when $\sigma=2$, some small local perturbations still exist and the pixels with local extreme curvatures cannot best describe the curve features. Also, It can be imagined that when σ approaches infinite, a curve can be smoothed to a straight line. Therefore, the appropriate selection of the smoothing width is important for the final shape description. By experimentation, it was found that the optimal smoothing width for hand-tools was $\sigma=8$.

However, in practice, when performing the convolution, the initial and terminal portions of the boundary curve are always false after the smoothing. The false portion

varies with the change of the smoothing width. A bigger σ causes a bigger false portion. This problem happens because the convolution is always out of bound at the initial and terminal portion. As shown in Figure 4.4, this problem can be overcome by adding a mask respectively to the initial and the terminal portion. Since the boundary is a closed curve, the data of the initial mask can be extracted from the terminal portion. Similarly, the data of the terminal mask can be extracted from the initial portion. After the convolution, the data retrieved from the middle portion will be the data of desired smoothed curve. In the experiment, the mask size is chosen as 40.

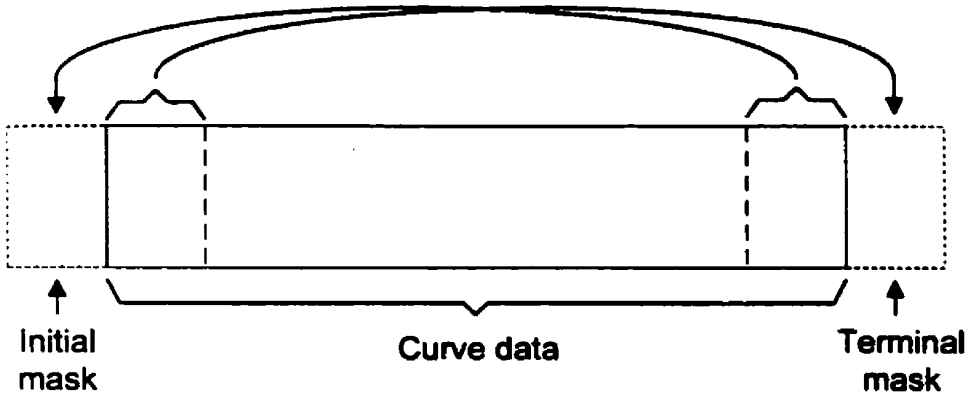
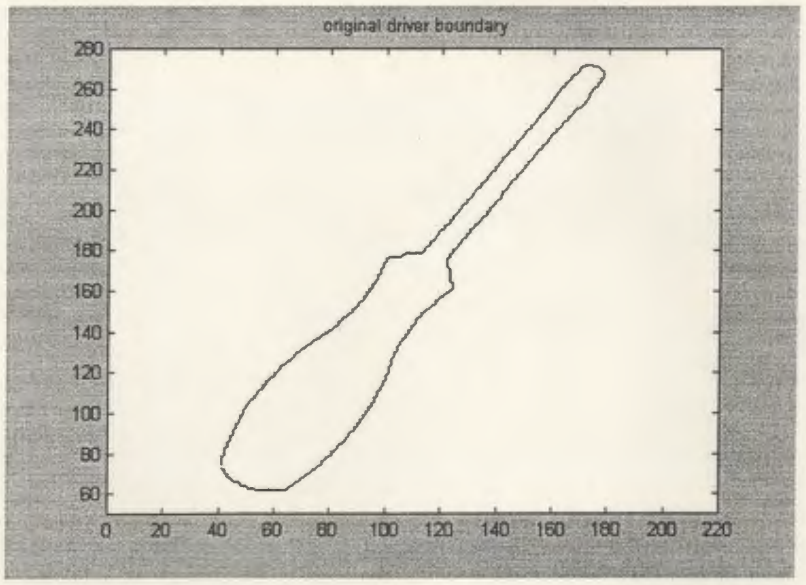
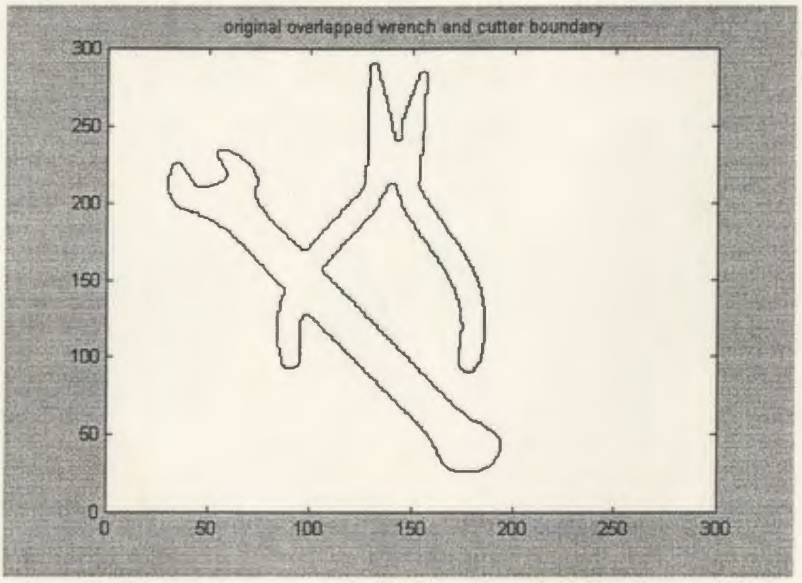


Figure 4.4 Formation of the initial and the terminal mask in smoothing



(a)



(b)

Figure 4.5 (a) Original screw driver boundary
(b) Original overlapped wrench and cutter boundary

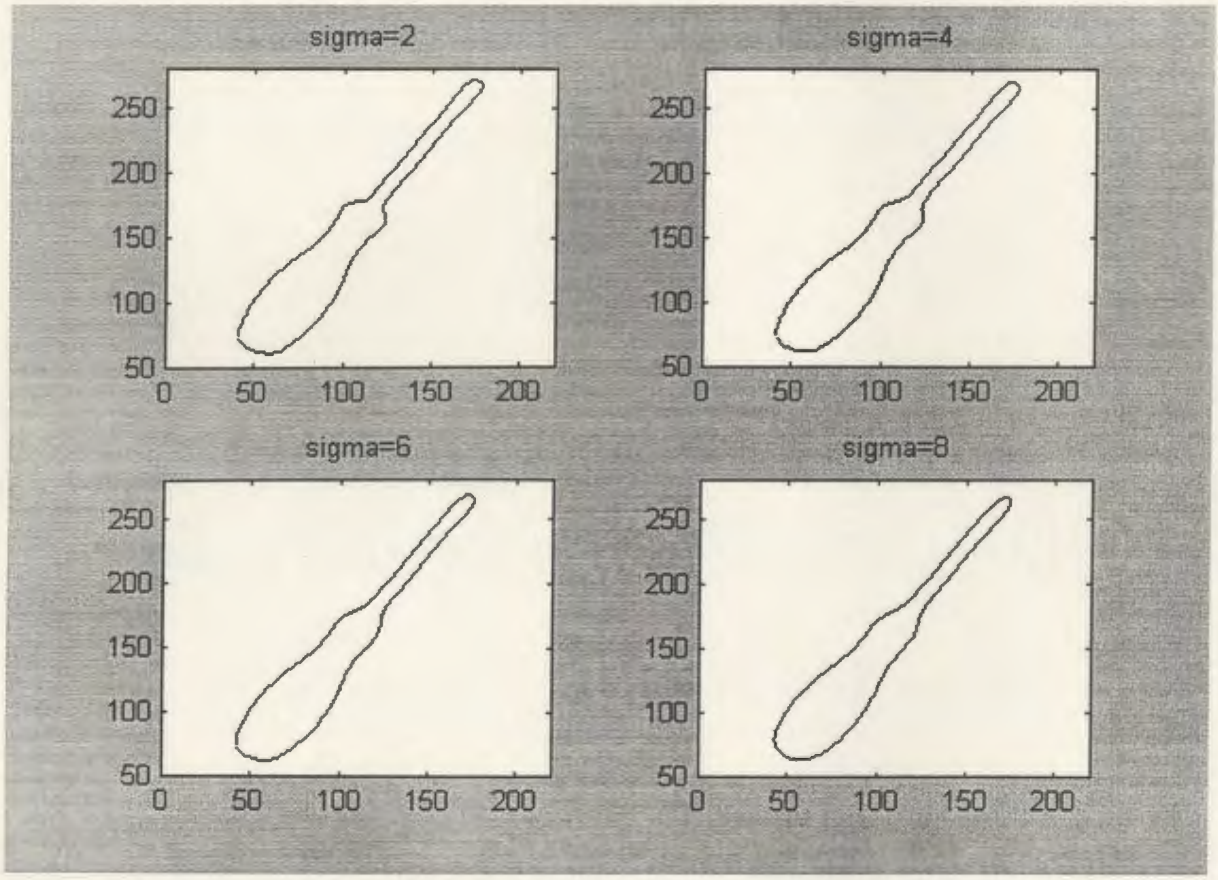


Figure 4.6 Smoothed boundaries of Figure 4.5 (a) under different smoothing widths

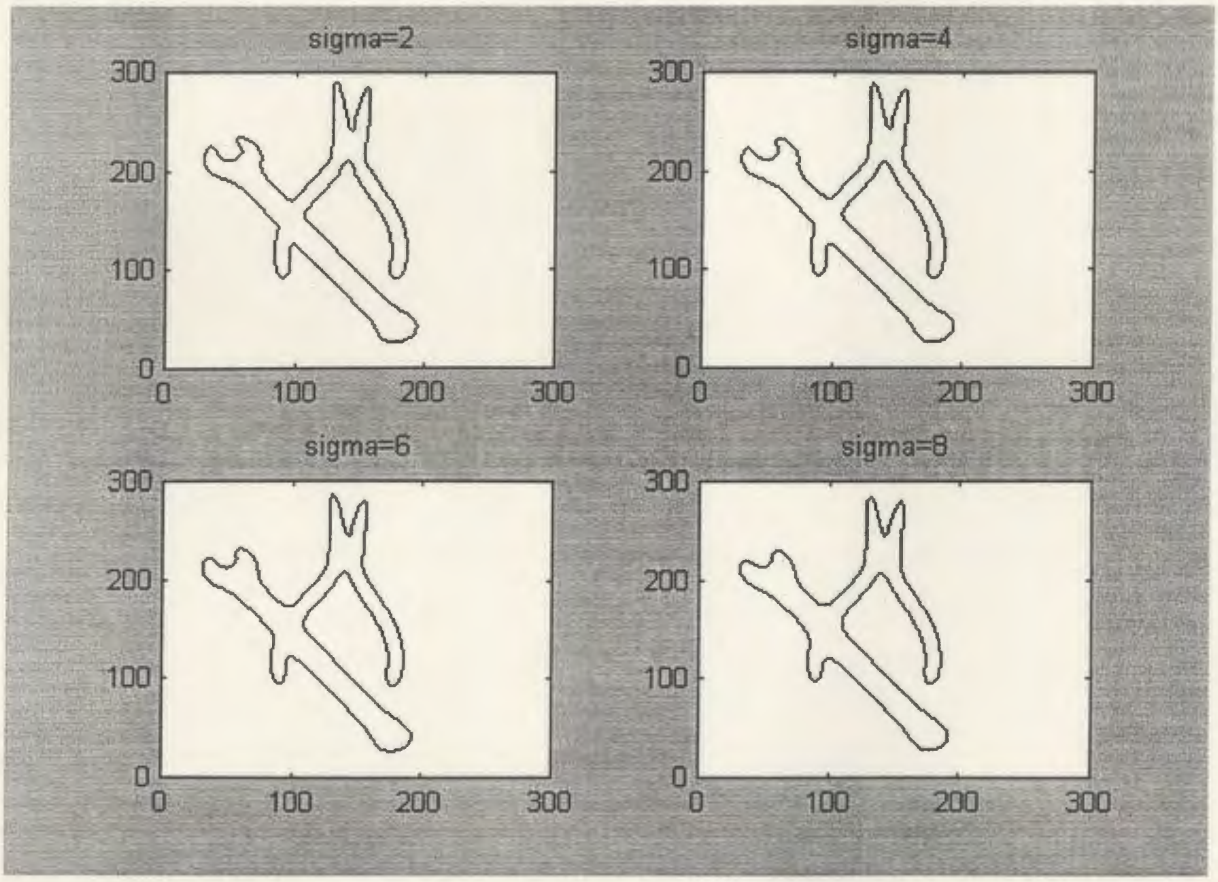


Figure 4.7: Smoothed boundaries of Figure 4.5 (b) under different smoothing widths

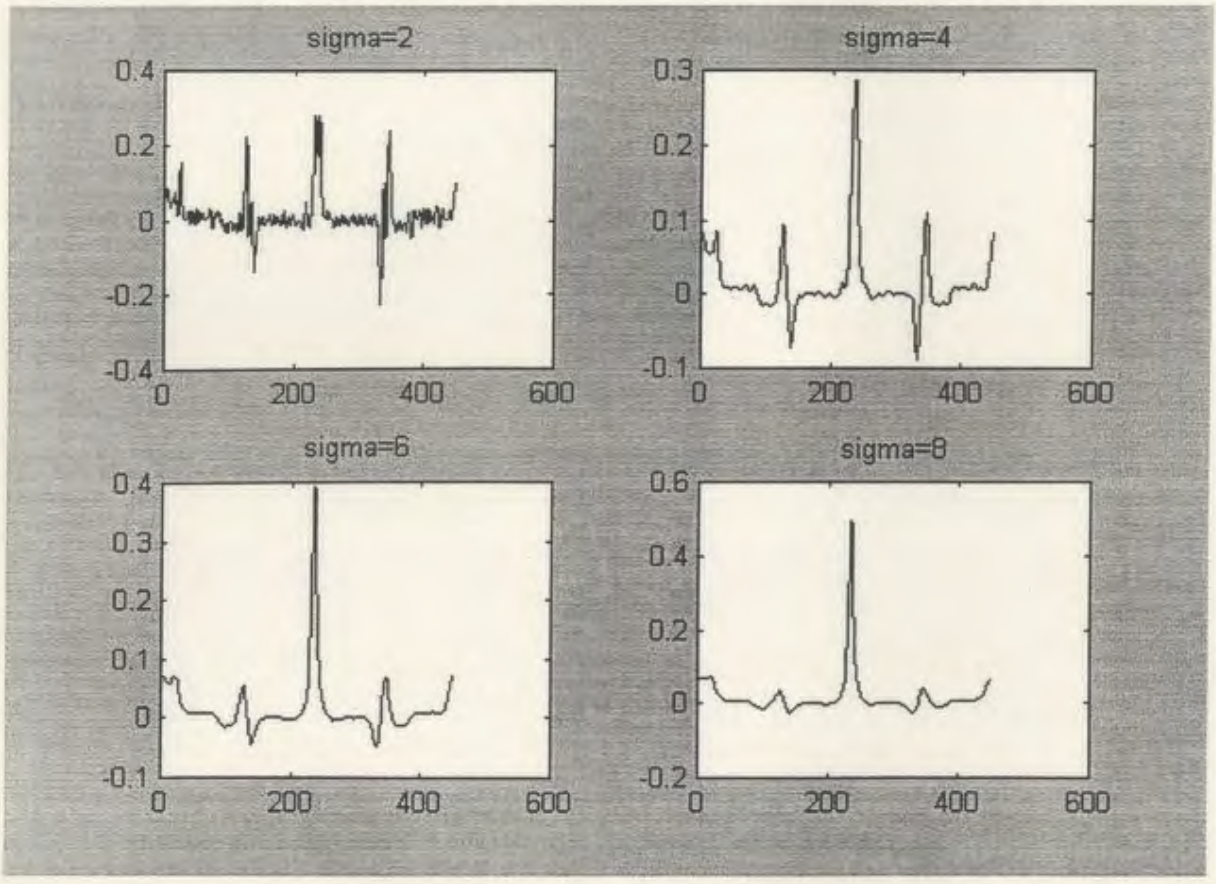


Figure 4.8 Corresponding curvature function plottings of Figure 4.6

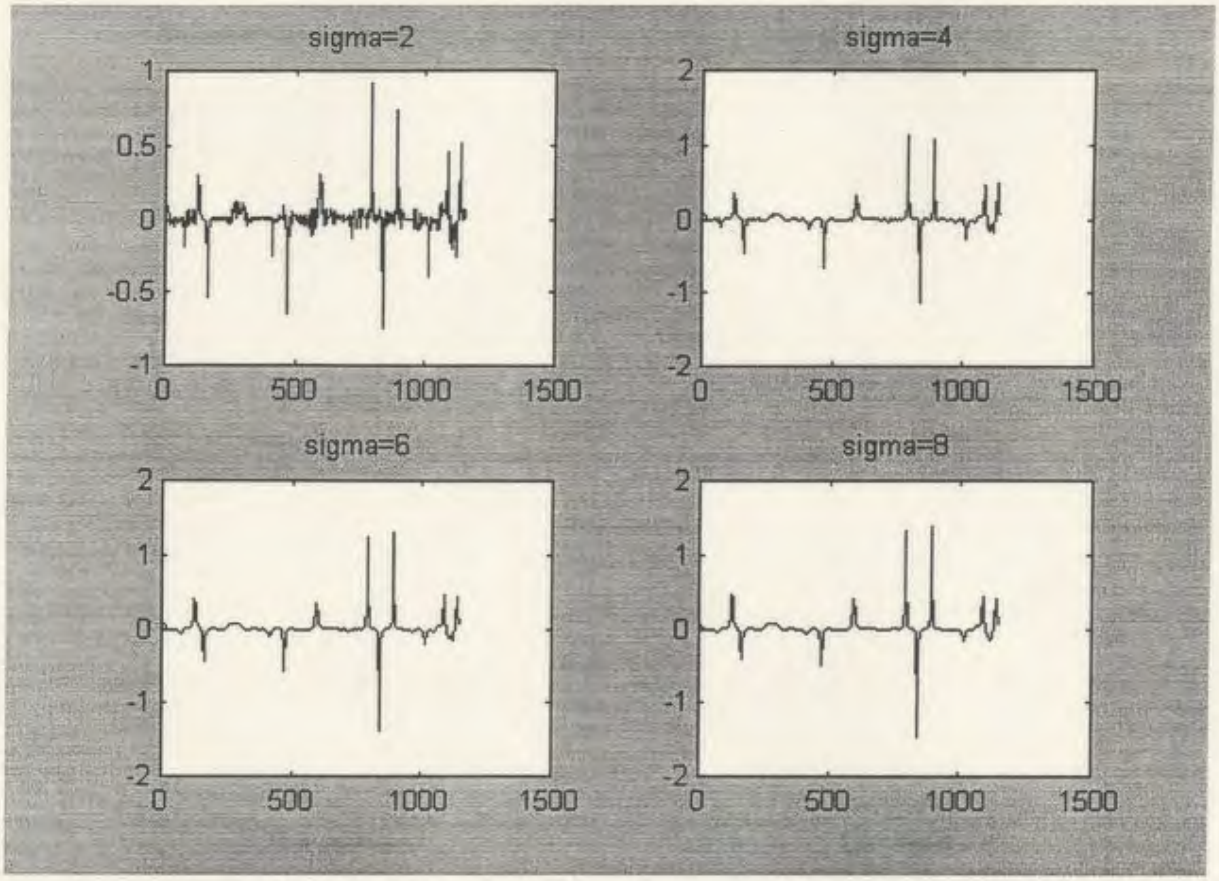


Figure 4.9 Corresponding curvature function plottings of Figure 4.7

4.2.3 Dominant Point Extraction

The dominant point extraction is performed on the smoothed boundary. A point whose curvature is either local positive maximum or negative minimum is regarded as a dominant point candidate, and the curvature is calculated according to (Equation 4.4). The term "candidate" is used here because not all of the points along the boundary satisfying the "local extreme" condition are final dominant points. As shown in the curvature function plots in Figure 4.8-4.9, some dominant point candidates having very small curvature magnitudes have to be discarded because the small perturbations along the boundaries cannot be viewed as the salient features. In order for automatic extraction of the dominant points, the candidates first are detected by the following derivative equation which is usually used to solve extreme value problems:

$$\frac{dc(t)}{dt} = 0 \qquad \text{(Equation 4.9)}$$

Where *c(t)* is the curvature of *tth* point along the boundary. (Equation 4.9) is the continuous form, its discrete expression may be described as:

$$\begin{aligned} &\textit{if} \quad c(t) > c(t-1) \ \textit{and} \ c(t) > c(t+1) \\ &\quad \textit{then} \ c(t) \ \textit{is a dominant point candidate} \\ &t = 0,1,2,\cdots,N, \ N \ \textit{is the boundary pixel number} \end{aligned}$$

As shown in Figure 4.8-4.9, many points along the boundaries may satisfy such a criterion. In order for the final feature points to be the most salient points along the boundaries, some spurious candidates have to be removed. The removing of the spurious candidates is performed by the following steps:

1) detect the maximum curvature magnitude of all the candidate points
2) setting a magnitude threshold (some percentage of the maximum curvature magnitude)
3) assign the candidates whose curvature magnitudes exceed the setting threshold as the final dominant points.

The selection of an appropriate threshold is critical to the final extraction of the dominant points and is greatly affected by the smoothing width. For different objects, the threshold may be different. The threshold selected in the proposed software package is an experimental parameter and 0.045 is optimal for most hand-tool images. Generally, the value is from 0.035 to 0.050. The dominant points extracted by this method remain stable under image rotation, shifting and scaling, as shown in Figure 4.10-4.13.

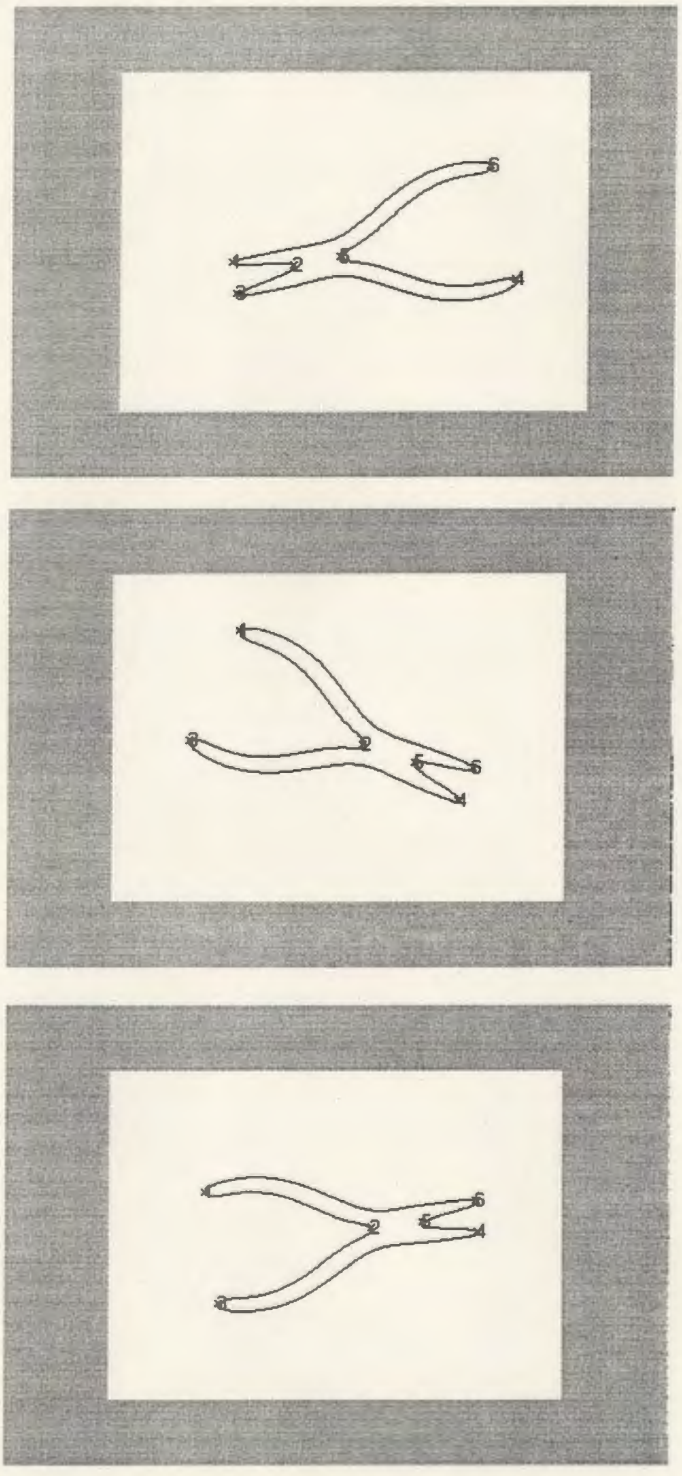


Figure 4.10 Dominant points of a cutter under rotation and shifting

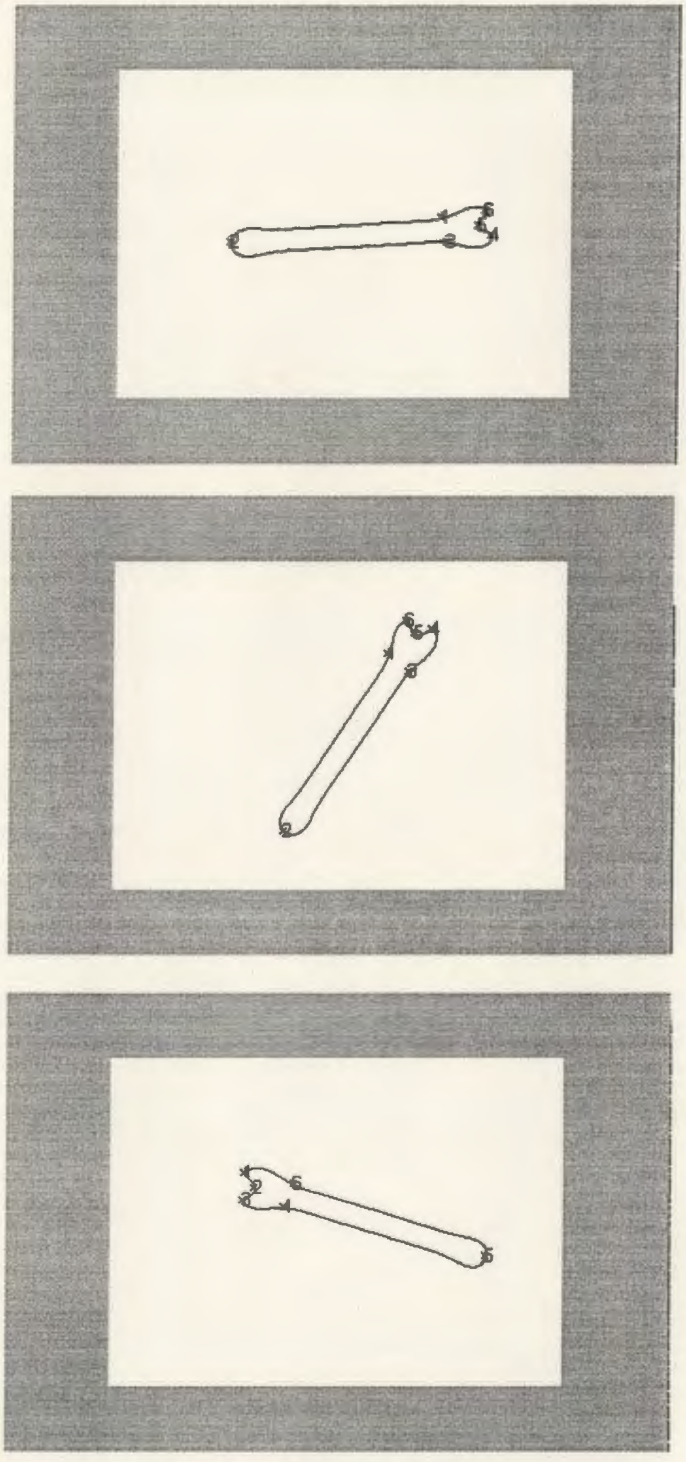


Figure 4.11 Dominants of a wrench under rotation and shifting

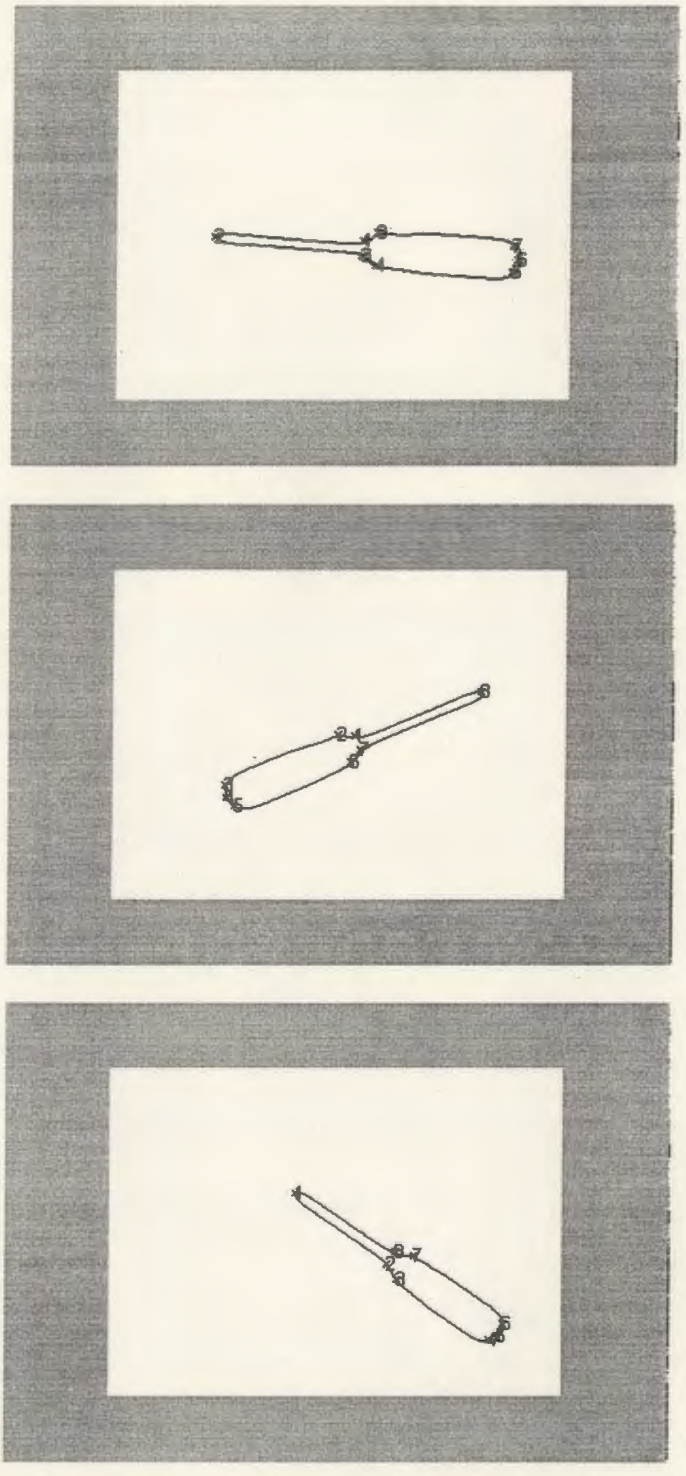


Figure 4.12 Dominant points of a screw driver under rotation, scaling and shifting

4.3 Building Disparity Matrix

As mentioned in Chapter 2, to explore the resemblance between the objects in the model and the test image, the proprieties of their feature points are compared. A disparity matrix is formed to perform the initial local comparison and a modified Hopfield neural network is used for the global optimization. For the local comparison, the elements in a disparity matrix indicate whether the corresponding feature points in the model and the test image are matched or not. To form the disparity matrix, the number of the dominant points of the model object is regarded as the column number. Similarly, the number of the dominant points of the test object is regarded as the row number. Therefore, the *ijth* element of the disparity matrix indicates the measure between the *ith* dominant point of the test object and the *jth* dominant point of the model object.

The method of building a disparity matrix for the occlude object recognition has been applied since late 1980s (Ansari, N. and Delp, E.J., 1990) (Han, M.H. and Jang, D.S, 1990) (Nasrabadi, N.M. and Li, W., 1991) (Tsang, P.W.M. *et al.*, 1994). In such applications, the identity of a partially occluded object in the scene image is determined by how well its dominant points match the dominant points of a model object. A matched dominant point pair between the test and the model object is called a "correspondence pair". To accurately reflect the resemblance between a partially occluded object and a model object in the disparity matrix, the properties of the

dominant points used for the comparison should be able to represent the geometry of the shape and denote the relationships of the dominant points.

Tsang (Tsang, P.W.M. *et al.*, 1994) used a 3-point matching algorithm to compare two objects (Chamfer ¾ Distance Transformation used as the similarity evaluation). He first used the curvature guided polygonal approximation to approximate a boundary curve and the vertices of the polygon were regarded as the feature points. Then the feature points were arranged as a set of ordered triplets (three consecutive points), the length ratio and the interior angle between the neighboring segments connecting the triplet points were regarded as the local proprieties of the feature points. If both the length ratio and interior angle of a feature point in the test and the model object is similar within a tolerant threshold, then they can be viewed as a correspondence pair. Nasrabadi and Li (Nasrabadi, N. and Li, W., 1991) used the Pavlidis' polygonal approximation method to extract the feature points on the object boundary. The line segments used as the feature properties were compared and a disparity matrix was formed. Ansari and Li (Ansari, N. and Li, K.W., 1993) used the polygonal approximation with the cardinal curvature detection to extract the feature points which were arranged as a set of ordered triplets. The sphericity (Appendix II) of a triplet is regarded as the local propriety. The disparity matrix was built based on the sphericity comparison of the feature points between the test and the model image.

Tsang's method, a 3-point matching algorithm for building the disparity matrix, was used in the thesis because this 3-point matching method can reflect the local proprieties between neighboring feature points and it is computationally simple

compared with the sphericity computation. In Tsang's disparity matrix, the elements are heuristic since they are the values of length ratios and interior angles. In the proposed disparity matrix, since the initial matching result is used as the input of the succeeding discrete modified Hopfield neural network for the global optimization, the elements are binary, either 1 or 0, which means matched or mismatched. Some early papers (Gorman, J.W. *et al.*, 1988) (Liu, H.C. and Srinath, M.D., 1990) extracted the local features and expressed them in a heuristic way which would be processed by the distance transformation and the dynamic programming. Since the elements in the proposed disparity matrix are binary, they actually simplify the computation.

As discussed earlier, the dominant points are detected by searching along the boundary counterclockwise, their coordinates and sequences are recorded in an ordered list. If the *ith* dominant point of the model object is denoted as T_i, the length ratio of the triplet (T_{i-1}, T_i, T_{i+1}) (indexing point T_i) is expressed as:

$$R(T_{i-1}, T_i, T_{i+1}) = \frac{L(T_{i-1}, T_i)}{L(T_i, T_{i+1})} \qquad \text{(Equation 4.10)}$$

Where $L(T_i, T_j)$ denotes the length (Euclidean distance) between the neighboring dominant point T_i and T_j . The interior angle of (T_{i-1}, T_i, T_{i+1}) (indexing point T_i) is expressed as following:

$$\theta(T_i) = \cos^{-1} \frac{< L(T_{i-1}, T_i), L(T_{i+1}, T_i) >}{L(T_{i-1}, T_i) \bullet L(T_{i+1}, T_i)} \qquad \text{(Equation 4.11)}$$

Where $< A, B >$ *is the inner product of A and B*

Suppose a triplet of the test object is (M_{j-1}, M_j, M_{j+1}) (indexing point M_j), the criterion for matching the local properties between T_i and M_j is expressed as following:

$$Thres1 < \frac{R(T_{i-1},T_i,T_{i+1})}{R(M_{i-1},M_i,M_{i+1})} < \frac{1}{Thres1} \qquad \text{(Equation 4.12)}$$
$$|\theta(T_{i-1},T_i,T_{i+1}) - \theta(M_{i-1},M_i,M_{i+1})| < Thres2$$

Where *Thres1* is the similarity tolerance for the length ratio comparison and *Thres2* is the similarity tolerance for the angle comparison. The selection of *Thres1* and *Thres2* is based on the specification condition. In the experiment, *Thres1* varies form *0.8* to *1* and *Thres2* varies from *3* to *10*.

The procedure for building the disparity matrix is first to choose a triplet from the ordered dominant points of the model object and compare it with every triplet from the test object, then to perform the matching using the similarity criterion (Equation 4.12). If the *ith* triplet from the model object and the *jth* triplet from the test object are matched, the (i, j) element of the disparity matrix is set to *1*, otherwise is set to *0*. Thus by indexing all the triplets of the model object, the binary disparity matrix can be built and its elements indicate the matching measure of the feature points between the model and the test object. Figure 4.13-Figure 4.16 illustrate a set of model and test images, Table 4.1-Table 4.4 show their corresponding disparity matrices.

However, the disparity matrix is only based on a local comparison. The neighboring length ratio, interior angle and sphericity only explore the local proprieties of a feature point. From a global view, the relationships of a feature point with the rest ones are not explored. For occluded object discrimination, it is quite possible for two

actually different objects have some similar local portions, which may cause fault of matching if only based on a local comparison. On the other side, when comparing the objects under occlusion, the matched elements may form a set of sporadic clusters in the disparity matrix (See Table 4.1-4.6), which are not reliable for the object recognition. Hence, the 3-point matching algorithm can only serve as an initial comparison and the global optimization is necessary for an accurate and reliable discrimination.

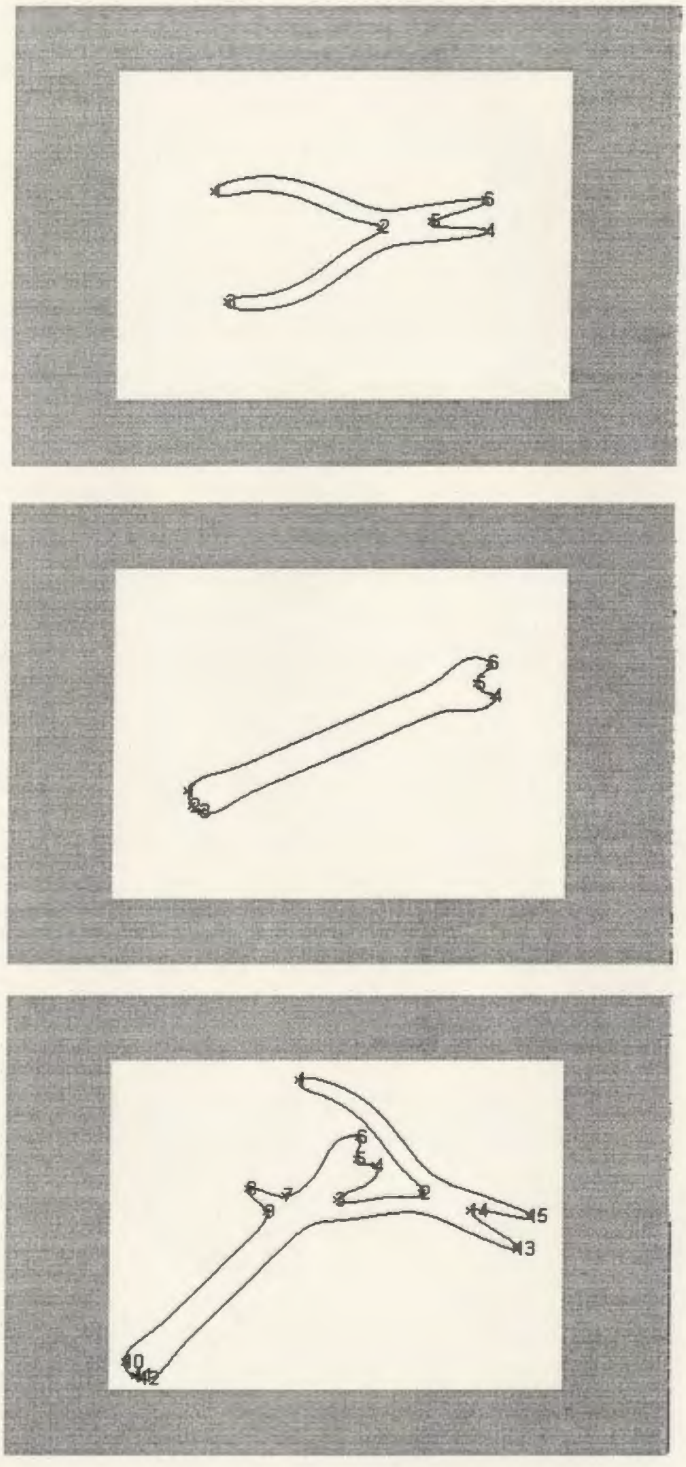


Figure 4.13 A model cutter, screw driver and their overlapped image

Table 4.1 Disparity matrix of the cutter and the overlapped image in Figure 4.13

(a) The indexed length ratios and the interior angles of the model cutter

(b)

Sequence	1	2	3	4	5	6
Length ratio	1.592	0.981	0.633	4.450	0.997	0.228
Interior angle	11.94	41.62	11.59	26.50	34.28	25.87

(b) The indexed length ratios and the interior angle of the overlapped cutter and wrench

Sequence	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Length ratio	1.60	2.01	1.64	2.64	0.87	0.24	2.55	1.23	0.14	12.1	2.72	0.02	6.42	1.02	0.22
Interior angle	12.1	48.1	38.0	63.8	107	46.8	129	41.7	100	102	151	134	58.9	34.2	25.9

(c) Disparity matrix forming by 3-point matching algorithm (Thres1=1.2, Thres2=10)

Index number (scene\model)	1	2	3	4	5	6
1	1	0	0	0	0	0
2	0	0	0	0	0	0
3	0	0	0	0	0	0
4	0	0	0	0	0	0
5	0	0	0	0	0	0
6	0	0	0	0	0	0
7	0	0	0	0	0	0
8	0	0	0	0	0	0
9	0	0	0	0	0	0
10	0	0	0	0	0	0
11	0	0	0	0	0	0
12	0	0	0	0	0	0
13	0	0	0	0	0	0
14	0	1	0	0	1	0
15	0	0	0	0	0	1

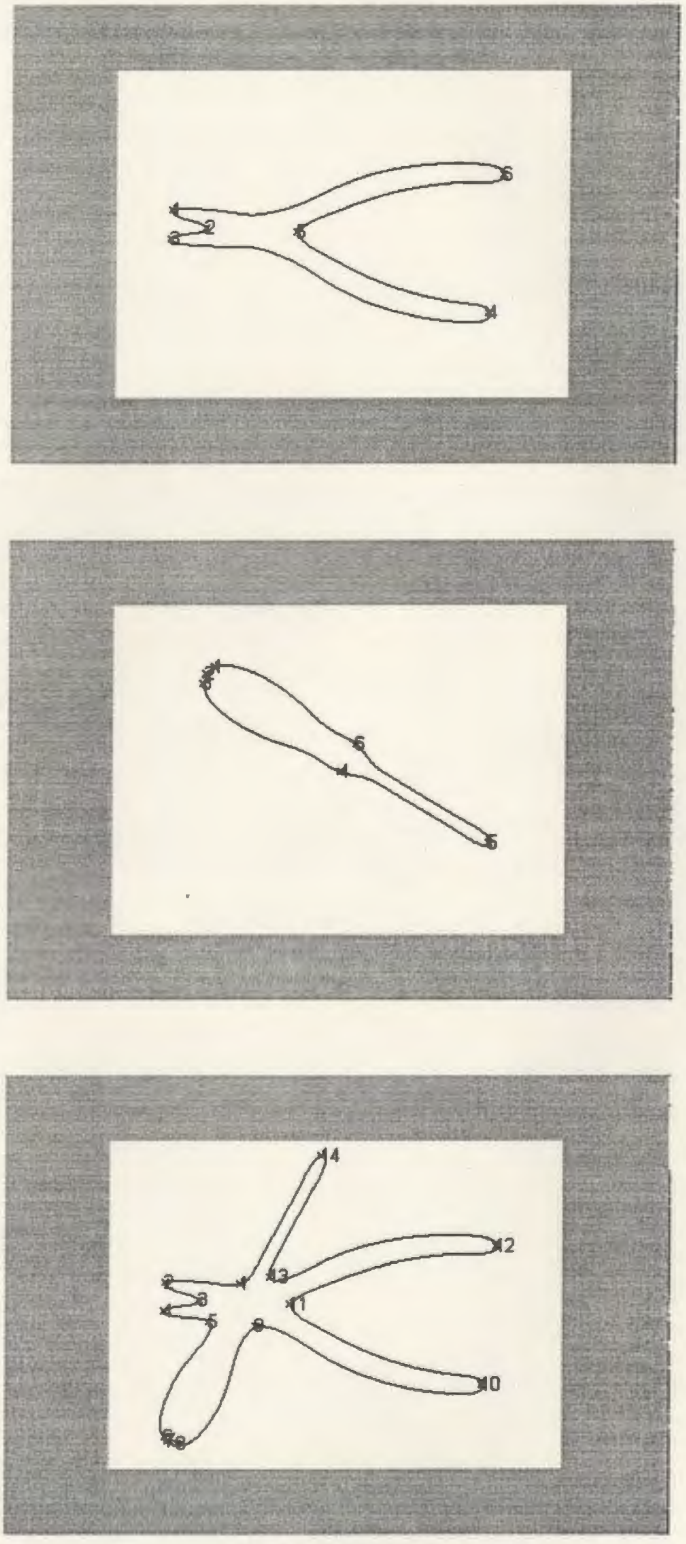


Figure 4.14 A wire clipper, screw driver and their overlapped image

Table 4.2 Disparity matrix of wire clipper and the overlapped image in Figure 4.14

(a) The indexed length ratios and the interior angles of the wire clipper

Sequence	1	2	3	4	5	6
Length ratio	8.391	1.036	0.117	1.567	0.976	0.643
Interior angle	33.79	44.58	30.54	10.01	39.50	9.736

(b) The indexed length ratios and the interior angles of the overlapped clipper and driver

Sequence	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Length ratio	2.02	1.92	1.04	0.80	0.38	21.2	0.58	0.07	0.62	1.12	0.98	0.94	1.70	0.87
Interior angle	120	26	44.5	30.7	82.8	114	151	106	107	8.41	39.5	8.22	59.9	9.26

(c) Disparity matrix forming by 3-point matching algorithm (Thres1=1.2, Thres2=10)

Index number (scene\model)	1	2	3	4	5	6
1	0	0	0	0	0	0
2	0	0	0	0	0	0
3	0	1	0	0	1	0
4	0	0	0	0	1	0
5	0	0	0	0	0	0
6	0	0	0	0	0	0
7	0	0	0	0	0	0
8	0	0	0	0	0	0
9	0	0	0	0	0	0
10	0	0	0	0	0	0
11	0	1	0	0	1	0
12	0	0	0	0	0	0
13	0	0	0	0	0	0
14	0	0	0	0	0	0

Figure 4.15 A model wire clipper, cutter, wrench and their overlapped image

Table 4.3 Disparity matrix of the cutter and the overlapped image in Figure 4.15

Index number (scene\model)	**1**	**2**	**3**	**4**	**5**	**6**
1	0	0	0	0	0	0
2	0	0	0	0	0	0
3	0	0	0	0	0	0
4	0	0	0	0	0	0
5	0	1	0	0	0	0
6	0	0	0	0	0	0
7	0	0	0	0	1	0
8	0	0	0	0	0	1
9	1	0	0	0	0	0
10	0	0	0	0	0	0
11	0	0	0	0	0	0
12	0	0	0	0	0	0
13	0	0	0	0	0	0
14	0	0	0	0	0	0
15	0	0	0	0	0	0
16	0	0	0	0	0	0
17	0	1	0	0	0	0
18	0	0	0	0	0	0
19	0	0	0	0	0	0
20	0	0	0	0	0	0

Table 4.4 Disparity matrix of the wire clipper and the overlapped image in Figure 4.15

Index number (scene\model)	**1**	**2**	**3**	**4**	**5**	**6**
1	1	0	0	0	0	0
2	0	1	0	0	0	0
3	0	0	0	0	0	0
4	0	0	0	0	0	0
5	0	0	0	0	0	0
6	0	0	0	0	0	0
7	0	0	0	0	1	0
8	0	0	0	0	0	0
9	0	0	0	0	0	0
10	0	0	0	0	0	0
11	0	0	0	0	0	0
12	0	1	0	0	0	0
13	0	0	0	0	0	0
14	0	0	0	0	0	0
15	0	0	0	0	0	0
16	0	0	0	0	0	0
17	0	0	0	0	0	0
18	0	0	0	0	0	0
19	0	0	0	0	0	0
20	0	0	0	0	0	1

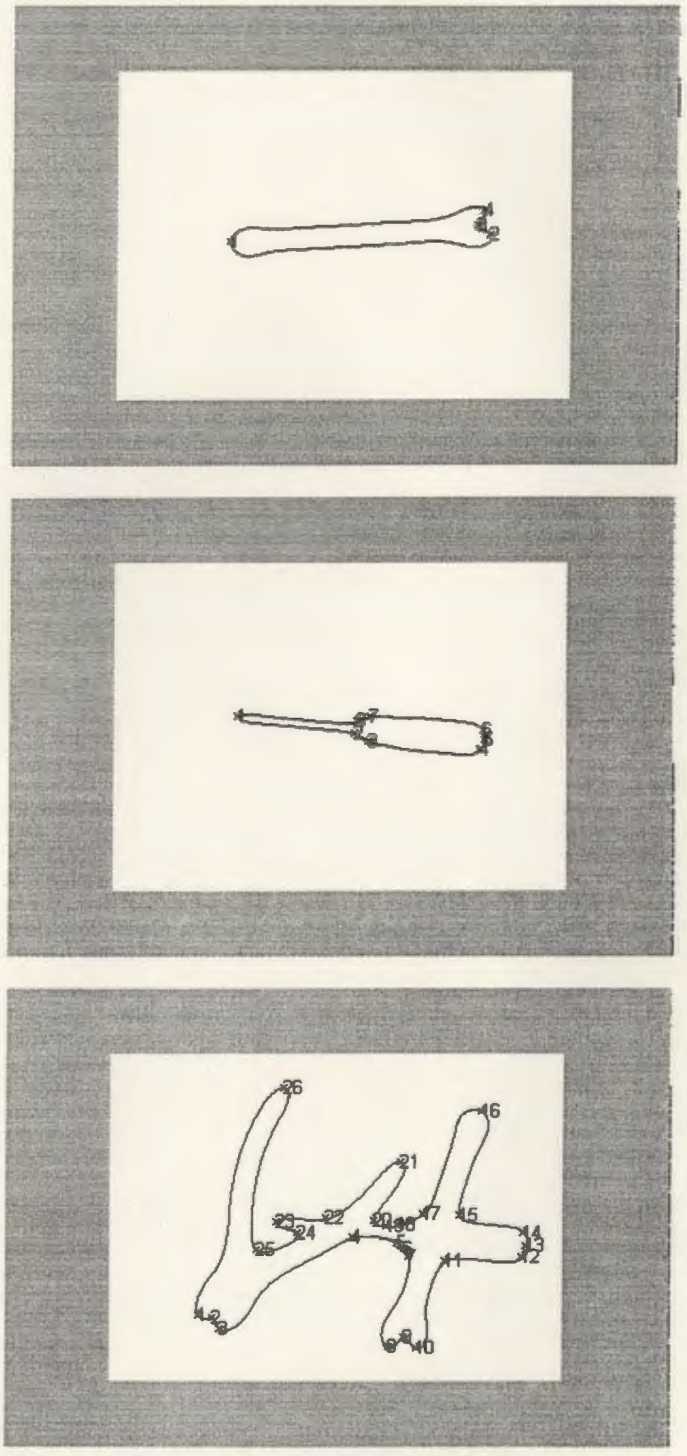


Figure 4.16 A wire clipper, screw driver, wrench and their overlapped image

Table 4.5 Disparity matrix of the wrench and the overlapped image in Figure 4.16

Index number (scene\model)	**1**	**2**	**3**	**4**
1	0	0	0	0
2	0	0	0	0
3	0	0	0	0
4	0	0	0	0
5	0	0	0	0
6	0	0	0	0
7	0	0	0	0
8	0	0	0	0
9	0	0	1	0
10	0	0	0	0
11	0	1	0	0
12	0	0	0	0
13	0	0	0	0
14	0	0	0	0
15	0	0	0	0
16	0	0	0	0
17	0	0	0	0
18	0	0	0	0
19	0	0	0	0
20	0	0	0	0
21	0	0	0	0
22	0	0	0	0
23	0	0	0	0
24	0	0	0	0
25	0	0	0	0
26	0	0	0	0

Table 4.6 Disparity matrix of the screw driver and the overlapped image in Figure 4.16

Index number (scene\model)	1	2	3	4	5	6	7	8
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0
6	0	0	0	0	1	0	0	0
7	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0
13	0	0	0	0	1	0	0	0
14	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0	0
22	0	0	0	0	0	0	0	0
23	0	0	0	0	0	0	0	0
24	0	0	0	0	0	0	0	0
25	0	0	0	0	0	0	0	0
26	0	0	0	0	0	0	0	0

4.4 Modified Hopfield Neural Network

4.4.1 Neural Networks In General

Neural networks are the computing structures simulating the human brain. As the brain neurons interconnect and interact between each other, computing units of neural networks are coupled by synaptic weights. The human brain can process the input information parallelly, similarly, neural networks provide high computation efficiency due to their inherently parallel structures. Under the applications involved extensive computation, neural networks are superior to conventional algorithms.

The feedforward backpropagation (BP) and the Hopfield network are the most popular network structures in the neural network domain. The BP network is used in weather predication, image analysis, voice recognition, etc (Tsoukalas, L.H. and Uhrig, R.E., 1997). The Hopfield network may be viewed as a kind of content-addressable memory and is usually used to recover the correct pattern from its incomplete or noise version (Schalkoff, R., 1992).

4.4.2 Hopfield Neural Network

The Hopfield network operates in an unsupervised manner and can be used to solve the optimization problem, which is the reason why the Hopfield is selected in the

thesis. This optimization property lies in the physical principle of the Hopfield network. In the Hopfield network, the information is stored in a dynamically stable configuration. This configuration can be described by a Liapunov (energy) function which is a monotonically decreasing function of the network state. The retrieval of the information stored in the network is accomplished by randomly picking a neuron and updating it according to an asynchronous dynamic procedure which is repeated until no further state changes (Pandya, A.S. and Macy, R.B., 1996).

4.4.3 Global Matching Problem

As discussed earlier this chapter, the initial matching between the occluded object and the model object is first performed by a 3-point matching algorithm which explores the local properties of the feature points. This local matching algorithm may cause problem in the occluded object discrimination. For example, two quite different objects may have some small similar portions and the feature points in an object may be divided into a set of small sporadic clusters that are not reliable for an accurate discrimination. As shown in Table 4.1-Table 4.6, the elements in the disparity matrix cannot provide a clear and accurate discrimination result.

A modified Hopfield network is applied to the thesis as a global optimization tool to obtain the optimal match between the model and the test object based on the initial comparison. Thus, the disparity matrix is regarded as the input of the network. In the standard Hopfield neural network (Figure 2.2) (Haykin, S., 1994), the symmetric synaptic weights are computed as the outer product of the fundamental memories. In

this modified version, the symmetric synaptic weights coupling the neurons (elements in the disparity matrix) are more complicated (As shown in Figure 4.19) because the network is two-dimensional.

4.4.3.1 Deficiencies of Local Property Comparison

As shown in Table 4.1-Table 4.6, the elements in the disparity matrices cannot accurately reflect the resemblance between the corresponding model and the overlapped objects because some actual correspondence pairs are not obtained since only the neighboring length ratio and the interior angle are considered as the matching criterion. For the point *5* (triplet points *4, 5, 6*) in the model clipper of Figure 4.16, it cannot be identified as the point *25* in the overlapped image when its indexed triplet points (*21, 25, 26)* are occluded or separated.

Another deficiency is that the geographic relationships of the remote dominant points are not explored by the 3-point matching algorithm. To overcome both deficiencies, the modified Hopfield neural network is applied here.

4.4.3.2 Modified Hopfield Neural Network

As shown in Figure 4.17, a 2-D matrix is constructed to form the modified Hopfield network. The row and the column number of the matrix respectively indicate the feature point number of the test and model object. The values of the elements denote the initial states of the neurons. Since the disparity matrix only contains the binary elements (1 or 0), the network is a discrete version. If the test object has M dominant

points and the model object has N dominant points, then a $M \times N$ network is formed and the matching process may be characterized as minimizing the following energy function:

$$E = -\frac{A}{2}\sum_{i=1}^{M}\sum_{k=1}^{N}\sum_{j=1}^{M}\sum_{l=1}^{N} C_{ikjl} V_{ik} V_{jl} + \frac{B}{2}\sum_{i=1}^{M}(1-\sum_{k=1}^{N} V_{ik})^2 + \frac{C}{2}\sum_{k=1}^{N}(1-\sum_{i=1}^{M} V_{ik})^2 \quad \text{(Equation 4.13)}$$

Where V_{ik} is a neuron state which converges to "1" if the *ith* feature point of the test object matches the *kth* feature point of the model object. Otherwise, it converges to 0. i and j are the indexing variables for the model feature points. k and l are the indexing variables for the scene feature points.

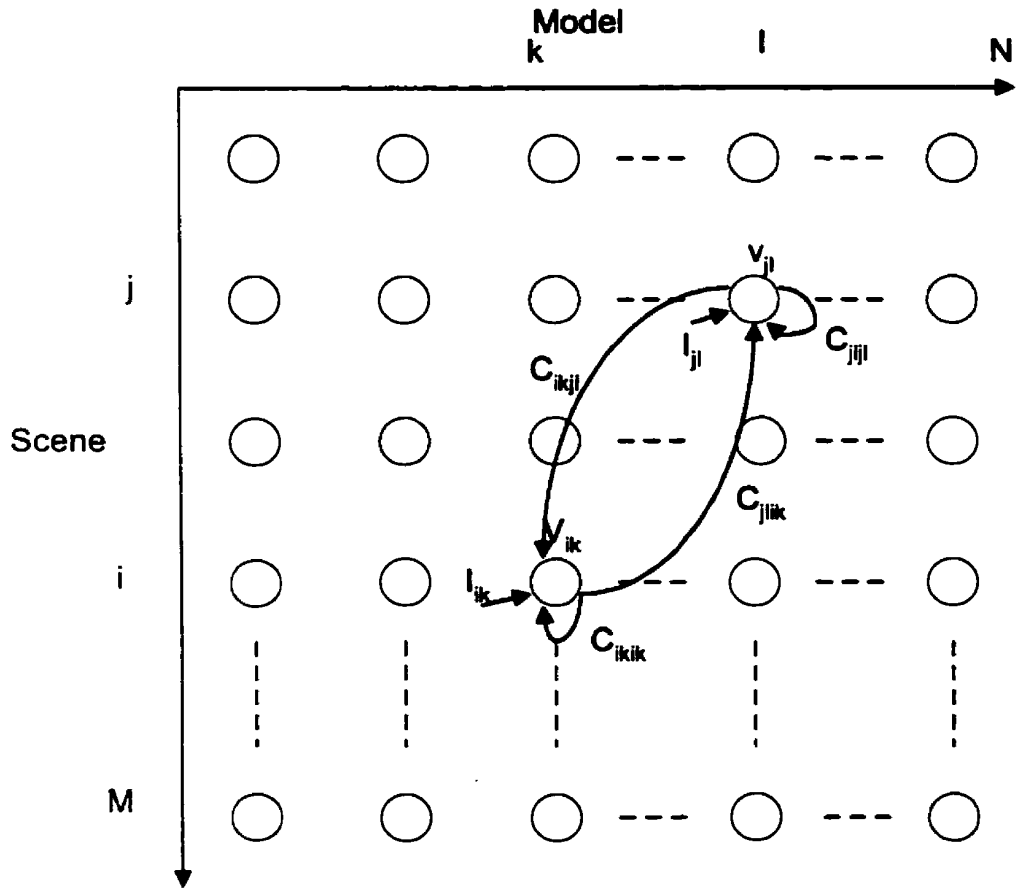


Figure 4.17 A 2-D modified Hopfield neural network model used for discrimination

Like the energy function in the Travelling Salesperson Problem (Appendix III) and the energy function presented by Nasrabadi-Li (Nasrabadi, N.M and Li, W., 1991), the first term of (Equation 4.13) is a compatibility constraint which is used to impose the condition that the lowest energy should favor the best set of correspondence pairs. The last two terms of (Equation 4.13) are used to enforce the uniqueness constraint so that each point in the model object matches only one point in the test object, that is, the summation of the outputs of the neurons in each row or column is no more than *1*.

In the energy function of the Travelling Salesperson Problem, the coefficient *B* and *C* are more emphasized than *A* because *B* and *C* contribute yielding valid solutions. For simplicity, let *A=1* and *B=C=2* in (Equation 4.13). The coupling symmetric synaptic weights C_{ikjl} is denoted as:

$$C_{ikjl} = W_1 \times F(f_i, f_k) + W_2 \times F(f_j, f_l) + W_3 \times F(r_{ij}, r_{kl}) \qquad \text{(Equation 4.14)}$$

Where *F(x, y)* is a step function as shown in Figure 5.1(a). *F(x, y)* has a value of 1 for a positive support and a value of −1 for a negative support. The value of *F(x, y)* is 1 if the difference between *x* and *y* is within a setting threshold, otherwise is −1. Hence, the first term $F(f_i, f_k)$ denotes the measure between the *ith* feature point of the test object and the *kth* feature point of the model object. Similarly, $F(f_j, f_k)$ denotes the measure between *jth* feature point of the test object and the *lth* feature point of the model object. The third term is related to the relational features between two neurons. If the distance between the *ith* and the *jth* feature point in the scene is similar to the distance between the *jth* and the *lth* feature point of the model object within a setting tolerance, then $F(r_{ij}, r_{kl})$ is set to

1, otherwise is set to −1. Hence, the third term actually explores the geographic relation between the two objects. This idea is the key to the successful retrieval of the optimal correspondence pairs.

For the comparison convenience, the weighting factors W_i in C_{ikjl} are normalized as following:

$$\sum_{i=1}^{3} W_i = 1 \qquad \text{(Equation 4.15)}$$

For a Hopfield-style expression, the energy function may be denoted as following:

$$E = -\frac{1}{2}\sum_{i=1}^{M}\sum_{k=1}^{N}\sum_{j=1}^{M}\sum_{l=1}^{N} T_{ikjl} V_{ik} V_{jl} - \sum_{i=1}^{M}\sum_{k=1}^{N} I_{ik} V_{ik} \qquad \text{(Equation 4.16)}$$

It can be shown that (Equation 4.16) is equivalent to (Equation 4.13) by some simple manipulations as following:

$$T_{ikjl} = C_{ikjl} - \delta_{ij} - \delta_{kl} \qquad \text{(Equation 4.17)}$$

and

$$I_{ik} = 2 \qquad \text{(Equation 4.18)}$$

Where δ is the Kronecker delta parameter and is denoted as:

$$\delta_{ij} = \begin{cases} 1 & if \quad i = j \\ 0 & if \quad i \neq j \end{cases} \qquad \text{(Equation 4.19)}$$

It is shown from the energy function that the optimization process tries to provide excitatory and inhibitory support to find the final matching nodes. The dynamic convergence operates parallelly under the global constraints and is characterized as a

stochastic process. The network arrives at a stable state when the energy function is at its minimum state. The convergence process can be summarized as following:

1) The initial states of neurons are set to 1 or −1 based on the measure between the corresponding feature points with the 3-point matching method.
2) Randomly pick up a neuron *(i, k)*.
3) Calculate its input as following:

$$U_{ik} = \sum_{j}\sum_{l}(C_{ikjl} - \delta_{ij} - \delta_{kl})V_{jl} + I_{ik} \qquad \text{(Equation 4.20)}$$

4) Decide the new state of each neuron according to the following rules:

$$\begin{cases} V_{ik} = 1, & \text{if } U_{ik} > 0.5 \\ V_{ik} = 0, & \text{if } U_{ik} < -0.5 \\ V_{ik} \text{ no change} & \text{if } -0.5 \le U_{ik} \le 0.5 \end{cases} \qquad \text{(Equation 4.21)}$$

5) Count the changes of the state. If there is no change after a given number of iterations, stop and go to next step or repeat to process from step (2).
6) Output the final states of neurons that denote the final correspondence pairs between the dominant points of the model object and the test object.

The final discrimination results of Table 4.1-Table4.6 are illustrated in Table 4.7-Table 4.12. After the optimization, the consecutive correspondence pairs reliable for the discrimination are achieved and some initial false correspondence pairs are eliminated.

4.4.3.3 Parameter Discussion

The selection of parameters plays a significant role in this experiment. As shown in (Equation 4.14), three items are used to characterize the symmetric weights C_{ikjl}. For the first two terms, $F(f_i, f_k)$ and $F(f_j, f_l)$, their values are obtained from the initial 3-point matching, that is, if the length ratio and the interior angle of a triplet from the model and the test object are similar within a setting threshold, then the value is set to *1*, otherwise is set to *−1*. The selection of the threshold must be careful and it depends on the specific situation. For example, if a model object and a test object in the scene are almost exactly the same (in scale), for an accurate discrimination, the threshold for the length ratio should be near *1* and the threshold of the interior angle should be around *0*. As the thresholds become larger, the robustness of the algorithm is increased, at the same time the mismatching may occur. On the contrary, as the thresholds become smaller, the algorithm becomes more sensitive to the variance of the object and an actually matched correspondence pair may be overlooked.

The third item indicates the geographic relationships of the feature points. Specifically, it indicates the distance. A structural threshold is applied here because for occluded object discrimination, the distance is usually a more reliable feature than the angle. The selection of the structural threshold is based on the specific objects too. In the experiment, some model objects in the scene have some variances with the compared model objects. Such variances may be obtained by changing the distance and the viewing angle between the camera and objects when taking the pictures. The variances may be used to test the robustness of the algorithm. Generally, the structural parameter is set from *3-10* for a *320X240* image.

The selection of the weighting factors greatly affects the operation time and the final convergence correctness. In Nasrabadi-Li's algorithm (Nasrabadi, N.M. and Li, w.,1991), W_i is different for the isolated and the occluded object discrimination and only the angle feature considered in setting the initial states of the neurons. As stated earlier, the distance is a more reliable feature than the angle. Hence, the values suggested by Nasrabadi and Li are $W_1=W_2=0.2$ and $W_3=0.6$. In Lee's algorithm (Lee, J.S. *et al.*, 1997), the feature points are detected by the wavelet transform and compared based on the integrated multiscale features (IMF). Lee regarded the IMFs nearly important as the feature distance, so he set $W_1=W_2=0.3$ and $W_3=0.4$. In the proposed algorithm, since both the interior angle and the length ratio are considered in determining the initial neuron states, the distance play a role of a compromise between Lee and Nasbaradi-Li's method. From the simulation result, it is found that $W_1=W_2=0.25$ and $W_3=0.5$ can achieve the best optimization result for most objects.

4.4.3.4 Energy Convergence Discussion

The success for the global optimization is based on the convergence of the energy function of the network. Suppose the change of the network energy due to a state change (Lee, J.S., *et al.*, 1997) in V_{ik} is given by:

$$\Delta E = -(\sum_{j=1}^{M}\sum_{l=1}^{N} T_{ikjl}V_{jl} + I_{ik} + \frac{1}{2}T_{ikik}\Delta V_{ik})\Delta V_{ik} \qquad \text{(Equation 4.22)}$$

For the decreasing of the system energy, the two items of (Equation 4.22) should have similar signs. Substituting (Equation 4.17) and (Equation 4.20) into (Equation 4.22), a more simplified version is obtained:

$$\Delta E = -(U_{ik} + \frac{1}{2} T_{ikik} \Delta V_{ik}) \Delta V_{ik} \qquad \text{(Equation 4.23)}$$

From (Equation 4.17), (Equation 4.18) and (Equation 4.19), it is found that

$$\begin{aligned} C_{ikik} &= 1 \\ T_{ikjl} &= C_{ikjl} - \delta_{ij} - \delta_{kl} \\ T_{ikik} &= C_{ikik} - \delta_{ii} - \delta_{kk} = 1 - 1 - 1 = -1 \end{aligned} \qquad \text{(Equation 4.24)}$$

Substituting (Equation 4.24) into (Equation 4.23):

$$\Delta E = -(U_{ik} - \frac{1}{2} \Delta V_{ik}) \Delta V_{ik} \qquad \text{(Equation 4.25)}$$

The neuron V_{ik} may change its state in two cases:

$$\left\{ \begin{array}{l} when \;\; U_{ik} > 0.5, \;\; V_{ik} = 0 \rightarrow 1, \;\; \Delta V_{ik} = 1, \;\; \Delta E < 0 \\ when \;\; U_{ik} < -0.5, \;\; V_{ik} = 1 \rightarrow 0, \;\; \Delta V_{ik} = -1, \;\; \Delta E < 0 \end{array} \right\} \qquad \text{(Equation 4.26)}$$

Thus, the energy of the network is always decreasing, which ensures the convergence of the network.

Table 4.7 Final discrimination result of Table 4.1 after optimization

number (scene\model)	1	2	3	4	5	6
1	1	0	0	0	0	0
2	0	1	0	0	0	0
3	0	0	0	0	0	0
4	0	0	0	0	0	0
5	0	0	0	0	0	0
6	0	0	0	0	0	0
7	0	0	0	0	0	0
8	0	0	1	0	0	0
9	0	0	0	0	0	0
10	0	0	0	0	0	0
11	0	0	0	0	0	0
12	0	0	0	0	0	0
13	0	0	0	1	0	0
14	0	0	0	0	1	0
15	0	0	0	0	0	1

Table 4.8 Final discrimination result of Table 4.2 after optimization

Index number (scene\model)	1	2	3	4	5	6
1	0	0	0	0	0	0
2	1	0	0	0	0	0
3	0	1	0	0	0	0
4	0	0	1	0	0	0
5	0	0	0	0	0	0
6	0	0	0	0	0	0
7	0	0	0	0	0	0
8	0	0	0	0	0	0
9	0	0	0	0	0	0
10	0	0	0	1	0	0
11	0	0	0	0	1	0
12	0	0	0	0	0	1
13	0	0	0	0	0	0
14	0	0	0	0	0	0

Table 4.9 Final discrimination result of Table 4.3 after optimization

Index number (scene\model)	1	2	3	4	5	6
1	0	0	0	0	0	0
2	0	0	0	0	0	0
3	0	0	0	0	0	0
4	1	0	0	0	0	0
5	0	1	0	0	0	0
6	0	0	1	0	0	0
7	0	0	0	0	1	0
8	0	0	0	0	0	1
9	0	0	0	1	0	0
10	0	0	0	0	0	0
11	0	0	0	0	0	0
12	0	0	0	0	0	0
13	0	0	0	0	0	0
14	0	0	0	0	1	0
15	0	0	0	0	0	0
16	0	0	0	0	0	0
17	0	0	0	0	0	1
18	0	0	0	0	0	0
19	0	0	0	0	0	0
20	0	0	0	0	0	0

Table 4.10 Final discrimination result of Table 4.4 after optimization

Index number (scene\model)	**1**	**2**	**3**	**4**	**5**	**6**
1	1	0	0	0	0	0
2	0	1	0	0	0	0
3	0	0	1	0	0	0
4	0	0	0	0	0	0
5	0	0	0	0	1	0
6	0	0	0	0	0	0
7	0	0	0	0	0	0
8	0	0	0	0	0	0
9	0	0	0	0	0	0
10	0	0	0	0	0	0
11	0	0	0	0	0	0
12	0	0	0	0	0	0
13	0	0	0	0	0	0
14	0	0	0	0	0	0
15	0	0	0	0	0	0
16	0	0	0	1	0	0
17	0	0	0	0	0	0
18	0	0	0	0	0	0
19	0	0	0	0	1	0
20	0	0	0	0	0	1

Table 4.11 Final discrimination result of Table 4.5 after optimization

Index number (scene\model)	1	2	3	4
1	0	0	0	0
2	0	0	0	0
3	0	0	0	0
4	0	0	0	0
5	0	0	0	0
6	0	0	0	0
7	0	0	0	0
8	0	1	0	0
9	0	0	1	0
10	0	0	0	1
11	0	0	0	0
12	0	0	0	0
13	0	0	0	0
14	0	0	0	0
15	0	0	0	0
16	1	0	0	0
17	0	0	0	0
18	0	0	0	0
19	0	0	0	0
20	0	0	0	0
21	0	0	0	0
22	0	0	0	0
23	0	0	0	0
24	0	0	0	0
25	0	0	0	0
26	0	0	0	0

Table 4.12 Final discrimination result of Table 4.6 after optimization

Index number (scene\model)	1	2	3	4	5	6	7	8
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0
7	0	0	1	0	0	0	0	0
8	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0
12	0	0	0	1	0	0	0	0
13	0	0	0	0	1	0	0	0
14	0	0	0	0	0	1	0	0
15	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0
18	0	1	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0	0
22	0	0	0	0	0	0	0	0
23	1	0	0	0	0	0	0	0
24	0	0	0	0	0	0	0	0
25	0	0	0	0	0	0	0	0
26	0	0	0	0	0	0	0	0

4.5 Location Estimation

4.5.1 Coordinate Transformation

After identifying the model object from the scene using the 3-point matching method and the modified Hopfield neural network optimization, the location of the model object in the scene can be estimated by performing a coordinate transformation (Ansari, N. and Li, K. W., 1993) (Lee, J.S. *et al.*, 1997). In the discrimination experiment, the scale of the test and the model objects is not changed, so only rotation and translation are involved in the coordinate transformation. The location estimation of the occluded model object in the scene is by superimposing the model object to the scene and checking if it matches the identified object in the scene or not.

In the coordinate transformation, two correspondence feature point pairs from the model object and the scene is extracted and used to calculate the translation parameter (denoted as tx and ty) and the rotation parameter (denoted as θ). Suppose two feature points in the model object be denoted as P_{m1} (x_{m1}, y_{m1}) and P_{m2} (x_{m2}, y_{m2}), and their corresponding feature points in the scene object be denoted by P_{s1} (x_{s1}, y_{s1}) and P_{s2} (x_{s2}, y_{s2}), then P_{m1} and P_{m2} are mapped to P_{s1} and P_{s2} by the following equation:

$$\begin{bmatrix} x_{s1} \\ y_{s1} \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x_{m1} \\ y_{m1} \end{bmatrix} + \begin{bmatrix} tx \\ ty \end{bmatrix} \qquad \text{(Equation 4.27)}$$

The mapping of point P_{m2} *to* P_{s2} can be obtained from the same equation. In (Equation 4.27), the rotation angle θ is computed from the angles of the two line segments $P_{m1}P_{m2}$ and $P_{s1}P_{s2}$ with respect to the x axis:

$$\beta = Angle(P_{s1}P_{s2}) - Angle(P_{m1}P_{m2}) = \tan^{-1}(\frac{y_{s2}-y_{s1}}{x_{s2}-x_{s1}}) - \tan^{-1}(\frac{y_{m2}-y_{m1}}{x_{m2}-x_{m1}})$$
$$Angle(.) \in [0,\pi]$$

(Equation 4.28)

In order to cover the whole range of rotation from 0 to 2π, the rotation angle θ is obtained as:

$$\theta = \begin{cases} \beta + \pi & if \quad \Delta x_m \Delta x_s < 0 \\ \beta & otherwise \end{cases} \qquad \text{(Equation 4.29)}$$

Where

$$\Delta x_m = x_{m2} - x_{m1}, \quad \Delta x_s = x_{s2} - x_{s1} \qquad \text{(Equation 4.30)}$$

And the translation parameters (tx, ty) are calculated as following:

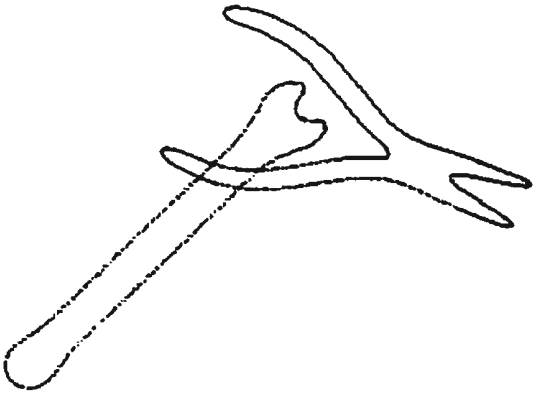
$$\begin{aligned} tx &= x_{s1} - (x_{m1}\cos\theta - y_{m1}\sin\theta) \\ ty &= y_{s1} - (x_{m1}\sin\theta + y_{m1}\cos\theta) \end{aligned} \qquad \text{(Equation 4.31)}$$

In the object discrimination, if two objects are matched with each other, all points in the model object and their corresponding points in the scene should satisfy the same equation for the coordinate transformation. Hence, the transformation parameters (tx, ty, θ) computed from two correspondence pairs satisfy the rest correspondence pairs. Figure 4.18 illustrates two recovered scene images from the model objects by the coordinate transformation.

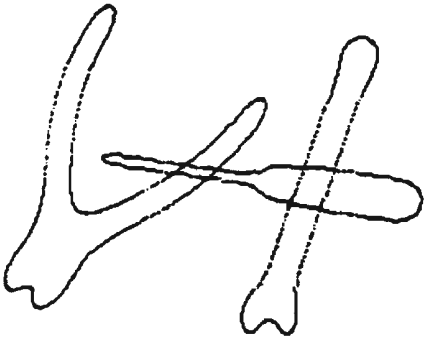
Ansari-Li (Ansari, N. and Li, K. W., 1993) and Lee (Lee, J.S. *et al.*, 1997) performed the location estimation by computing the distance between the model object and its recovered object in the scene. A score based on the distance was used to indicate the location difference between the two objects and regarded as a criterion to evaluate the discrimination accuracy. However, such a method has deficiencies. First, it is quite possible for the recovered boundary curve in the scene is not closed and an one-to-one correspondence between the points in the model and the recovered object cannot be obtained. Secondly, even if a model object is identified from the scene, its boundary curve usually does not exactly fit the recovered model object in the scene. The pixels of the identified object in the scene usually shift in curve smoothing, which may cause some error in calculating the parameters of the coordinate transformation.

The location estimation is used to eliminate the false corresponding pairs in this thesis. As stated above, all of the correspondence pairs between two matched objects satisfy the same transformation parameters (tx, ty, θ). If there exists a false matched pair, its transformation parameters will not conform to those of the rest pairs. This idea can be implemented to eliminate the false matched pairs in object discrimination (Nasrabadi, N.M. and Li, W., 1991).

4.5.2 Elimination of the False Matched Pair



(a) Recovered image of Figure 4.9



(b) Recovered image of Figure 4.12

Figure 4.18 Recovered images after coordinate transformation

As illustrated in Table 4.6-Table 4.12, although the Hopfield network enforces the constraint that each dominant point of the model object cannot match more than one dominant point of the scene and vice versa, it cannot guarantee a unique match in the final result. As discussed earlier, the correctly discriminated correspondence pairs satisfy the same transformation parameters (θ, tx, ty). Generally, a set of consecutive correspondence pairs (not less than 3) can be viewed as the reliable matched pairs and the parameters computed from these consecutive pairs can be regarded as the correct parameters for the coordinate transformation. Other correspondence pairs whose transformation parameters do not conform to the correct parameters may be regarded as the false matched pairs and eliminated. Figure 4.19 illustrates a model wire clipper and its overlapped image with a cutter, Table 4.13 lists the discrimination pairs which include a false matched pair *(4,8)*. To eliminate it, the transformation parameters for all the discriminated pairs are computed as shown in Table 4.14, which indicates *(4,8)* is a false matched pair because its parameters are too different from those of the rest ones. By eliminating the false matched pairs, the discrimination accuracy of the algorithm was improved.

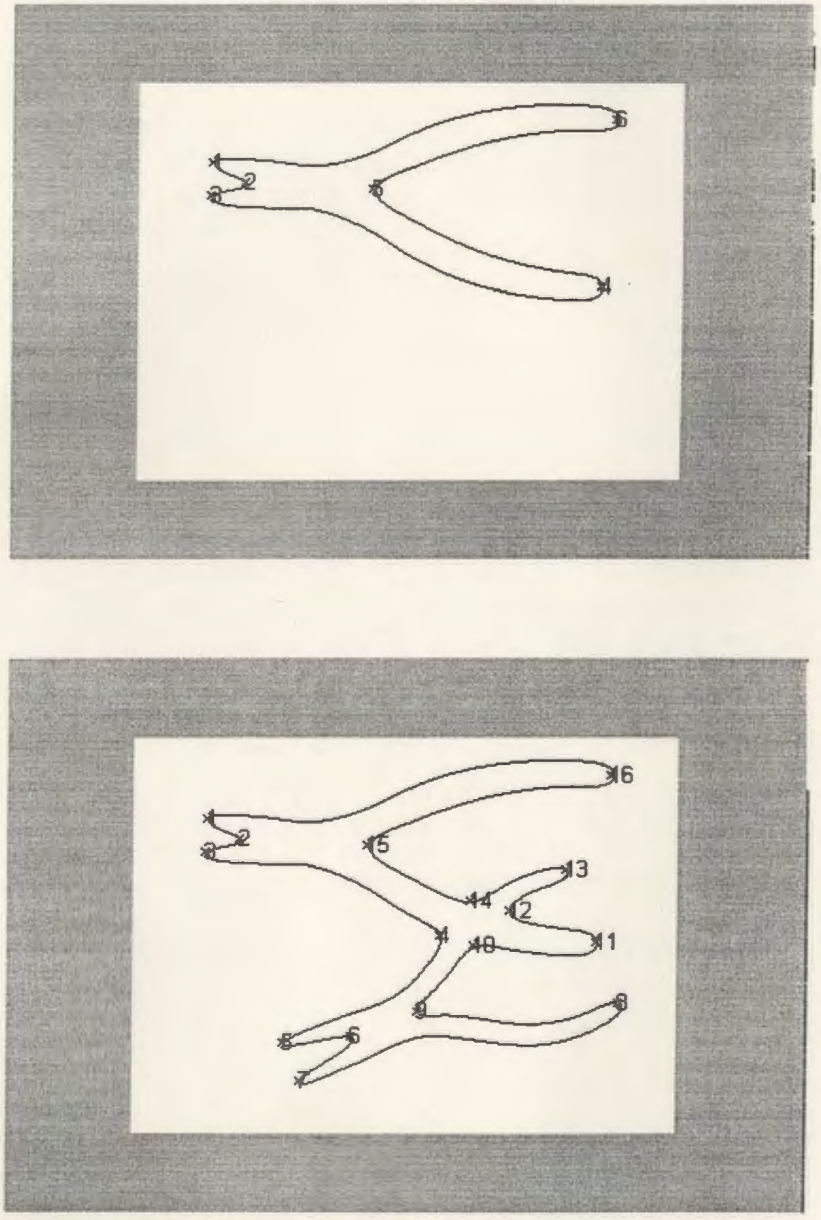


Figure 4.19 A model clipper and its overlapped image with a cutter

Table 4.13 Discrimination result for Figure 4.19

Index number (scene\model)	1	2	3	4	5	6
1	1	0	0	0	0	0
2	0	1	0	0	0	0
3	0	0	1	0	0	0
4	0	0	0	0	0	0
5	0	0	0	0	0	0
6	0	0	0	0	0	0
7	0	0	0	0	0	0
8	0	0	0	1	0	0
9	0	0	0	0	0	0
10	0	0	0	0	0	0
11	0	0	0	1	0	0
12	0	0	0	0	0	0
13	0	0	0	0	0	0
14	0	0	0	0	0	0
15	0	0	0	0	1	0
16	0	0	0	0	0	1

Table 4.14 False matched pair elimination by coordinate transformation

Matched pair	(tx, ty, θ)
(1, 1)	(1, 1, 0)
(2, 2)	(0, 1, 0)
(3, 3)	(1, 1, 0)
(4, 8)	(7, 81, 213)
(4, 11)	(0, 0, 0)
(5, 15)	(1, 0, 1)
(6, 16)	(1, 0, 0)

Chapter 5

Result and Discussion

5.1 Result

A model image bank, which includes 15 model screw drivers, 15 model cutters, 15 model wire clippers and 15 model wrenches, was built. The model objects were obtained by changing the rotation, translation and viewing distance (size) of the hand-tool objects when taking the photoes. 52 scene images consisting of the different overlapped combinations of the model objects were used as occluded samples to test the discrimination accuracy. To identify a model object in a scene image which consists of two or three overlapped objects, the scene image is compared with all of the model hand-tools in the image bank. For a scene image consisting of a cutter and a screw driver, if only one cutter in the model cutter bank is identified from the scene, then the

cutter in the scene can be regarded as a "detected" object. If the cutter in the scene cannot be retrieved after being compared with all the model cutters, it is viewed as a "missed" object. If the cutter or driver in the scene is falsely retrieved as an another model object (for example, the cutter is retrieved as a wire clipper), then it is regarded as a "false matched" object.

In the scene images, the overlapped objects can be classified into lightly, moderately and heavily occluded. The cutter in Figure 4.13, the wire clipper in Figure 4.14 and the screw driver in Figure 4.16 may be viewed as lightly, moderately and heavily occluded respectively. A successful discrimination object means that at least three consecutive dominant points (the number of the dominant points in the model object varies from 4 to 8) of the model object are identified from its occluded version in the overlapped image.

By the above standard, the discrimination result is illustrated in Table 5.1 as following:

Table 5.1: Discrimination result by comparing scene images with model image bank

	Real object in scene	Detected object	Missed object	Falsely detected object
Scene 1	Cutter, wrench	Both	No	No
Scene 2	Screw driver, cutter	Both	No	No
Scene 3	Wire clipper, cutter	Both	No	No
Scene 4	Wrench, cutter	Both	No	No
Scene 5	Screw driver, cutter	Both	No	No
Scene 6	Wire clipper, cutter	Both	No	No
Scene 7	Cutter, screw driver	Both	No	No
Scene 8	Wire clipper, screw driver	Both	No	No
Scene 9	Screw driver, cutter	Both	No	No
Scene 10	Cutter, wrench	Both	No	No
Scene 11	Cutter, wrench	Both	No	No

Scene 12	Wire clipper, screw driver	Both	No	No
Scene 13	Wire clipper, wrench	Both	No	No
Scene 14	Cutter, screw driver	Both	No	No
Scene 15	Wire clipper, screw driver, wrench	Screw driver, wire clipper	wrench	No
Scene 16	Wire clipper, wrench, screw driver	Screw driver, wrench	clipper	No
Scene 17	Wire clipper, wrench, screw driver	Three	No	No
Scene 18	Wire clipper, wrench, screw driver	Wire clipper	Wrench , driver	No
Scene 19	Cutter, wire clipper	Both	No	No
Scene 20	Cutter, wire clipper	Both	No	No
Scene 21	Cutter, screw driver	Both	No	No
Scene 22	Cutter, wire clipper	Both	No	No
Scene 23	Cutter, wrench, wire clipper	Wire clipper	Cutter, wrench	No
Scene 24	Cutter, wrench	Both	No	No
Scene 25	Cutter, wrench	Both	No	No
Scene 26	Wrench, screw driver	Wrench	Screw driver	No
Scene 27	Wrench, screw driver, wire clipper	Wire clipper, wrench	Screw driver	No
Scene 28	Screw driver, wire clipper, wrench	Three	No	No
Scene 29	Wire clipper, cutter	Both	No	No
Scene 30	Wrench, cutter	Both	No	No
Scene 31	Cutter, wire clipper	Both	No	No
Scene 32	Wrench, screw driver	Both	No	No
Scene 33	Wire clipper, cutter	Both	No	No
Scene 34	Wrench, screw driver	Both	No	No
Scene 35	Wire clipper. Screw driver	Wire clipper	wrench	No
Scene 36	Screw driver, wire clipper	Both	No	No
Scene 37	Two wrenches(small and big)	Both	No	No
Scene 38	Wire clipper, screw driver, wrench	Three	No	No
Scene 39	Wire clipper, screw driver	Both	No	No
Scene 40	Screw driver, wrench	Both	No	No
Scene 41	Wrench, cutter	Both	No	No
Scene 42	Wrench, cutter	Both	No	No
Scene 43	Screw driver, cutter	Both	No	No
Scene 44	Screw driver, cutter	Both	No	No
Scene 45	Wire clipper, cutter	Both	No	No
Scene 46	Screw driver, cutter, wire clipper	Cutter, wire clipper	Screw driver	No
Scene 47	Wire clipper, cutter, wrench	Three	No	No
Scene 48	Screw driver, wrench	Both	No	No
Scene 49	Screw driver, wrench	Screw driver	wrench	No
Scene 50	Screw driver, wrench	Both	No	No
Scene 51	Cutter, wrench, screw driver	Wrench, cutter	Screw driver	No
Scene 52	Cutter, wrench, screw driver	Three	No	No

The summation for the result is shown in Table 5.2:

Table 5.2 Summation of the discrimination result

Total scene images	52
Total objects in scene images	118
Discriminated scene images (all objects are discriminated)	42
Total discriminated objects in scene images	106
Percentage of discriminated scene image	81%
Percentage of discriminated object in scene image	90%
Percentage of falsely discriminated object	0%

The discrimination result shows that the proposed discrimination algorithm is successful. It almost discriminated out all of the lightly and moderately occluded objects, and could discriminate out some heavily occluded objects.

5.2 Problem and Discussion

5.2.1 Discrimination Rate

The discrimination accuracy is affected by the sample objects. Generally, more difference between the occluded objects in the scene, more opportunity for them to be discriminated. For example, if two objects are almost similar and overlapped together, the algorithm may encounter difficulty in determining their identities from the model image bank.

Another factor affecting the discrimination rate is the number of feature points extracted from the objects. More feature points will provide more information for the discrimination. For example, in the map discrimination (Ansari, N. and Li, K.W., 1993),

it is easy to extract many feature points from the map boundaries. Sometimes a map can provide 30-40 feature points after the curve smoothing. Consequently, when a model map is occluded in the scene, many consecutive feature points can still be obtained even more than half of the map is occluded. In the proposed experiment, the heavily occluded objects have less opportunity to be identified than the lightly and moderately occluded objects.

5.2.2 Robustness

Robustness is an important criterion to evaluate a discrimination algorithm and it was considered in the algorithm development of the thesis. As discussed in Chapter 4, when extracting the dominant points from the object boundary, the curvature estimation rather than the polygonal approximation was selected because the dominant points obtained by the curvature estimation are more stable under image rotation and translation. Experiments showed that the same dominant points were extracted under rotation and translation of the object (See Figure 4.10-Figure 4.12).

The algorithm was tested in the experiment by deliberately setting some variances between the model objects and their occluded versions in the scene. For example, the size and viewing angle of the hand-tools may be changed when they are put into the scene. Under such circumstances, the correct discrimination still can be obtained by adjusting the tolerance parameters in the 3-point matching process and the weighting factors and the structure parameter in the modified Hopfield network. However, since the algorithm is only rotation and translation independent (because

distance is considered in the Hopfield optimization), it is sensitive to the size change of the objects. In the discrimination test, some objects in the scene cannot be identified from the model image bank because their size is too different from that of the model objects in the image bank.

The coordinate transformation which was applied to eliminate the false correspondence pairs also improved the robustness of the algorithm.

5.2.3 Parameters

Almost every step of the discrimination algorithm is involved with parameters. The functions and problems of the parameters have been discussed in detail in Chapter 4. In the dominant point extraction, the curve smoothing parameter and the magnitude threshold parameter for removing the spurious dominant candidates were introduced. Experiments showed that the smoothing width chosen as 8 was optimal for most objects. A bigger width may eliminate some salient points, while a smaller width may not remove the small concavities and convexities. The magnitude threshold parameter for removing the spurious dominant point candidates is important for the final dominant point extraction. For example, by selecting different magnitudes, different dominants can be obtained (See the wrench in Figure 4.11 and Figure 4.13). The number of dominant points can be adjusted to meet specific requirement by changing the magnitude threshold.

The disparity matrix formation step is involved with two parameters, the length ratio threshold and the interior angle threshold. The length ratio may vary from *0.8~1.2*, while the angle threshold may vary from *5* to*15* in the experiment.

As discussed in the part of the modified Hopfield network, the weighting factors W_i and the structural parameter greatly affect the accuracy and computing time of the discrimination. During the optimization, the distances between the points are more emphasized, so W_3 is bigger than W_1 and W_2. Experiments showed that $W_1=W_2=0.25$ and $W_3=0.5$ were optimal for the hand-tool objects. The structural parameter was selected from *3* to *10* (the image size is *320×240*). The performance of the network is sensitive to these parameters. A small change in the weighting factors probably introduces several false correspondence pairs and a small change of the structural parameter may increase the computing time from several seconds to several minutes.

5.2.4 Algorithm Deficiencies and Future Improvement

Two steps in the discrimination algorithm may be improved in the future. One is the 3-point matching method for building the disparity matrix, another is the modified Hopfield neural network for the global optimization.

As discussed earlier, the 3-point matching method may cause problem under two circumstances. One is that two different objects have similar small portions, another is an overlapped version of a model object is divided into several parts in the scene. The correspondence feature point pairs obtained under both cases will form sporadic clusters in the disparity matrix, which is not reliable for an accurate discrimination. The same

problem happens when using the sphericity analysis (Appendix II) to perform the local comparison.

The Integrated Multiscale Feature (IMF) (Lee, J. S. *et al.*, 1997) (Parker, J. R., 1997) for extracting the object feature points may be an improvement for the 3-point matching method. The IMF uses the wavelet transformation to extract the feature points from the object boundary and express them in a heuristic form. By using the IMF, Lee (Lee, J. S. *et al.*, 1997) offered the following equation to compute the coupling symmetric weights between the neurons:

$$C_{ikjl} = W_1 F(\sum_{y=1}^{5} | f_{i,y} - f_{k,y} |) + W_2 F(\sum_{y=1}^{5} | f_{j,y} - f_{l,y} |) + W_3 F(| d_{i,j} - d_{k,l} |)$$

(Equation 5.1)

Where $f_{i,y}$ and $f_{j,y}$ are the *yth* elements of the IMF for the *ith* and the *jth* feature points of the model object, $f_{k,y}$ and $f_{l,y}$ are the *yth* elements of the IMF for the *kth* and *lth* feature points of the test object. The advantage of the above expression is that the threshold function $F(x,y)$ can be expressed in a ramp-like form (Figure 5.1 (b)), not in the step form which was used in the proposed algorithm. The ramp-like function used to describe the symmetric weights can provide a more stable matching in the optimization process (Lee, J. S. *et al.*, 1997). The wavelet transformation is very computationally extensive, but its fast computing algorithm may be used in extracting the object features.

The "local minima" problem is another problem which may be overcome in the future. In the discrimination experiment, the correct discrimination result cannot always

be obtained because the modified Hopfield network cannot guarantee to converge to the global minimum energy state. A Monto Carlo algorithm such as simulated annealing (Haykin, S., 1994) (Herault, L. *et al.*, 1990) may be used to overcome such a problem.

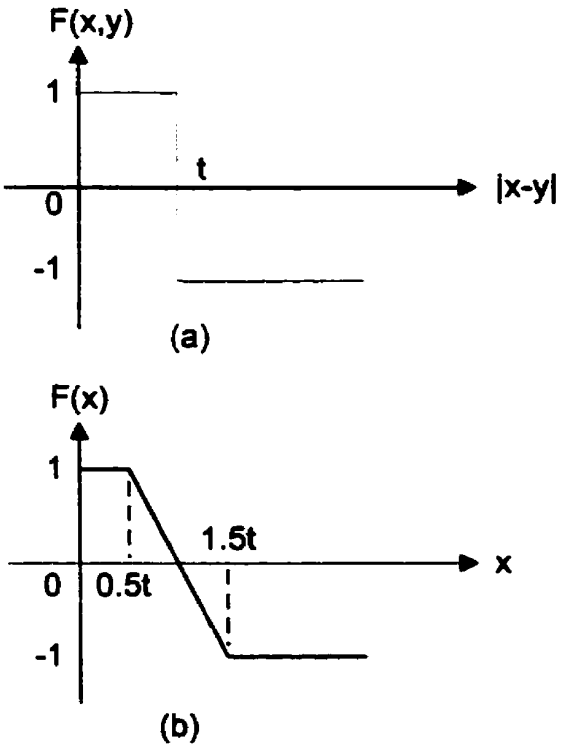


Figure 5.1 (a) A step function (b) A ramp-like function

Chapter 6

Conclusion

An algorithm applying a modified Hopfield neural network was designed for occluded object discrimination and a software package realizing the algorithm was developed with Unix C++. A model hand-tool image bank was built and 52 scene images containing overlapped model objects were compared with the image bank. The experiment result showed that the discrimination algorithm was successful.

The thesis reviewed the current shape representation and discrimination methods, including global and local shape descriptors, graphic isomorphism and their specific applications. A Gaussian function was applied to smooth the object boundary and the curvature estimation was used to extract the feature points of the objects. A 3-point matching method was implemented to build the disparity matrix for the initial comparison. A modified Hopfield neural network was used for the global optimization and the coordinate transformation was used to eliminate the false matched pairs. Finally, the parameters involved in the algorithm were discussed and suggestions for future improvement were presented.

Appendix I

Dynamic programming

A fundamental property of the dynamical programming is the "principle for optimality" (Dreyfus, S.E. and Law, A.M., 1977). The dynamic programming is usually used as a tool for the global optimization in image matching (Gorman, J.W., et al., 1988) (Ansari, N. and Delp, E.J., 1990) and voice recognition (Sakoe, H. and Chiba, S., 1978).

The optimization of the dynamic programming is based on an initial comparison. For example, in shape matching, the Fourier descriptor or the sphericity can be extracted as the feature property of the object. The feature point matching can be performed by comparing the properties of all of the feature points of the model object with those of the test object. The comparison can be clearly illustrate by building a disparity matrix whose row indexing number is the feature point number of the test object and column indexing number is that of the model object. An elements in the disparity matrix, *ijth*, indicates the measure between *ith* feature point of the test object and *jth* feature point of the model object. This measure is described by a value. For example, if the property is described by the sphericity, a value of 1 means a perfect match. However, from a global view, the above local matching method for identifying a model object from its occluded version in a scene is not good enough.

This problem may be overcome by the dynamic programming. The task for the dynamic programming is to find a shortest path which traverses through either all the rows or all the columns of the disparity matrix (the elements along this shortest path will be the optimized matched correspondence pairs). The traversing path should satisfy the following two constraints:

1) A feature point of the model object cannot match with more than one feature point of the test object.
2) A feature point of the test object cannot match with more than one feature point of the model object.

Obviously, the dynamic programming can be viewed as a global optimization approach. However, it produces desired paths when the locally optimal path is also globally optimal. The case that the locally optimal path is not globally optimal is still to be explored.

Appendix II

Sphericity under Triangle Transformation

The sphericity (Sonka, M. et al., 1998) is usually used in 2-D shape analysis and regarded as a similarity or dissimilarity measure method. In shape classification, after extracting the shape features from a model object and a test object. The similarity or dissimilarity measures are used to quantify the difference between the shape features.

As illustrated in Figure A2.1, a triangle has an inscribed ellipse, the sphericity is defined as *the ratio of the geometric mean* to *the arithmetic mean* of the lengths of the principal axes (denoted as d_1 and d_2) of the inscribed ellipse (Ansari, N. and Delp, E.J., 1990), that is:

$$S = \frac{2\sqrt{d_1 d_2}}{d_1 + d_2}$$

For two similar triangles, their sphericities are same. To explore the resemblance between two triangles based on the sphericity analysis, a triangle transformation is introduced

The triangular transformation is uniquely defined by an affine transform. The affine transform is the mapping of X to U (X, U belong to R^2), and defined as:

$$U = AX + T$$

Where

$$X = \begin{bmatrix} x \\ y \end{bmatrix}, \quad U = \begin{bmatrix} u \\ v \end{bmatrix}, \quad T = \begin{bmatrix} e \\ f \end{bmatrix}, \quad A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \text{ and } \det(A) \neq 0$$

Coefficients of the affine transform which maps one triangle into another, as shown in Figure A2.1, are computed using the following equation:

$$\begin{bmatrix} a \\ b \\ c \end{bmatrix} = B^{-1} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} c \\ d \\ f \end{bmatrix} = B^{-1} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}$$

Where

$$B = \begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{bmatrix}$$

(u_i, v_i) are the mapped points of (x_i, y_i), $i=1,2,3$. Since vertices of a triangle are noncollinear, $det(B) \neq 0$ and B^{-1} exists. The sphericity of the affine transform can be computed as follows:

$$S = \frac{t_1^2 + t_3^2 - (t_2^2 + t_4^2)}{t_1^2 + t_2^2 + t_3^2 + t_4^2}$$

$$t_1 = a + d, \quad t_2 = a - d, \quad t_3 = b - c, \quad t_4 = b + c$$

If two triangles are similar, the sphericity under the affine transformation will be 1. The less similar are the two triangles, the smaller is the value of the sphericity. The affine transformation is invariant under image rotation and translation. Since the sphericity can be used to measure the similarity between two sets of triplets, thus can be extended to measure the similarity between two objects.

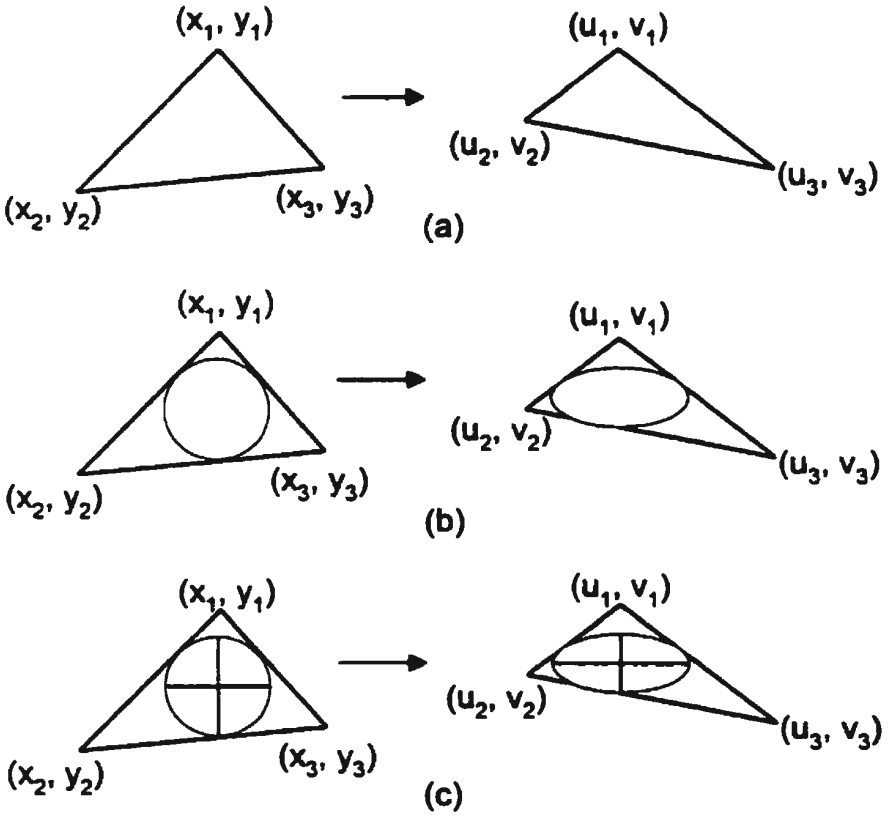


Figure A2.1 Sphericity under the affine transformation

Appendix IV

Travelling Salesperson Problem

The travelling Salesperson Problem (Schalkoff, R.J., 1992) is a good example for understanding the Hopfield neural network. The problem statement is for a travelling salesperson to travel n cities, how he can find an optimal route to travel all of the cities while achieving the shortest total travelling distance.

As shown in Figure A3.1, for n cities, $N=n^2$ nodes are required to represent the tour. The output will be V_{Xi}, where the subscript X refers to the city and i refers to the position in the tour. The tour begins and ends at the same city:

$$V_{X(n+1)} = V_{X1} \qquad and \qquad V_{X0} = V_{Xn}$$

The Liapunov function describing the problem should satisfy the following conditions:

1) Energy minimum must favor states that have each city once on the tour.
2) Energy minimum must favor states that have each position on the tour only once.
3) Energy minimum must favor states that include all n cities.
4) Energy minimum must favor states with shortest total distance.

So the energy function may be expressed as:

$$E = \frac{A}{2}\sum_{X=1}^{n}\sum_{i=1}^{n}\sum_{j=1,j\neq i}^{n} V_{Xi}V_{Xj} + \frac{B}{2}\sum_{i=1}^{n}\sum_{X=1}^{n}\sum_{Y=1,Y\neq i}^{n} V_{Xi}V_{Yj} + \frac{C}{2}(\sum_{X=1}^{n}\sum_{i=1}^{n}V_{Xi} - n)^2$$
$$+ \sum_{X=1}^{n}\sum_{Y=1,Y\neq X}^{n}\sum_{i=1}^{n} d_{XY}V_{Xi}(V_{Y,i+1} + V_{Y,i-1})$$

Where A, B, C and D are positive, E is non-negative. The meaning of each item in the energy function is as following:

1) First term is 0 iff there is a single $V_{xi}=1$ and all the other $V_{xi}=0$, which corresponds to that each city is only in the tour once.
2) Second term is 0 iff each column of the matrix contains a single value of 1, which corresponds to that each position on the tour has unique city associated.
3) Third term is 0 iff only n of the terms of the sum of V_{xi} that have value of 1, which corresponds to total number of "1" in the matrix is n.
4) The final term gives the total distance travelled, the last term also enforces the constrains:

$$V_{X(n+1)} = V_{X1} \qquad and \qquad V_{X0} = V_{Xn}$$

Where the connection between the nodes (X_i and Y_j) is:

$$T_{Xi,Yj} = -A\delta_{XY}(1-\delta_{ij}) - B\delta_{ij}(1-\delta_{XY}) - C - Dd_{XY}(\delta_{j,i+1} + \delta_{j,i-1})$$
$$\delta_{XY} = \begin{cases} 1 & X = Y \\ 0 & X \neq Y \end{cases}, \quad \delta_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$$

The first item is the lateral inhibition on the rows in the matrix with weight $-A$, the second item is the lateral inhibition on the columns with weight $-B$, the third item is the global inhibition with weight $-C$ and the forth item is the inhibition on the two neighboring columns and the weight $-D$ is in proportion of the distance between the two corresponding cities.

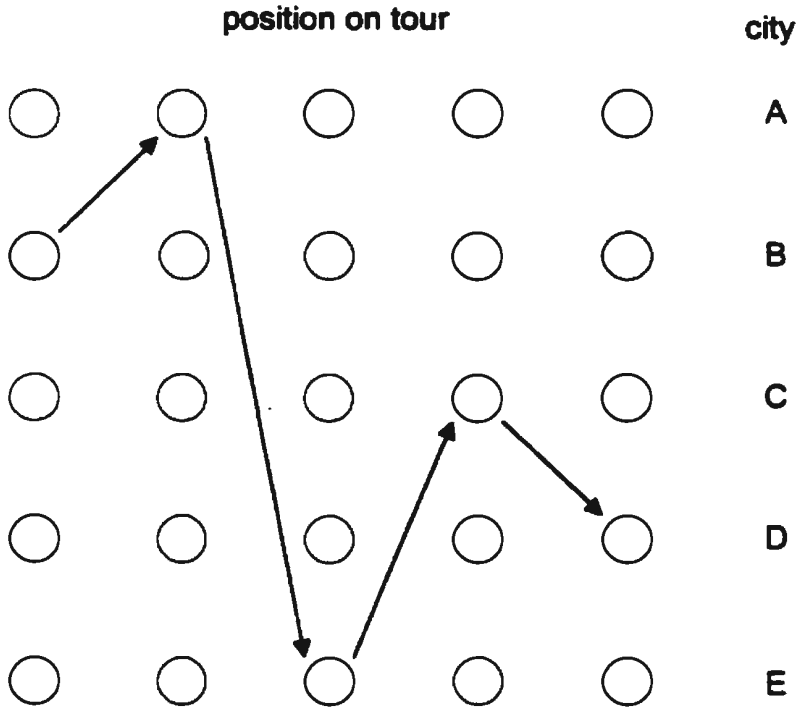


Figure A3.1 A 5-city example of travelling salesperson problem

References

Aas, K., Eikvil, L., and Huseby, R.B. (1999). "Applications of Hidden Markov Chains in Image Analysis," Pattern Recognition, Vol. 32, no. 4, pp.703-713.

Adler, S.L., and Krishnan, R. (1998). "Similarity and Affine Normalization of Partially Occluded Planar Curves Using First and Second Derivatives," *Pattern Recognition*, Vol. 31, no.10, pp. 1551-1556.

Ansari, N., and Delp, E.J. (1990). "Partial Shape Recognition: A landmark-based Approach," *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol. 12, no. 5, pp. 470-483.

Ansari, N., and Delp, E.J. (1991). "On Detection Dominant Points," *Pattern Recognition*, Vol. 24, no. 5, pp. 441-451.

Ansari, N., and Li, K.W. (1993). "Landmark-based Shape Recognition by a Modified Hopfield Neural Network," *Pattern Recognition*, Vol. 26, no. 4, pp. 531-542.

Asada, H., and Brady, M. (1986). "The Curvature Primal Sketch," *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol. PAMI-8, no. 1, pp. 34-43.

Attneave, F. (1954). "Some Informational Aspect of Visual Perception," *Psych. Rev.*, 61, pp. 183-193.

Ayache, N., and Faugeras, O.D. (1986). "Hyper: A New Approach for the Recognition and Position of Two-Dimensional Objects," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-8, no. 1, pp. 44-54.

Ballard, D.H. (1981). "Generalizing the Hough Transform to Detect Arbitrary Shape," *Pattern Recognition*, Vol. 13, pp.111-122.

Bassmann, H., and Besslich, P.W. (1992). *Ad Oculos: Digital Image Processing, Student Version 2.0*, Thomson Publishing Inc., Berlin.

Beatty, D.A. (1993). "2-D Contour Shape Analysis for Automatic Herring Roe Quality Grading by Computer Vision," *Master thesis of Computer Science*, University of British Columbia, Vancouver, British Columbia, Canada.

Brzakovic, D., Luo, X.M., and Brzakovic, P. (1990). "An Approach to Automated Detection of Tumors in Mammograms," *IEEE Transactions on Medical Imaging*, Vol. 9, no. 3, pp. 233-241.

Borgefors, G. (1988). "Hierarchical Chamfer Matching: A Parametric Edge Matching Algorithm," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 10, no. 6, pp. 849-865.

Canny, J.F. (1986). "A Computational Approach to Edge Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*," Vol. PAMI-8, no.6, pp. 679-698.

Cantoni, V., Cinque, L., Guerra, C., Levialdi., and Lombardi, L. (1998). "2-D Object Recognition by Multiscale Tree Matching," *Pattern Recognition*, Vol. 31, no.10, pp. 1443-1454.

Chung, P.C., Tsai, C.T., Chen, E.L., and Sun, Y.N. (1994). "Polygonal Approximation Using a Competitive Hopfield Neural Network," *Pattern Recognition*, Vol. 27, no.11, pp. 1505-1512.

Dubois, S.R. and Glanz, F.H. (1986). "An Autoregression Model Approach to Two-Dimensional Shape Classification," *IEEE Transaction on Pattern Recognition and Machine Intelligence*, Vol. PAMI-8, no. 1, pp. 55-65.

Dreyfus, S.E. and Law, A.M. (1977). *The Art and Theory of Dynamical Programming*, Academic Press, New York.

Freeman, H. (1961). "On the Encoding of Arbitrary Geometric Configuration," *IRE Transactions on Electronic Computers*, EC-10 (2), pp. 260-268.

Fukushima, K. (1988) "Neocognition: A hierarchical Neural Network Capable of Visual Pattern Recognition," *Neural Networks* 1, pp. 119-130.

Gamage, L.B., Gosine, R.G., and de Silva, C.W. (1996). "Extraction of Rules from Natural Objects for Automated Mechanical Processing," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, Vol. 26, no. 1, pp. 105-120.

Gorman, J.W., Mitchell, O.R., and Kuhl, F.P. (1988). "Partial Shape Recognition Using Dynamic Programming," *IEEE Transactions on Pattern Analysis and Machine Intelligence.* Vol. 10, no.2, pp. 257-266.

Gupta, L., Sayeh, M.R., and Tammana, R. (1990). "A Neural Network Approach to Robust Shape Classification," *Pattern Recognition,* Vol. 23, no. 6, pp. 563-568.

Han, M.H., and Jang, D.S. (1990). "The Use of Maximum Curvature Points for the Recognition of Partially Occluded Objects," *Pattern Recognition*, Vol. 23, no. ½, pp. 21-23.

Haykin, S. (1994). *Neural Networks: A Comprehensive Foundation,* IEEE Computer Society Press, Macmillan College Publishing Company, Inc., New York.

He, Y., and Kundu, A. (1991). "2-D Shape Classification Using Hidden Markov Model," *IEEE Transactions on Pattern Analysis and Machine Intelligence.* Vol. 13, no. 11, pp.1172-1183.

Heinemann, P.H., Morrow, C.T., and Sommer, H.J. (1996). "An Automated Inspection Station for Machine Vision Grading of Potatoes," *Machine Vision and Application*, pp. 225-231.

Hopfield. J.J. (1982). "Neural Networks and Physical Systems with Emergent Collective Computational Abilities," *Proceeding of the National Academy of Sciences of U.S.A.*79, pp. 2554-2558.

Hopfield, J.J., and Tank, D.W. (1985). "Neural Computation of Decisions in Optimization Problems," *Biological Cybernetics,* Vol. 52, pp. 141-152.

Howarth, M.S., and Searcy, S.W. (1989*)* "Algorithm for Grading of Carrots by Machine Vision," *ASAE Paper*, No. 89-7502, St. Joseph, Mich.

Hu, B.G., Gosine, R.G., Cao, L.X., and de Silva, C.W. (1998). "Application of a Fuzzy Classification Technique in Computer Grading of Fish Products," *IEEE Journal of Fuzzy System,* 6(1), pp. 144-152.

Hu, M.K. (1962). "Visual Pattern Recognition by Moment Invariants," *IRE Transactions on Information Theory*, Vol. 8, pp. 179-187.

Jiang, X.Y., and Bunke, H. (1991). "Simple and Fast Computation of Moments," *Pattern Recognition*, Vol. 24, no. 8, pp. 801-806.

Kim, J.H., Yoon, S.H., and Sohn, K.H. (1996). "A Robust Boundary-based Object

Recognition in Occluded Environment by Hybrid Hopfield Neural Networks, " *Pattern Recognition*, Vol. 29, no. 12, pp. 2047-2060.

Koch, M.W., and Kashyap, R.L. (1987). "Using Polygons to Recognize and Locate Partially Occluded Objects," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. PAMI-9, no. 4, pp. 483-494.

Lamdan, Y., Schwartz, J.T., and Wolfson, H.J. (1990). "Affine Invariant Model-based Object Recognition," *IEEE Transactions on Robotics and Automation*, Vol. 6, no. 5, pp. 578-589.

Lee, S.H., Chen, C.H., Sun, Y.N., and Tseng, G.S.(1997). "Occluded Objects Recognition Using Multiscale Features and Hopfield Neural Network," *Pattern Recognition*, Vol.30, no.1, pp. 113-122.

Lee, J.S., Sun, Y.N., Chen, C.H., and Tsai, C.T. (1993). "Wavelet Based Corner Detection," *Pattern Recognition*, Vol.26, no.6, pp. 853-865.

Li, B.C., and Shen, J. (1991). "Fast Computation of Moment Invariants," *Pattern Recognition*, Vol. 24, no. 8, pp.807-813.

Li, B.C., and Shen, J. (1994). "Two-Dimensional Local Moment, Surface Fitting and Their Fast Computation," *Pattern Recognition*, Vol. 27, no. 6, pp. 785-790.

Liu, H.C., and Srinath, M.D. (1990). "Partial Shape Classification Using Contour Matching in Distance Transformation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 12, no. 11, pp. 1072-1079.

Liu, X.F., Tan, S.H., Srinivasan, V., Ong, S.H., and Xie, W.X. (1994). "Fuzzy Pyramid-based Invariant Object Recognition," *Pattern Recognition*, Vol. 27, no. 5, pp. 741-756.

Loncaric, S. (1998). "A Survey of Shape Analysis Techniques," *Pattern Recognition*, Vol. 31, no. 8, pp. 983-1001.

Maes, M. (1991). "Polygonal Shape Recognition Using String-Matching Techniques," *Pattern Recognition*, Vol. 24, no. 5, pp. 433-440.

Marshall, S. (1989). "Review of Shape Coding Technique," *Image and Vision Computing*, Vol. 7, no. 4, pp. 281-294.

McClure, J.E., and Morrow, C.T. (1987). "Computer Vision Sorting of Potatoes", *ASAE Paper*, No. 87-6501, St. Joseph, Mich.

Meulders, S.(1980). "Some shape parameters for cell recognition," *Proc. Pattern Recognition in Practice Workshop North-Holland*, The Netherlands, pp. 131-142.

Mitaias, D.A., and Mertzios, B.G. (1994). "Shape Recognition with a Neural Classifier Based on a Fast Polygon Approximation Technique," *Pattern Recognition*, Vol. 27, no. 5, pp. 627-636.

Mokhtarian, F., and Mackworth, A. (1986). "Scale-based Description and Recognition of Planar Curves and Two-Dimensional Shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol. PAMI-8, no.1, pp. 34-43.

Mott-Smith, J. (1970). *Medial Axis Transformations, in Picture Processing and Psychopictories*, Lipkin and Rosenfeld, eds, pp. 267-283, Academic Press, New York.

Nasrabadi, N.M., and Li, W. (1991). "Object Recognition by a Hopfield Neural Network," *IEEE Transaction on Systems, Man, and Cybernetics*, Vol. 21, no. 6, pp.1523-1535.

Parker, J.R. (1997). *Algorithms for Image Processing and Computer Vision*, Wiley Inc., New York.

Pavlidis, T., and Horowitz, S.L. (1974). "Segmentation of plane curves," *IEEE Trans. Comput.* C-23, pp. 860-870.

Persoon, E., and Fu, K.S. (1977). "Shape Discrimination Using Fourier Descriptors," *IEEE Transaction on System, Man, and Cybernetics*, Vol. SMC-7, no. 3, pp.170-179.

Pitas, I., and Venetsanopoulos, A.N. (1992). "Morphological Shape Representation," *Pattern Recognition*, Vol. 25, pp. 555-565.

Ramer, U. (1972). "An Iterative Procedure for the Polygonal Approximation of Plane Closed Curves," *Comput. Graphics Image Process.* 1, pp. 244-256.

Ray, B.K., and Ray, K.S. (1997). "Scale-Space Analysis and Corner Detection on Digital Curves Using a Discrete Scale-Space Kernel", *Pattern Recognition*, Vol. 30, no. 9, pp.1463-1474.

Russ, J.C. (1995). *The Image Processing Handbook (Second Edition)*, CRC Press Inc., Boca Raton.

Sakoe, H., and Chiba, S. (1978). "Dynamic Programming Algorithm Optimization for

Spoken Word Recognition," *IEEE Transactions on Acoustic, Speech and Signal Processing*, Vol. ASSP-26.

Salari, E., and Balaji, S. (1991). "Recognition for Partially Occluded Objects Using B-Spline Representation," *Pattern Recognition*, Vol. 24, no. 7, pp.653-660.

Schalkoff, R. (1992). *Pattern Recognition: Statistical, structural and Neural Approaches*, John Wiley & Sons, Inc., New York.

Shen, D.G., and Ip, H.H.S. (1999). "Discriminative Wavelet Shape Descriptors for Recognition of 2-D Patterns," *Pattern Recognition*, Vol. 32, no. 2, pp. 151-165.

Sonka, M., Hlavac, V., and Boyle, P. (1998). *Image Processing, Analysis, and machine vision (Second Edition)*, Brooks/Core Publishing Company.

Tao, Y., Morrow, C.T., Heinemann, P.H., and Sommer, H.J. (1995). "Fourier-Based Separation Technique for Shape Grading of Potatoes Using Machine Vision," *Transaction of ASAE*, Vol. 38(3), pp. 949-957.

Tsang, P.W.M., Yuen, P.C., and Lam, F.K. (1992). "Recognition of Occluded Objects," *Pattern Recognition*, Vol. 25, no. 10, pp. 1107-1117.

Tsoukalas, L.H., and Uhric, R.E. (1997). *Fuzzy and Neural Approaches in Engineering*, John Wiley and Sons, Inc., New York.

Wang. S.S., Chen, P.C., and Lin, W.G. (1994). "Invariant Pattern Recognition by Moment Fourier Descriptor," *Pattern Recognition*, Vol. 27, no. 12, pp. 1735-1742.

Wen, W., and Lozzi, A. (1992). "Recognition and Inspection of Two-Dimensional Industrial Parts Using Subpolygons," *Pattern Recognition*, Vol. 25, no. 12, pp. 1427-1434.

Wolfson, H.J. (1990). "On Curving Matching," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 12, no. 5, pp. 483-489.

Yoon. S.H., Kim, J.H., Alexander, W.E., Park, S.M., and Sohn.K.H. (1998). "An Optimum Solution For Scale-Invariant Object Recognition Based On the Multiresolution Approximation," *Pattern Recognition*, Vol. 31, no.7, pp. 889-908.

Zahn, C.T., and Roskies, R.Z. (1972). "Fourier Descriptors for Plane Closed Curves," *IEEE Transactions on Computers*, Vol. C-21, no. 3, pp. 269-281.