Tina Memo No. 1996-003
Internal.

Tutorial: Algorithms For 2-Dimensional Object Recognition.

A.Ashbrook and N.A.Thacker.

Last updated
1 / 12 / 1998

Imaging Science and Biomedical Engineering Division,
Medical School, University of Manchester,
Stopford Building, Oxford Road,
Manchester, M13 9PT.

1 Abstract

Representation of arbitrary shape for purposes of visual recognition is an unsolved problem. The task of representation is intimately constrained by the recognition process and one cannot be solved without some solution for the other. We have already done some work on the use of an associative neural network system for hierarchal pattern recognition of the sort that may be ultimately useful for generic object recognition. The conclusions from this work were that

- Networks can be constructed which are robust to noisy and missing data.
- The input to the network should preferably be significance measures of homogenous features.
- The required invariance properties must be explicit in the form of input representation.

We restrict here the recognition task to pre-segmented rigid bodies in the hope that a solution for this case will suggest ways of solving the more general case. Areas which are relevant to the extension of this work will be identified where possible.

2 Invariant Representations.

The main idea behind generating invariant representations of shape is to obtain a compact set of descriptions sufficient to describe all relevant views of an object. It should also be considered as a way of building principled generalisation mechanisms into learning systems. Therefore, a representation which is not compact but gives good generalisation properties for a trained system would still be considered useful. The invariance properties that we believe to be important for general vision are translation, rotation and scale. This would prevent a learning system from having to discover all of the various manifestations of a rigid object in its vision module. This argument will be valid even for arbitrary occluding boundaries when there may be no true 3D invariance to rotation except in the image plane.

There are caveats to these invariance requirements which make the solution of this problem non-trivial. The first of these is descriptive power, the representation must have sufficient descriptive power to allow discrimination between all dissimilar objects, theoretically up to the differences due to the required invariance properties. If we find that such a representation cannot be constructed this does not mean that representational object recognition is impossible. An alternative method would be to generate several representations with the required invariance properties and combine them in a composite system which is capable of the discrimination task (in much the same way that we believe the brain actually solves this problem). This approach we will leave for the time being while we concentrate on the performance of the fundamental zeroth order system. The representation must be one which can be applied to arbitrary shapes if it is to be considered a useful step towards the goal of generic vision. Finally, the representation needs to be robust to the types of noise expected from the input vision data. For most machine vision systems we are referring here to missing data fragmented representations and extraneous data from other objects.

But what do we use as our source of data? We could try using the grey level image directly, but remember we wish to construct homogenous representations which are measures of evidence for the existence of an object, a grey level image does not have this property. Clearly we should use features we can reconstruct at all rotations translations and scale. A good representation is expected to be one based on local shape, non-local representations are not robust to addition and loss of data. A sensible choice is oriented line segments, for several reasons. Lines are reasonably invariant to object rotation, they are reliably reconstructed at the same position and with the same relative orientation to the object regardless of the absolute rotation of the object and details of illumination. This fact is borne out by the success of feature based stereo matching algorithms. Secondly, our own visual system is thought to generate oriented line segments as a fundamental part of image preprocessing, and thirdly (somewhat conveniently) this is what our computer vision system delivers. We make a distinction here between 2D data obtained directly from individual images and 3D data obtained from a stereo vision system. The latter will clearly have complete invariance to 3D rotation for particular groups of features describing one aspect, while the former although only coping with rotations in the image plane is perhaps closer to human visual recognition.

A variety of different algorithms have been developed to perform 2-dimensional object recognition, utilizing many different types of features and matching methods. For the purpose of this report it has not been practical to

consider them all in detail but it is hoped that the selection which follows in this section conveys all of the important principles used and that any other algorithms are simply variations on a theme.

3 Block Matching

In block matching schemes the occurrence of an object within a scene is identified by correlating a template image of the object with the given scene image and looking for a significant peak. One possible correlation function is defined in equation 7.1 below.

$$c(x,y) = \sum_{j=0}^{N-1} \sum_{i=0}^{N-1} [t(i,j) - s(x+i, y+j)]^2 \tag{1}$$

If a grey level instance of an object is used to form the template then it can be correlated with the scene image directly but the match will be sensitive to variations in lighting. This matching sensitivity can be avoided by forming templates from silhouetted instances of objects and correlating these with binary threshold images although in practise this only transfers the lighting variation problem to the thresholding stage. A more principled method to improve the robustness of block matching to lighting variation is to correlate edge enhanced object templates with edge enhanced scene images.

Block matching has no inherent invariance characteristics. To recognise objects of different scales and orientations a different template is required for each instance, whilst the problem of positional variability is avoided by the virtue that the correlation function is applied across the whole image. Another feature of the correlation function is its robustness to small levels of random noise. This robustness provides some tolerance when matching with noisy scene images.

4 Shape Skeletons

The skeleton of a binary shape is obtained by repeatedly thinning until it becomes a unit pixel width network, see figure 7.1.

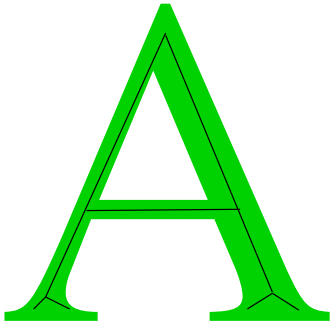

Figure 1: A binary shape and its skeleton

The philosophy behind the skeleton approach to object recognition is that most of the information about a particular shape is contained within its topology. Consider the case of hand-drawn characters where the character is not identified by the thickness of the pen strokes but by the interconnection of the strokes. Object recognition is achieved using shape skeletons by extracting shape topologies from an image and finding a match between this and any of the set of training set topologies.

Thinning strategies generally work on the principle of stripping away successive layers of shape boundary points on the condition that the removal of a point does not change the connectedness of the shape. When all allowable points have been removed the shape skeleton is left. An efficient thinning algorithm is presented by Xia [16]. In this scheme the shape to be thinned is imagined to be an area of grass and the boundary of this area is set alight. As the fire propagates across the grass, fire-fronts begin to meet and the fire at these points becomes extinguished. This set of points defines the shapes skeleton.

Topological networks posses all of the invariant properties required of a general vision system although for most applications a purely topological description is too ambiguous and some structural constraints have to be added. These structural constraints may well compromise the invariance properties of the description.

Thinning algorithms are fairly robust to random noise although in its presence may produce shorts spurs, but these can be removed from the skeleton. By their nature, thinning algorithms are sensitive to occlusion (to a thinning algorithm an occluded shape looks like a different shape with different topology) and in general recognition schemes based on skeletons cannot cope with occluded schemes.

5 Moment Invariants

The use of moments as invariant binary shape representations was first proposed by Hu in 1961 [10]. Hu successfully used this technique to classify handwritten characters.

The regular moment of a shape in an M by N binary image is defined as:

$$u_{pq} = \sum_{j=0}^{N-1} \sum_{i=0}^{N-1} i^p j^q f(i,j) \tag{2}$$

Where $f(x,y)$ is the intensity of the pixel (either 1 or 0) at the coordinates (x,y) and $p+q$ is said to be the order of the moment.

Because the calculation is a function of the distance between shape pixels and the origin measurements are taken relative to the shapes centroid (x',y') to remove translational variability. The coordinates of the centroid are determined using the equation above:

$$i = \frac{u_{10}}{u_{00}} \qquad \text{and} \qquad j = \frac{u_{01}}{u_{00}} \tag{3}$$

Relative moments are then calculated using the equation for central moments which is defined as:

$$u_{pq} = \sum_{j=0}^{N-1} \sum_{i=0}^{N-1} (i-i)^p (j-j)^q f(i,j) \tag{4}$$

Individual moments values do not have the descriptive power to uniquely represent arbitrary shapes, nor do they posses the required invariance characteristics, but, sets of functions based on these moments can be determined which do. Hu derived a set of seven rotational invariant moment functions which form a suitable shape representation (or vector).

$$M_1 = (u_{20} + u_{02}) \tag{5}$$

$$M_2 = (u_{20} - u_{02})^2 + 4u_{11}^2 \tag{6}$$

$$M_3 = (u_{30} - 3u_{12})^2 + (3u_{21} - u_{30})^2 \tag{7}$$

$$M_4 = (u_{30} + u_{12})^2 + (u_{21} + u_{03})^2 \tag{8}$$

$$\begin{aligned} M_5 &= (u_{30} - 3u_{12})(u_{30} + u_{12}) \left((u_{30} + u_{12})^2 - 3(u_{21} + u_{03})^2\right) \\ &+ (3u_{21} - u_{03})(u_{21} + u_{03}) \left(3(u_{30} + u_{12})^2 - (u_{21} + u_{03})^2\right) \end{aligned} \tag{9}$$

$$M_6 = (u_{20} - u_{02}) \left((u_{30} + u_{12})^2 - (u_{21} + u_{03})^2\right) + 4u_{11}(u_{30} + 3u_{12})(u_{21} + u_{03}) \tag{10}$$

$$
\begin{aligned}
M_7 &= (3u_{21} - u_{03})(u_{30} + u_{12})\left((u_{30} + u_{12})^2 - 3(u_{21} + u_{03})^2\right) \\
&- (u_{30} - 3u_{12})(u_{21} + u_{03})\left(3(u_{30} + u_{12})^2 - (u_{21} + u_{03})^2\right) \qquad (11)
\end{aligned}
$$

Classification is achieved by matching a shape vector extracted from an image with previously encountered shape vectors from the training set. The shape representation can be improved to exhibit scale invariance by a process of normalization.

The moments described above can be calculated either from a segmented binary image or from a shapes's boundary. Jiang and Bunke [11] show that the two different calculations are mathematically equivalent although Dudani et al [8] suggest that the binary image calculation is less susceptible to noise.

Moment invariants do not inherently possess translational invariance and this variability is removed by centering the coordinate system on a shapes centroid. Unfortunately, moment calculations are sensitive to the position of a shapes centroid and attempts to determine this are marred by random noise, poor segmentation and occlusion. Hence, moment invariant schemes are not robust to these types of problem.

6 Log-Polar Mapping

An invariant shape representation can be formed using the log-polar mapping. This scheme maps points in the image space to points in the log-polar parameter space. Consider a point z in the image space where $z = x + yj$. The log-polar mapping will map to a point w in the parameter space, where:

$$w = ln(z) \qquad (12)$$

$$w = ln(|z|) + j\theta_z \qquad (13)$$

The invariant nature of the mapping arises because changes in scale and orientation in the image space are mapped to translations in the parameter space. The mapping of scale to translation can easily be demonstrated using the one dimensional case:

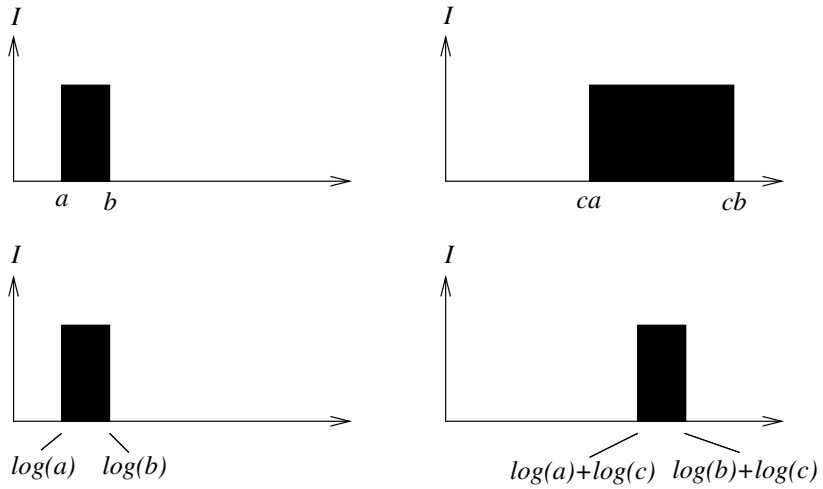

Figure 2: Features when scaled in the image space become translated in the *log* space

In the figure above, the features a and b which are scaled by a factor c in image space are shifted by a factor $ln(c)$ in parameter space.

Although strictly speaking the log-polar mapping does not introduce scale and rotation invariance, by simplifying these modes of variation to translation the invariance can more easily be achieved. Both Wechsler et al [15] and Rak et al [12] take the Fourier Transform of the log-polar space and use its magnitude as an invariant measure. This works because the magnitude of the Fourier Transform is invariant to translation.

Because the representation is not intrinsically translation invariant the shape being analyzed first has to be moved to the centre of view. This is achieved in the literature by determining the centroid of the shape and moving

the origin to that point. Wechsler et al identify the problem that small variations in located centroid result in dramatic variations in the resulting log-polar representation. Unfortunately, random noise and occlusion produce such variations.

Another problem with the log-polar mapping is that because of the singularity at $log(0)$, objects lying on the origin in practice tend to become stretched rather than translated as they rotate and scale. Rak et al attempt to avoid this problem by using edge enhanced images and hoping that no edges lie on the origin. Wechsler et al edge enhance the image after the log-polar mapping which has a similar effect.

The nature of the mapping is that many samples are taken at the centre of the image where the radial lines are closer together but the resolution falls off moving out from the centre. This has two consequences. Firstly the method is really only suitable for objects which are significantly smaller than the image size so that the resolution of the representation is sufficiently high. Secondly, because outlying objects have only a minor effect on the representation because of the low sampling, the object under analysis does not have to be segmented out from the image (providing that it can be centered correctly).

7 Geometric Shape Descriptors

Given an arbitrary shape, either in the form of a silhouette or a boundary, there are a variety of simple geometric measures which can be used to construct a representation for that shape. Examples of such measures include width, height, perimeter, area and so on. The shape descriptor is constructed by concatenating a predetermined number of these measures to from a feature vector. Object recognition is achieved by comparing training set feature vectors with feature vectors extracted from a scene.

Before any measurements can be taken the shape under analysis must be located. This removes any translational variability but requires a suitable segmentation scheme. Other invariance properties depend upon the types of measurements used to construct the feature vector, for example perimeter and area measurements are invariant to rotation, ratios of widths or lengths are invariant to scale. In general, however, a suitably descriptive feature vector which is constructed from a number of different measures will not possess any invariance as a whole and scene variability must be dealt with by normalization and shape alignment techniques. Unfortunately, the need for reliable normalization and alignment considerably weakens the method's robustness to noise and occlusion.

Strachan et al [14] use a simple geometric shape descriptor to automatically classify different fish species. Their descriptor consists of eleven length measurements taken from fish profiles, see Figure 7.3 below.

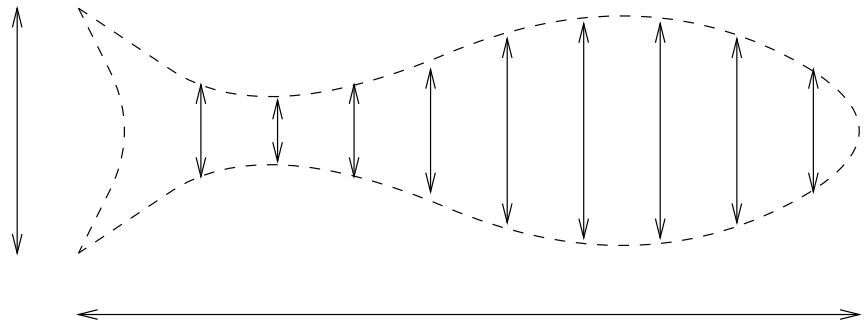

Figure 3: A simple geometric shape descriptor for classifying fish species consisting of eleven length measurements

These measures are the fishes total length, its total width and nine other widths taken at equal spacings along it length. Invariance was achieved in this scheme by scaling the fishes area to units and aligning its principle axis along the x-axis. Using this technique a 90

8 Boundary Profiles

To simplify the matching process shape boundaries can be extracted from the two dimensional image domain can described as a one dimensional boundary profile [6].

8.1 (r, θ) Plots

One possible profile is obtained by describing a boundary in polar coordinates relative to its centroid. Each point along the boundary is defined by its distance from the centroid, r, and its angular displacement around the boundary from some arbitrary reference, θ.

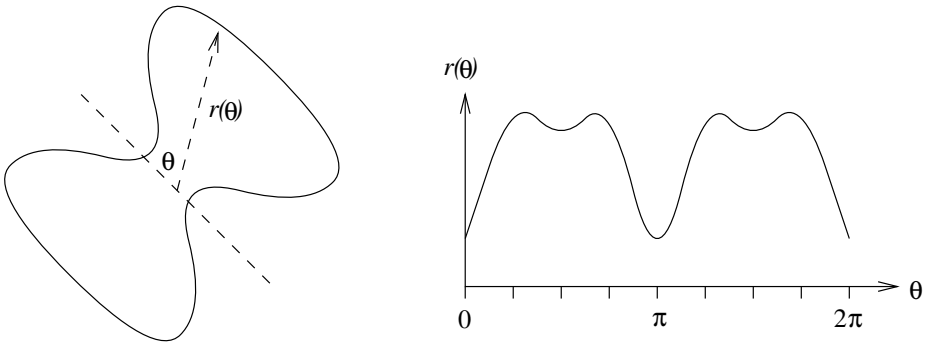

Figure 4: A two dimensional shape and its (r, θ) plot

Recognition is simply achieved by matching model profiles from the training set with boundary profiles extracted from a scene. If the objects within the scene can be at arbitrary orientations then their profiles will be arbitrarily shifted along the theta axis relative to the model profiles. Matching these objects is achieved by sliding their profiles along the model profiles and finding a significant fit.

A particular problem with the (r, θ) plot is that for all but the most simple shapes, the profile becomes multivalued, that is, for some values of theta their may exist a number of different values of r. This means that the profile is no longer one dimensional and the matching one again becomes a two dimensional problem. It has been suggested that multiple r values can be avoided by discarding all but either the smallest or the largest r values at each value of but this significantly reduces the descriptive power of the representation and leads to ambiguities.

Like all representations which rely on centroid measurements (r, θ) plots become globally distorted in the presence of occlusion and matching the profile to its corresponding model becomes impossible.

8.2 (s, ψ) Plots

The (s, ψ) plot is an alternative boundary profile which is not multivalued and distorts locally in the presence of occlusion so that matching can still be achieved. The plot is started at any arbitrary position along the boundary and the tangential orientation of each boundary point, ψ, is plotted against the distance travelled along the boundary, s.

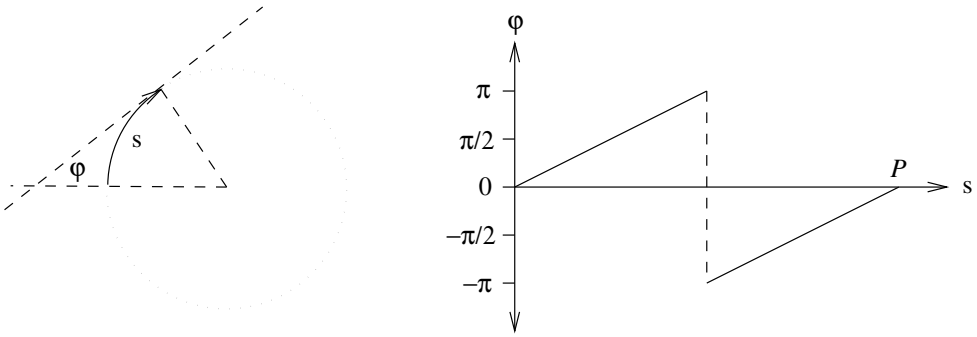

Figure 5: A two dimensional shape and its (s, ψ) plot. P is the perimeter of the boundary

The effect of occlusion on an (s, ψ) profile is to *insert* boundary information associated with the occluding object into the original profile at the point where the occlusion begins whilst leaving the profile either side of the occlusion untouched. This is demonstrated in figure 7.6.

Recognition can now be achieved in the presence of occlusion by matching sections of the model profiles with sections of the occluded object profiles.

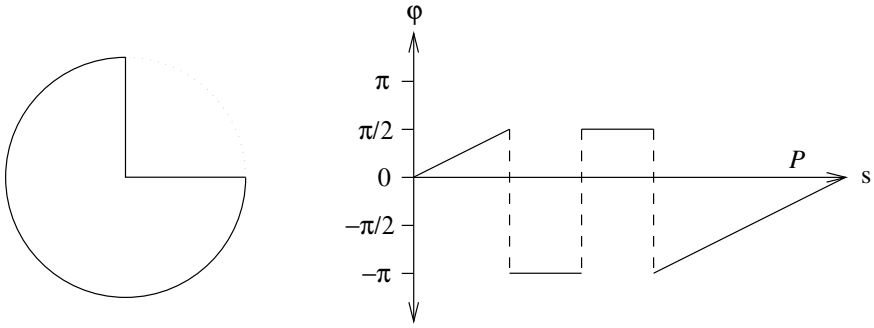

Figure 6: The square section occluding the cirlcle is shown as a dotted line. Note that the plot of the unoccluded section of the circle either side of the square section has not been distorted

A significant problem with the (s, ψ) profile is accurately determining the distance travelled along the boundary as this measure is sensitive to the way the boundary is quantised in the scene.

Neither of the profiles considered above intrinsically provide scale invariance but this can be achieved, to a degree, by normalization. For the (r, θ) plot, the values of r are not scale invariant but can be normalized to the maximum value of r encountered. For the (s, ψ) plot, the arc length s is not scale invariant but this can be normalized to the total arc length, the perimeter, P. Shape representations based on boundaries are robust to random noise and lighting variation (because edge detectors are robust to these types of noise) but suffer from the boundary segmentation problem discussed earlier.

9 Fourier Descriptors

The use of Fourier Descriptors as a representation for closed curves was first suggested by Cosgriff in 1960 [4]. Initially a curve is plotted as tangential orientation against arc length (as described in the last section on boundary profiles). The resulting one dimensional boundary profile is then normalized to a length of 2 and then expanded as a Fourier Series using the Fourier expansion:

$$c_n = \frac{1}{N} \sum_{s=0}^{N} \Psi(s) e^{-j2\pi ns} \tag{14}$$

The boundary is now uniquely represented by the infinite series of Fourier coefficients, cn. The lower order terms of this series describe the macroscopic behavior of the boundary curve while the higher order terms describe the curves detailed microscopic behavior. In practice this series can be truncated to give a finite shape descriptor whilst still retaining sufficient descriptive power. Zahn et al [12] successfully use the first ten Fourier coefficients to represent hand-written numerals. Recognition is achieved by finding matches between Fourier descriptors extracted from scenes and training set descriptors.

Like boundary profiles, Fourier descriptors do not possess translational invariance and the respective shapes need to be located by segmentation. Although scale invariance is inherent in the representation it only arises because of the necessity to normalize the boundary length and is not an intrinsic property of the boundary. Fourier descriptors do, however, possess rotational invariance.

A significant problem with Fourier descriptors is that each Fourier coefficient is calculated from, and therefore sensitive to, every boundary point. The result is that the representations behaves very poorly in the presence of occlusion or any distortion of the boundary. Fortunately, boundary edge detection is robust in the presence of random noise and lighting variation.

10 Active Shape Models

Cootes et al [5] present an active shape representation based on a point distribution model. The term 'active' is used as the representation incorporates information about how the shape was seen to deform in the training set.

The point distribution model defines a two dimensional shape as a set of labelled points along its boundary or any other significant edge features. The points are defined by their mean positions plus the main modes of variation

which describe how the points move away from their means as the shape deforms.

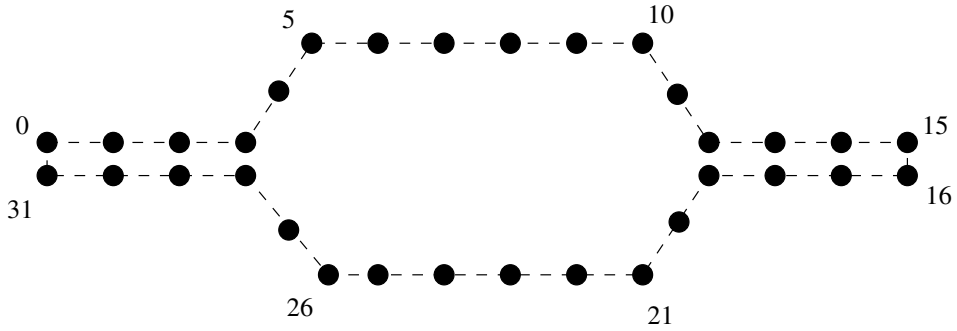

Figure 7: The point distribution model of a resistor defined by 32 points

Recognition is achieved by first estimating the pose and scale of an object within a scene, positioning the point distribution model accordingly and then adjusting the model points until they best fit the underlying scene. If the fit can be refined until it is sufficiently close then the object has been detected. By constraining the adjustments of the model points to the modes of variation encountered in the training set the best fit is found without deforming the shape beyond a degree that might be expected.

After the shape model has been positioned over the scene, adjustments are determined for each point by looking for and then moving towards the strongest edge along the boundary normal at each point. After the set of adjustments has been determined they are resolved into a global pose and scale adjustment plus residual point adjustments which result in the shape being deformed. It is this deformation which is constrained by the modes of variation encountered in the training set. The process is repeated until no significant changes occur.

If the initial pose and scale estimate is weak then points may be attracted to edges of outlying objects within the scene and the initial guess will not be refined. An attempt to rectify this has been proposed where rather than moving model points towards the strongest edge along the boundary normal, the points are moved towards edges which exhibit intensity profiles similar to those found in the test data. To achieve this the representation must incorporate the edge intensity profiles encountered at each point in the training set.

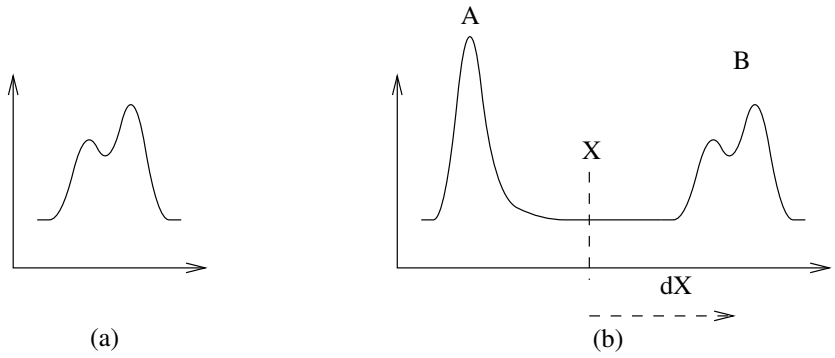

Figure 8: Point X is associated with the edge intensity profile shown in (a). (b) shows the edge intensity profile along the normal to the boundary at point X in a real scene. Traditionally X would have moved in the wrong direction to the closer and stronger edge at A but instead uses the intensity profile to see that the correct edge is infact B

The use of active shape models in object recognition is severely limited by the necessity to initially estimate the pose and scale of objects within a scene accurately. If this can be done then the recognition problem has already been solved. Despite this lack of invariance, active snake models do perform very well in the presence of random noise, lighting variation and occlusion because of the global constraints. The scheme can be used to hypothesize whether objects are present at a given location within a scene and to refine the pose and scale estimates if they are present. Consider the inspection of a complex assembly. If the assembly can be aligned at a known position within the scene, this scheme can be used to look for the different components of the assembly in order to determine whether they are in place and correctly oriented.

11 The Hough Transform

The Hough Transform was first devised in 1962 as a means of detecting the paths of high energy particles [9]. It has since evolved and has been applied to many different image processing applications.

The Hough Transform technique works by transforming complex patterns of pixels in the image domain into compact features in a chosen parameter space. The transformation operates such that many points in the image space map to single points in the parameter space. This means that searching for complex patterns of pixels is simplified by working in the parameter space and that the technique is robust to some loss of data due to occlusion and other sources of noise.

11.1 The Straight Line Hough Transform

Any line in an image can be described by the slope-intercept equation and appears as a single point in (m, c) space:

$$y = mx + c \tag{15}$$

By transforming from image space to (m, c) space the problem of line detection is simplified to a problem of point detection. In practise the transformation has to be applied to each individual point in the image which transform into lines in the parameter space, however, collinear points in the image map to intersecting lines in the parameter space. If votes are accumulated for points plotted in the parameter space then the intersection will be seen as a point of local maxima. It follows that lines in the image are found by searching for points of local maxima in the parameter space.

A modification has to be made to this technique to cope with vertical or near vertical lines for which the size of the parameter space becomes infinite. This is achieved by using two parameter spaces, one in (m, c) space and the other in (m', c') space, where:

$$m' = \frac{1}{m} \tag{16}$$

$$x = m'y + c \tag{17}$$

Peaks in the first space represent lines with gradients less than one. Peaks in the second space represent lines with gradients greater than or equal to one.

The main problem with this technique is the large amount of computation required. This arises because for every point in the image a whole line has to be plotted in two parameter spaces and then these spaces need to be searched for points of local maxima.The technique was improved by Duda et al in 1972 [7] by replacing the slope-intercept equation by the normal equation:

$$P = x\cos\theta + y\sin\theta \tag{18}$$

Where P is the length of the normal from the line being detected to the origin and θ is the angle between the normal and the positive x-axis.

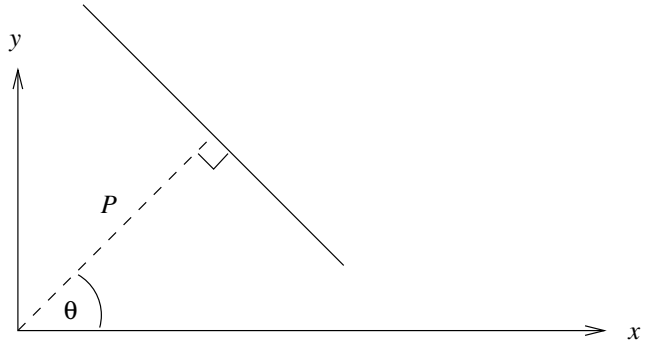

Figure 9: A line in image space defined by the two parameters P and θ

A point in image space now appears as a sinusoid in (p, θ) space and collinear points in the image appear as intersecting sinusoids. The advantage of this method is that only one parameter space is required so the storage and computational requirement is reduced.

Further improvements have been made to reduce the computational load. Rather than transforming every point in image space into a sinusoid in parameter space, the image can be first searched for short line segments and these used instead. Because the orientation of the segments can be determined only a single point for each has to be plotted in parameter space. Also, rather than using a single two dimensional parameter space, two one dimensional parameter spaces can be used instead. Initially a histogram of line segment orientations is built and significant peaks identified. The line segments associated with these peaks are then plotted in p space and peaks in this space correspond to lines in the images.

It is worth noting that the straight line Hough Transform identifies the presence of lines in an image but the description is that of an infinitely long line passing through the image. To determine the actual position and length of a line which has been detected requires further processing on the image.

11.2 Circle Detection Using The Hough Transform

The Hough Transform can be used to detect circles within an image. In this scheme the parameter space is congruent with the image space, that is, each point in the image maps to a point in the same position in the parameter space. To detect a circle of radius R circles of this radius are plotted in parameter space centered on edge segments found in the image. Peaks in the parameter space indicate the centre of detected circles.

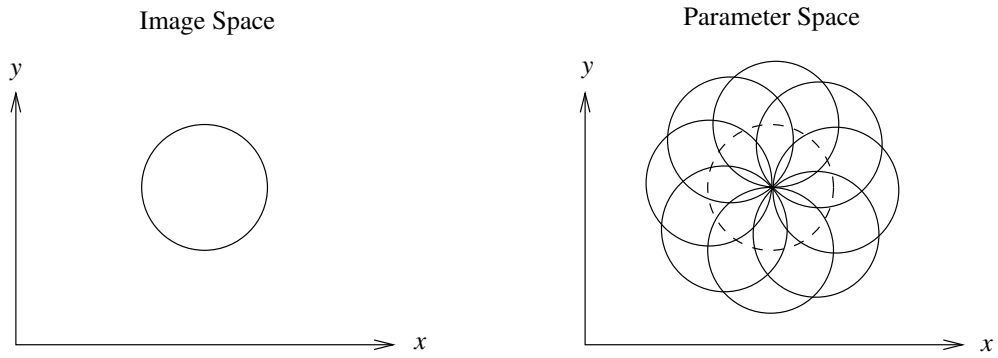

Figure 10: The location of a circle in the image space is indicated by a peak of votes in the parameter space

This technique is particularly computationally intensive as a whole circle has to be plotted in parameter space for every edge found in image space. A significant saving is made by determining the orientation of the edge segments and plotting a single point in parameter space a distance R along the normal to the edges. If rather than plotting a single point, all points between a distance R0 and R1 along the normal are plotted then circles with a radius between $R0$ and $R1$ will be detected, although another stage of computation is then required to determine the specific radius of the detected circles.

11.3 The Generalized Hough Transform

The Hough Transform techniques for line and circle detection discussed in the last section are very useful in object recognition but many recognition tasks require a method for detecting arbitrary shapes. The Generalized Hough Transform has been developed to do this (Ballard [1]).

The Generalized Hough Transform is generalization of the Hough circle detection technique. Rather than plotting a point in parameter space at a fixed distance R along the normal to the current edge segment, a point is plotted at a variable distance $R(\theta)$ along a line which is angularly displaced from the normal by a variable angle $a(\theta)$, where θ is the angle between the normal and the positive x-axis. An arbitrary shape is described by defining $R(\theta)$ and $a(\theta)$ for all θ in a table (usually referred to as the R-Table).

If the shape described by the R-Table is present in the image then each edge segment of the shape will contribute to a point L in parameter space and a large vote will be accumulated.

There is a complication with this approach. If the arbitrary shape contains straight edges or concavities then some value of θ will be associated with several values of $R(\theta)$ and $a(\theta)$.

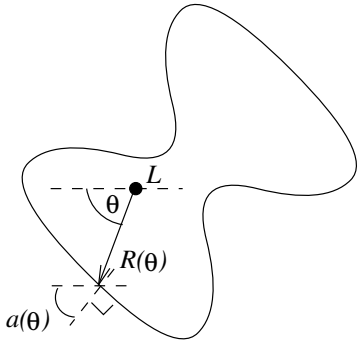

Figure 11: The values R and a are recorded for each value of θ, the orientation of the boundary normal. This set of values defines the shape boundary relative to the point L

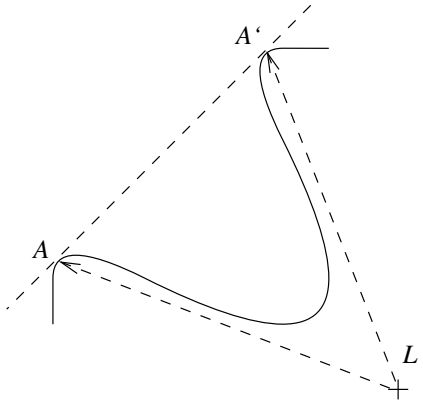

Figure 12: The orientation of the boundary at points A and A' are identical but the values of R and a which define the position of L are different

This problem is overcome by storing multiple values in the R-Table. Although for values of θ where multiple entries are stored several points will be plotted in the parameter space, the largest accumulation of points will still occur at L.

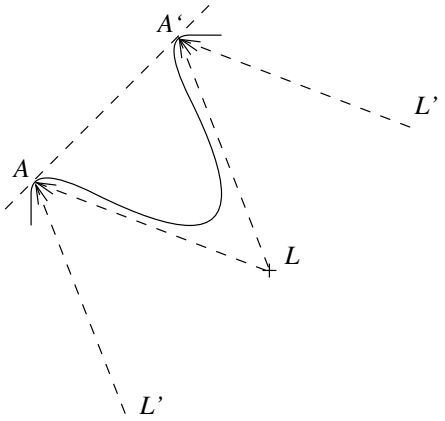

Figure 13: Because the boundary points A and A' have the same orientation two sets of values for R and a are recorded in the R-Table. However, when these values are used to accumulate votes in parameter space the largest peak still occurs at L

The generalized hough transform possess translational invariance but not scale or rotational invariance. To detect shapes at different orientations and scale an explicit search must be made for each instance by applying a suitably transformed R-table. The table is transformed for scale variations by simply scaling the R values accordingly whilst it is transformed for orientation variations by shifting the R values along the axis. The result is a four dimensional parameter space with two axis for image position, an axis for orientation and an axis for scale. Significant peaks

in this four dimensional space then indicate the presence of an object along with its position, orientation and scale within a scene.

12 2D Projective Invariants

A shape representation for two dimensional planar objects which is invariant to changes in perspective as well as changes in pose and scale is presented by Rothwell et al [13]. This shape representation is then used in a planar object recognition strategy.

The representation relies upon the fact that points of tangency on a two dimensional planar object are preserved under different projections and also that the mapping of any four points from one plane to another is sufficient to determine the transformation matrix T which fully defines that transformation. Consequently, by mapping four points of tangency from a planar object to four fixed but arbitrary points in a second plane, this second plane will possess the required invariant properties, and by determining the transformation matrix T from the four mappings all points on the planar object can be mapped onto the invariant plane.

In the literature, planar object concavities are used to determine four tangency points, referred to as distinguishing points. See figure 3q below.

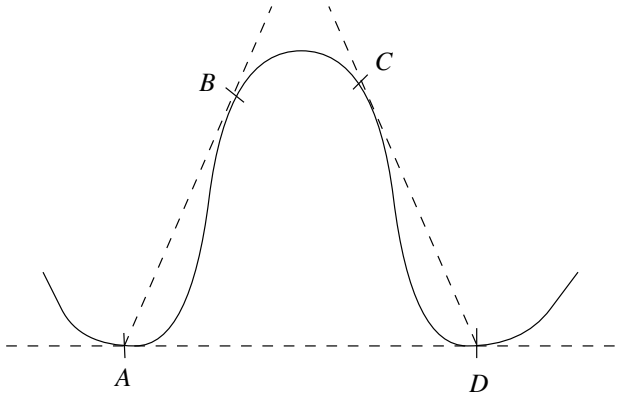

Figure 14: Two distinguishing points (A and D) are located by the bitangent that marks the entrance to the concavity. The other two distinguishing points (B and C) are located by the tangents to the inner curve of the concavity which pass through each of the first two distinguishing points

Two distinguishing points (A and D) are located by the bitangent that marks the entrance to the concavity. The other two distinguishing points (B and C) are located by the tangents to the inner curve of the concavity which pass through each of the first two distinguishing points.

These four points are then mapped to the corners of a unit square on the invariant plane, which is referred to as the canonical plane, and then the same transformation is used to map all other boundary points within the concavity onto this plane. The mapping of the concavity shown in figure 3q to the canonical plane is shown in figure 3r below.

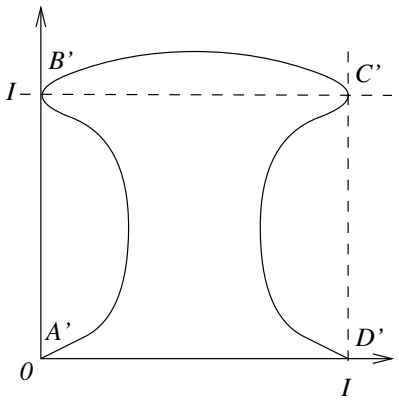

Figure 15: The mapping of the concavity in Figure 3q onto the canonical plane

Rothwell et al use this projective invariant to perform object recognition. They do this by projecting planar objects onto the invariant plane and then taking a number of area and moment measurements which then constitute invariant feature vectors. These features vectors are then used to build a hash table during training and to address the hash table during recognition. Shapes which are addressed during recognition are used as hypothesis for the presence of those shapes, and each hypothesis is verified by projecting each shape back into the image and looking for a significant overlap.

Because the representation is formed from local concavities, planar shapes with a number of concavities are placed in the hash table a number of times. This redundancy gives the representation some robustness to occlusion but the requirement for concavities restricts the representation from being used on arbitrary shapes. By utilizing shape edges the representation is robust to random noise and lighting variation. In real applications of object recognition, shapes are very rarely planar so an extension of this approach into three dimensions is desirable; Rothwell et al demonstrate that the representation loses its invariance even for small deviations from planararity. Unfortunately it has been proved by Burns et al [3] that no such projective invariants can exist for the three dimensional case.

13 Conclusions.

The above attempts to recognise objects can be considered as typical in machine vision taskss. Often the requirements of an application are such that many assumptions can be made regarding the image under analysis. However, in unconstrained environments, recognition systems must deal with a variety of factors, and ultimately be capable of selecting one (or a few candidates) from amongst a large variety of possible shapes. No local representation of shape can be completely invariant to loss of data, but what can be done is to ensure that the small change in input data results in a small change in representation. Continuous smooth variation in representation is necessary if we are to develop systems that cope with occlusions and are robust to the segmentation process. The representation must also have sufficient scope to deal with arbitrary shapes, even those which cannot be extracted as continuous boundaries.

The issue of shape segmentation is of obvious importance, if we assume that object shapes come to us pre-segmented from a scene then subsequent recognition is almost trivial and the simplest approaches (measuring total length of lines etc.) will have some capabilities. If on the other hand we assume that we do not know which parts of an image are from a specific object the problem is almost impossible. Prior knowledge of the shape is clearly of some relevance to the recognition problem, in the papers which follow we explain an approach to shape recogntion which uses knowledge of learned shape in order to simultaneously solve the segmentation and recognition problem. This is done in the context of an analysis of what characteristics a generic representation should posess and consideration of the statistical matching process. The technique we propose we call Paiwise Geometric Histogrammes (PGH). It can be considered as a version of "shape context", yet it directly addresses the issue of representational completeness and predates these ideas by almost a decade.

References

[1] Ballard, D. H., "Generalizing the Hough Transform to Detect Arbitary Shapes", Pattern Recognition, Vol. 13, pp. 111-122, 1981.

[2] Ballard, D. H. and Brown, C., "Computer Vision", Prentice Hall, 1982.

[3] Burns, J. B., Weiss, R. S. and Riseman, E. M., "View Variation of Point-Set and Line- Segment Features", IEEE trans. Pattern Analysis and Machine Intelligence, Vol. 15, No. 1, pp. 51-68, 1993

[4] Cosgriff, R. L., "Identification of Shape", Ohio State University Research Foundation, Columbus, Rep. 820-11, ASTIA AD 254 792, 1960.

[5] Cootes, T. F. and Taylor, C. J., "Active Shape Models - 'Smart Snakes'", Department of Medical Biophysics, University of Manchester, 1992.

[6] Davies, E. R., "Machine Vision - Theory, Algorithms, Practicalities", Academic Press, 1990.

[7] Duda, R. O. and Hart, P. E., "Use of the Hough Transformation to Detect Lines and Curves in Pictures", Comm. ACM, Vol. 15, pp. 11-15, 1972.

[8] Dudani, S. A., Breeding, K. J. and McGhee, R. B., "Aircraft Identification by Moment Invariants", IEEE trans. Computing, Vol. 26, No. 1, pp. 39-45.

[9] Hough, P. V. C., "Method and Means for Recognizing Complex Patterns", US Patent 30696 54.

[10] Hu, M. K., "Visual Pattern Recognition by Moment Invariants", IRE trans. Information Theory, Vol. 8, pp. 179-187.

[11] Jiang, X. Y. and Bunke, H., "Simple and Fast Computation of Moments", Pattern Recognition, Vol. 24, No. 8, pp. 801-806, 1991.

[12] Rak, S. J. and Kolodzy, P. J., "Performance of a Neural Network Based 3-D Object Recognition System", SPIE Automated Object Recognition, Vol. 1471, pp. 177-184.

[13] Rothwell, C. A., Zisserman, A., Forsyth, D. A. and Mundy, J. L., "Canonical Frames for Planar Object Recognition", Proc. ECCV, Santa Margherita, Italy, pp. 757-772, 1992.

[14] Strachan, N. J. C., Nesvadba, P. and Allen, A. R., "Fish Species Recognition by Shape Analysis of Images", Pattern Recognition, Vol. 23, No. 5, pp. 539-544, 1990.

[15] Wechsler, H. and Zimmerman, G. L., "2D Invariant Object Recognition Using Distributed Associative Memory", IEEE trans. Pattern Analysis and Machine Intelligence, Vol. 10, No. 6, pp. 811-821, 1988.

[16] Xia, Y., "Skeletonization Via the Realization of the Fire Front's Propagation and Extinction in Digital Binary Shapes", IEEE trans. Pattern Analysis and Machine Intelligence, Vol. 11 No. 10, pp. 1077-1086, 1989.