Image Primitives

Jason M. Kinser Bioinformatics and Computational Biology George Mason University Manassas, VA 20110 USA

Abstract: Towards the construction of a content based image recognition system it is important to isolate and also represent image segments. There have been several proposed methods of representing a shape but many are not sensitive to deviations of the appearance of an object. Furthermore, it is noted that the set of possible shapes is not distributed equally within a representation space. Thus, liberating the requirement that a basis set be orthonormal is justified. A method is presented here that extracts a basis set from an image database and defines shapes from this basis set. The results indicate that describing shapes from this basis set is robust to alterations of the shape such as small occlusions, limited skew, and limited range.

Keywords: Segmentation, clustering, shapes

1. Justification

Isolating the inherent shapes within an image is important for many applications including content based image recognition. This is not an easy task for several reasons including illumination gradients, object occlusions, shadows, and perspective. However, the isolation of a segment shape is only part of the required task. This shape still needs to be represented in a format that is conducive for high volume fast comparisons.

Several researchers have proposed methods for describing shapes [1-5] However, there are requirements that need to be met for an representation to be effective with real images. Consider a case in which the same object is shown in two images and the shape of this object onto the 2D image plane is isolated. There are still alterations that are often experienced between these two shapes. These include perspective and occlusion, and, in real applications the actual isolation of a segment is difficult due to illumination gradients, detector calibrations, and the presence of shadows. Thus, the same object may produce two shapes with some differences.

The differences in these two shapes quite often are not uniform. In other words, a good portion of the shape perimeters are similar and the differences appear only in one small part of the shape. Consider two shapes *A* and *B* which are similar except for one small portion (perhaps *B* has a small occlusion). Any method that relies on the centroid of the two shapes to mathematically describe the two shapes is inherently at a disadvantage since the centroids of the two shapes will not be in the same location.

Other methods that describe shape rely on the decomposition of the shape by a set of basis functions. Mathematically, there is an argument for making this set of functions an orthonormal set. For example, the image can be replicated using the decomposition coefficients and the basis set. However, in practice this causes problems. The set of shapes found within images is most certainly not distributed equally in R^{N} space (where N is the number of pixels). Thus, some of the basis functions are not useful and others are overworked. While an orthonormal basis function can reproduce shapes it may not be the best representation for the purposes of comparing shapes.

Thus, this study offers a different approach. Segments from several images were collected and clustered in a fashion the matches perimeter similarity as opposed to centroid based systems. These clusters (some of which may be purged as explained later) form the basis set for representing a shape. This basis set is not orthonormal but is instead quite sensitive to the idiosyncratic nature of the distribution of shapes. Thus, shapes that are somewhat common can be delineated with the same efficiency as shapes that are quite dissimilar.

2. System Design

There are three major components to this work of which the first one is not addressed in detail here. Discussion of its optimization is beyond the scope of this presentation. The three major steps are segment isolation, clustering, and generation of the descriptive vectors. Segment isolation is a major task in itself and no attempt will be made here to claim that the best method has been employed. This work is more concerned with the manipulation of the segments once they have been isolated.

The clustering step will accumulate into groups segments that have similar shape. This process uses the original shapes and relies heavily on matching the shapes according to size and perimeter. The final step is to represent a shape in terms of these clusters. [6]

2.1. Segment Isolation

As stated, this is a difficult task that warrants a separate research effort. Thus, a method was chosen here based on previous work and no attempt to optimize this step is considered here. Object isolation from grey-scale images has been successfully demonstrated through the use of cortical modeling. [7-11] The model [12] receives an input **S** and for each pixel establishes a neuron with a potential F, a threshold Θ , and an output state Y. The process iterates allowing the neurons to cycle through a few pulsations,

$$\mathbf{F}[n+1] = f\mathbf{F}[n] + \mathbf{S} + W\{\mathbf{Y}\},\tag{1}$$

$$Y_{ij}[n+1] = \begin{cases} 1 & if \ F_{ij}[n+1] > \Theta_{ij}[n] \\ 0 & Otherwise \end{cases}, (2)$$

$$\Theta[n+1] = g\Theta[n] + h\mathbf{Y}[n+1], \qquad (3)$$

where f, g, and h are constants. The neural activity of each iteration is captured in the arrays **Y**[n]. These are called *pulse images* and due to neuron synchronization the segments inherent in the image cause their associated neurons to fire in unison. Thus, image segments become readily available. The only caveat is that there may be more than one segment present in an iteration of the neural network thus there is a simple step to extract disconnected segments into individual frames.

2.2. Clustering

A typical image produces between 5 and 15 isolated segments that are of sufficient size. In this step of the process groups are created that contain similar segments. This system is based on a k-means clustering algorithm but there are major modifications required.

2.2.1. Similarity Method

Before a clustering algorithm can be employed there must first be a definition of how two entities are compared. In this case, how are two segments compared for similarity?

These segments are binary and solid (interior pixels are 1, exterior pixels are 0, and the edges are sharply defined). Thus, the pertinent information is contained in size and the perimeter of the shape. The perimeter is more than edge information in that it includes the edges and interior information near the edge. As stated earlier it is common to have two shapes from the same object that are similar except for segments of the perimeter. Thus, the first step in measuring the similarity is to align the two segments based mostly on the perimeter.

This step is accomplished by employing the *fractional power filter*. [13] This filter has the ability to train on several images concurrently and to also manipulate the trade off between generalization and discrimination that is inherent in first order filters. A matrix \mathbf{X} is created such that each column contains the rasterized pixels of the Fourier components (denoted by the caret) of an image. In our current application one of the cluster images *A* or *B* is the single column in this matrix. The Fourier components of the filter are computed by,

$$\hat{\mathbf{h}} = \mathbf{D}^{-1/2} \, \hat{\mathbf{Y}} \Big[\, \hat{\mathbf{Y}}^T \, \hat{\mathbf{Y}} \Big]^{-1} \, \mathbf{c} \tag{4}$$

$$\hat{\mathbf{Y}} \equiv \mathbf{D}^{-1/2} \hat{\mathbf{X}}$$
(5)

$$D_{ijkl} = \frac{\boldsymbol{d}_{ijkl}}{N} \sum_{n} / \hat{\boldsymbol{x}}_{n,ij} / p$$
(6)

The fractional power is the term p which ranges from 0 to 2. In the case of solid segments the perimeter information is enhanced and the interior is suppressed as p gets closer to 2. At p=2 the only surviving information is the edges and at p=0 the entire interior is included in the filter. If p is too large then only the edge information will survive and the system becomes too discriminatory, and, A and *B* would have to have exactly the same perimeter to create a match. Furthermore, it should be noted that effect of p is not linear. Best performance has been accomplished with p near 0.5.

The filter created from one of the segments creates an image with enhanced perimeter information. This is different than edge enhancement since this filter includes interior information scaled by the distance to the perimeter. The correlation of this filter with the other segment will produce a correlation spike (if there is similarity) and the displacement of the spike from the center of the frame is the shift required to properly align the two segments. An example is shown in figure 1.



Figure 1. An original shape and the FPF of that shape. The darker pixels correspond to the importance of the information in the filter.

Once aligned the measure of similarity matches the number of pixels that are 1 in both aligned images divided by the average number of ON pixels in the two images.

2.2.2. Modified Clustering

The clustering method is based upon the kmeans clustering algorithm but modifications were required. The original method requires the user to define K number of clusters and seed them either randomly or through a combination of the training data. The system then iterates through two steps. The first assigns each constituent (a segment in our case) to a cluster and second re-computes the cluster (usually as the average of its members). The system iterates until no constituent moves to a different cluster.

In the application to this problem this method in its original form got stuck in oscillations. In one iteration some segments would move to a new cluster and in the subsequent iteration they would move back to their original clusters.

Thus, modifications were required. The first modification was that there was no predefined

number of clusters. If a newly considered constituent didn't fit well with any of the existing clusters then a new cluster with it as its inaugural member was created. The second modification was to iteratively move constituents from one cluster to another. Thus, as soon as one constituent is moved the two clusters are updated to reflect the addition or loss of a member. This prevented getting trapped in an oscillation and also allowed the number of clusters to adjust to the complexity of the problem.

Another problem that was experienced was that it was possible that one cluster became widely varied and collected several segments that were not necessarily similar to each other. The solution to this was to not add a segment to a cluster if it caused the intra-cluster variance to increased beyond a threshold.

Finally, there was a purge step. Some of the clusters contained many constituents that were similar to each other. There were, however, other clusters that contained only a single constituent. The reason for this was that this constituent had a shape that was not similar to any of the others in the database. Clusters with less than three constituents were removed.

2.2.3. Shape Representation

Once the clusters have been developed the final step is to represent a probe shape in terms of the clusters. The set of clusters is **C** and thus the number of clusters is $|\mathbf{C}|$. Using the alignment method described above (with the fractional power filter) the probe is aligned with each cluster and a measure of similarity is computed between each cluster and the probe. In this case it is possible to build a filter from all of the constituents within a cluster with each column of **X** being the rasterized Fourier components of the individual segments. However, the filter calculations are heavily dependent upon the number of constituents and thus an alternative method was employed. The filter was constructed from the average of the constituents.

Since there are $|\mathbf{C}|$ clusters there are $|\mathbf{C}|$ number of scalars produced when comparing the probe to the clusters. Thus, a vector of this same length is created from this set of similarity scalars. The vector \mathbf{V} contains the elements V[i] which is the comparison of the probe to the i-th cluster.

In this manner the probe shape is reduced to a descriptive vector that relates how similar the probe is to each cluster. Again these comparisons are based upon size and perimeter information and thus alterations to the probe such as small occlusions will not drastically alter the measure of similarity to the clusters and thus will not significantly alter the vector representation. This is the strong goal of shape representation.

3. Experiment

In this case images selected from random web sites were used to create the database. Some of the images were obtained from the same web site and were similar in content to each other. From a subset of these images 650 segments were isolated.

3.1. Clusters

From this 63 individual segments were created. Figure 2 shows the constituents of a single cluster. As can be seen there is common perimeter information. The cluster average is also shown and this is the description of the cluster.





Figure 2. Four primitives from a single cluster and the aligned average image that defines the cluster.

Other clusters are similar in nature with a varied number of constituents.

3.2. Shape Vectors

Each of the 650 segments was then encoded as prescribed in section 2.2.2. Thus, each segment within a cluster also now had a shape vector. It is expected that the shape vector for all similar segments be likewise similar. To test this the shape vectors for all of the constituents within a cluster were gathered and the statistics computed. The plot in figure 3 depicts the values of the components. In this case only 40 of the elements are shown to enhance clarity. Each error bar displays the average and standard deviation of the respective element. In other words the error bar for x=5 is associated with cluster#5. The shapes used in this test were all from cluster#0 and this chart indicates that the average similarity score for the shapes in comparison to cluster#5 was 0.11 with a standard deviation of 0.015.



Figure 3. The averages and standard deviations of the responses to all clusters from the constituents of cluster#0.

There are a couple of features that need to exist in order for these vectors to be useful. The first is that the error bars need to be small. Large error bars would indicate that the vectors in this group had a varied response with respect to a cluster. That would make recognition via the shape vector difficult. The second is that a plot for another group of vectors should look significantly different.

Consider the plot in figure 4. This plot corresponds to the cluster response for the vectors collected in cluster#2. It is noticed that the response of these vectors were of course strongest when compared to cluster#2. Furthermore, the response to the other clusters is significantly different than those from the vectors of cluster#0.



Figure 4. The averages and standard deviations of the responses to all clusters from the constituents of cluster#2.

Now, consider a shape that has not been used in creating the clusters. It will create a vector of similarity measures to the clusters. We can then compare that vector to the vectors such as shown in figures 3 and 4. For argument assume that this probe shape is similar to those in cluster#2. It is expected that the similarity measures of the probe to the clusters fall within the error bars as shown in figure 4. Thus, the classification of an unknown vector is a measure of similarity to the cluster averages (such as figures 3 and 4). This measure is,

$$H_{k} = \sum_{i} \frac{\left| v_{i} - c_{i}^{k} \right|}{\boldsymbol{s}_{i}^{k}} \tag{7}$$

where v is the similarity measures between the probe and the clusters and *c* is the average similarity measures for cluster *k* (the plotted points in figures 3 and 4). The standard deviation of cluster *k* is σ .

The probe is mostly similar to cluster *m* if *H*[*m*] is the smaller value of the similarity measures.

3.3. Comparing Shape Vectors to a Database

Consider the shape shown in figure 5. It is a shape not used in creating the set of primitives.



Figure 5. A probe shape.

The shape vector was computed for this shape and compared to the average shape vector of each cluster. The best matching clusters are shown in figure 6, and figure 7 shows the worst matching cluster.





Figure 6. Best matching clusters according to shape vectors.



Figure 7. The worst matching cluster.

The efficacy of the shape vectors is portrayed in the similarities of the best matching clusters to the probe and the dissimilarity of the cluster in figure 7. Recall that this comparison was performed using the 1D shape vectors which in this case had 41 elements. Yet, this comparison found the similar shapes according to size and perimeters.

4. Conjecture

A group of segments were isolated and clustered according to similarity of shape as defined by size and perimeter. Each segment was then compared to every cluster and a defining shape vector was calculated.

The shape vectors of all of the constituents of a cluster were gathered and the average and standard deviation of this group was calculated. This was repeated for all clusters.

A similarity measure was established to associate an unknown segment to a cluster via this signature. Thus, it is possible to associate the unknown shape to a cluster.

Acknowledgements

This project was funded in part through a grant from Lockheed Martin.

References

[1] N. B. Karayiannis, "MECA: Maximum Entropy Clustering Algorithms", Tech.Rpt. 94-06 1994, Dept. of EE, Cullen College of Eng., Univ. of Houston, 1994

[2] C-Y. Huang, O I. Camps, T. Kanungo, "Object Recognition Using Appearance-Based Parts and Relations", IEEE CVPR, 877-883, 1997.

[3] Ortega, M., Rui, Y., Chakrabarti, K., Mehrotra, S., Huang, T. S., 1997, Supporting Similarity Queries in MARS. Proc. 5th ACM Intl. Multimedia Conf, 403-413.

[4] J. Z. Wang, J. Li, G. Wiederhold, "SIMPLIcity: Semantics-Sensitive Integrated Machine for Picture Libraries", IEEE Trans. PAMI, **23**(9), 947-963, 2001.

[5] S-C. Zhu, A. L. Yuille, "FORMS: A Flexible Object Recognition and Modeling System", Int. Journal of Computer Vision, **20**(3), 187-212, 1996.

[6] J. Kinser, "Image Primitive Signatures", Applied Image and Pattern Recognition, APIR04, Washington DC, 2004.

[7] J. L. Johnson, H. S. Raganath, H. J. Caulfield, "Image Segmentation Using Pulse Coupled Neural Networks", IEEE Conf. on Neural Networks, Orlando FL 1994.

[8] A. N. Skourikhine, L. Prasad, B. R. Schlei, "Neural Network for Image Segmentation", Proc. SPIE, **4120**, 28-35, 2000.

[9] H.J. Ranganath, G. Kuntimad, J.L.Johnson, "Pulse Coupled Neural Networks for Image Processing", PCNN International Workshop, Huntsville, Alabama, April 5, 1995.

[10] T. Lindblad, J.M. Kinser, Image Processing using Pulse-Coupled Neural Networks, Springer, London, 1998. [11] J. M. Kinser, "Object Isolation", Optical Memories and Neural Networks, **5**(3), 137-145 (1996).

[12] U. Ekblad, J.M. Kinser, N. Zetterlund, "The Unified Cortical Model in Image Processing", Imaging 2003 Intl. Conf. On Imaging Techniques in Subatomic Physics, Astrophysics, Medicine, Biology and Industry, June 2003.

[13] J. Brasher and J. M. Kinser, "Fractional-Power Synthetic Discriminant Functions", Pattern Recognition **27**(4), 577-585 (1994).