Generalized Hough Transforms with Flexible Templates

Kyoung-Mi Lee
Computer Science Department
The University of Iowa
Iowa City, IA 52242, U.S.A.

W. Nick Street
Management Sciences Department
The University of Iowa
Iowa City, IA 52242, U.S.A.

Abstract *The generalized Hough transform (GHT) is useful for detecting and segmenting 2-dimensional (2D) object shapes. However, a weakness of GHT is that one must enumerate many templates corresponding to variation of the desired shape. This paper presents a new approach that uses unsupervised learning to find a set of templates specific to the objects being outlined by the user. The templates are formed by averaging the shapes that belong to a particular cluster. Further, uncertainty regions are added by calculating the standard deviation of points on the template of the cluster. The effectiveness of the resulting system is demonstrated on a medical diagnosis task using cytological images. Our approach can serve as a fully automated substitute to the tedious and time-consuming task of manual shape registration and analysis.*

Keywords: generalized Hough transform, flexible templates, possibilities, segmentation, unsupervised learning

1 Introduction

A main task of computer vision is to locate objects of known shape in a scene. The unknown objects are usually assumed to be identical to one of a set of possible models that the vision system has been taught to recognize, usually by showing it the analytic forms of the models' boundaries or, for arbitrary shapes, sets of discrete points on the boundaries.

Many methods have been proposed to detect and recognize 2D shapes, the most common of which is template matching. The Hough transform (HT) [4], or its variant the generalized Hough transform (GHT) [1], is a robust, well known and very powerful template-matching algorithm to detect an arbitrary, but known, shape in a digital image [1, 8]. GHT is useful for shape analysis in noisy, occluded and multiple-object environments. Generally, an array of points, H(x,y), of the same size as the input image, comprises the 2D accumulator, in which each point has a gradient value G(x,y), specifying the possibility that the reference point of an object shape to be detected is located at the point. These values are accumulated by having each edge point in the image "vote" for the positions that could correspond to the reference point of the particular template. For the test object, the gradient value are obtained from each edge point and are used to compute possible locations of reference points in the accumulator.

There have been several proposals to reduce the huge memory storage capacity and computational time required for GHT. Ma and Chen [9] used analytical features consisting of local curvature and slope to represent shapes, and reduced the 4D parameter space to two 2D parameter spaces. Ser and Siu [12, 13] used edge point pairs to construct a modified look-up table that is orientation- and scale-invariant, and manages to reduce the parameter space to 2D. Lee et al. [8] reduced the parameter space to 1D, defining an edge boundary as the reference point. Jeng and Tsai [5] decreased the parameter space by two degrees, by using half-lines and circles instead of points in their templates to achieve scale and orientation invariance. Lee and Street [7] proposed an iterative approach which eliminates the extra dimensions by using a local accumulator (to find each template in turn) and a global accumulator (to collect the best scores from the local accumulator).

In addition to time and space, another important performance factor is the predictive accuracy of the method. In order to increase the accuracy, Garrido and Blanca [3] introduced the concept of uncertainty regions in the GHT. The points in the parameter space are no longer updated at a single position but for the whole uncertainty region. More concretely, Ser and Siu [12, 13] used neighborhoods (pixels to the left and right of each edge) as the uncertainty region and updated them in the HT space in order to solve the problem from roundoff errors of the scaling factor.

We propose an on-line learning method that uses two important concepts: flexible templates using averaged shapes, and an uncertainty region using the standard deviation of the shapes.

The remainder of the paper is organized as follows. Section 2 briefly describes the foundation of our system, a combination of template matching and deformable contours. Section 3 introduces our shape representation. Section 4 proposes a new algorithm, including the clustering of shapes for flexible templates and the uncertainty region. Finally Sections 5 and 6 present experimental results and conclude the paper.

2 Background

The Xcyt program [10, 15] is a graphical computer program for diagnosing breast cancer and predicting the course of the disease. It performs analysis of cytological features based on a digital scan of a breast fine-needle aspirate, diagnosis of the image as benign or malignant along with an estimated probability of malignancy, and a prediction of when the cancer is likely to recur for cancerous samples. The program has proven highly effective in clinical practice, correctly diagnosing 97.6% of new cases since 1993 and providing accurate and individualized prognosis without lymph node information [14, 16].

Xcyt uses an iterative GHT (IGHT) [11] to detect the boundaries of the cell nuclei in an image. To segment the exact boundaries of the cell nuclei, Xcyt uses an adaptive spline curve-fitting technique known as a snake [6]. The snakes are initialized using the results of IGHT, which searches for elliptical shapes of various sizes. Previous work [7, 14] has shown that the Xcyt system isolates the cell nuclei very well using the combination of IGHT and snakes.

Still, the major problem of IGHT, as with other template matching approaches, is the need to predefine templates to represent shapes. If precise segmentation is necessary and the objects' shapes vary, many templates are needed, requiring significant search time. In medical domains, the shape of the specific organ varies by person, and shape representation must be very precise in order to quantify morphological features - or, even to initialize more precise methods such as snakes. To overcome this drawback, a learning system is proposed that is able to generalize from segmentation examples.

3 Shape Representation

In order to model a scale- and orientation-invariant shape, we represent it by a set of points (r_i, θ_i) in polar coordinates:

$$\mathbf{v} = (r_1, \theta_1, r_2, \theta_2, \cdots, r_n, \theta_n).$$

Assume that OA is an arbitrary radius of the template. In our algorithm, starting from OA and moving clockwise, we divide the circle into n equal arcs to place points around the boundary, with each arc being $\frac{360}{n}$ degrees. So the above set of points can be represented as

$$\mathbf{v} = (r_1, r_2, \cdots, r_n).$$

The shape shown in Figure 1 is represented with 16 linked nodes. Translation invariance is

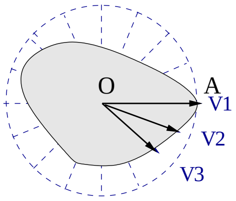


Figure 1: Scale- and orientation-invariant representation.

achieved through the use of an object-centered coordinate system. Shapes are normalized for size during the algorithm, as described in Section 4.2. In this application we limit ourselves to topologically simple shapes with no holes.

4 Method

4.1 Initialization

In this application the objects in question are human cell nuclei, which are more or less elliptical in shape. Therefore we begin by creating a set of elliptic templates with different shapes. Figure 2 shows some examples of elliptic boundaries. Since the algorithm is invariant to size and orientation, only one template with the same shape is created. The initial templates are created automatically for this application but could easily be drawn by the user, or avoided completely, without changing the fundamentals of the algorithm.

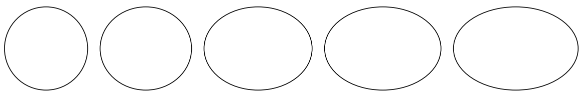


Figure 2: examples of training templates

4.2 Clustering Shapes

During training, the shape of each new object is compared to all existing templates. The difference between a shape, $\mathbf{v}$, and the template, $\mathbf{t}$, is computed with the Manhattan distance metric:

$$\mathbf{e^v} = \sum_{i=1}^{n} |r_i^{\mathbf{v}} - r_i^{\mathbf{t}}| \ ,$$

where $r^{\mathbf{v}}$ is a radius in the shape, $r^{\mathbf{t}}$ is the corresponding radius in the template, and n is the number of points in the shape.

If the minimum difference between a shape and one of the templates is below a threshold, the shape is added to the cluster of shapes represented by that template. This is done using a Point Distribution Model (PDM) [2]. A PDM is a statistical shape model which is generated from a set of examples of the objects to be modeled. Given a set of N clustered shapes in template $\mathbf{t}$, $\mathbf{v}^k = (r_i^k)_{i=1\cdots n}^{k=1\cdots N}$, the average shape, $\bar{\mathbf{v}}$, is calculated using

$$\bar{\mathbf{v}} = (\bar{r}_i)_{i=1}^n = \left(\frac{1}{N} \sum_{k=1}^{N} r_i^k \right)_{i=1}^n$$

If the minimum difference between a shape and one of the templates is greater than the threshold, a new template is created with the shape of the new object. Thus, after training objects, each template represents a cluster of objects that are nearby one another in the space of possible shapes. The similarity threshold is set empirically and can be adjusted by the user.

4.3 Proposed Uncertainty Region

We use the standard deviation as a measure of uncertainty. A deviation is defined here as any distance from the average of a distribution, so the standard deviation is a measure of the width of a distribution equal to the square root of the average of the squared deviations. The standard deviation (**std**) on the average shape, $\bar{\mathbf{v}} = (\bar{r}_i)_{i=1}^n$, is defined as:

$$\begin{aligned}\mathbf{std}(\bar{\mathbf{v}}) &= \mathbf{std}(\bar{r}_i)_{i=1}^n \\ &= \left(\sqrt{\frac{\sum_{k=1}^{N} (r_i^k - \bar{r}_i)^2}{N}} \right)_{i=1}^n\end{aligned}$$

The proposed uncertainty region (**ur**) using the standard deviation is:

$$\mathbf{ur}(\bar{\mathbf{v}}) = \bar{\mathbf{v}} \pm \mathbf{std}(\bar{\mathbf{v}}).$$

Figure 3(a) shows the scattering of points in the same cluster. It can be seen that some of

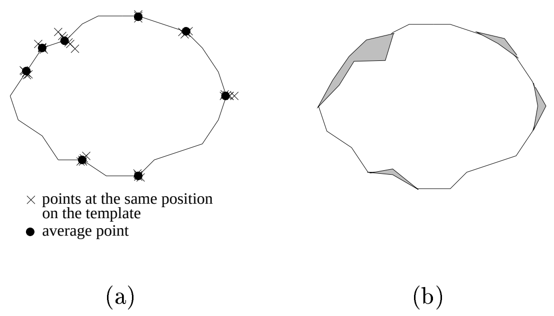


Figure 3: (a) Scatter of points from an aligned set of a templates (b) Flexible template with uncertainty regions

the points show little derivation over the training set, while others form more diffuse collection. Figure 3(b) shows a flexible template consisting uncertainty areas, instead of consisting of edges.

4.4 Detecting Shapes

The proposed GHT algorithm incorporates a flexible template with an averaged shape and an uncertainty region to an iterative approach by voting in every position in the uncertainty region of the accumulator. Accumulator pixels are incremented proportionally to the certainty of the point. Suppose that (*fx*, *fy*) is a flexible point, an x-y coordinate in an uncertainty region, and *ft* is the number of flexible points in the uncertainty region. We define the certainty of the flexible point to the average shape $\mathbf{std}(\bar{\mathbf{v}})$ as:

$$C(fx,fy) = 1 - \frac{D}{\mathbf{std}(\bar{\mathbf{v}})} ,$$
$$0 \leq C(fx_j,fy_j)_{j=1}^{ft} \leq C(\bar{r}\cos\theta,\bar{r}\sin\theta) = 1 ,$$

where D is the distance between the average, $(\bar{r}\cos\theta,\bar{r}\sin\theta)$, and the flexible point.

Given an image and a set of templates, a global accumulator is initialized. For each scale (S) and orientation (Θ) of each template, a local accumulator $\mathrm{L}(x,y)$ is initialized and every edge point (x_c,y_c) in the image performs the following:

for each radius r_i, $1 \leq i \leq n$, of the template,
1. $(x_i',y_i') = (x_c,y_c) + (S \times r_i\cos(\Theta+\theta_i)$, $S \times r_i\sin(\Theta+\theta_i))$, where θ_i is $\frac{360}{n} \times i$,
2. $\mathsf{L}(x_i',y_i') = \mathsf{L}(x_i',y_i') + |\mathsf{G}(x_c,y_c)|$
3. for each flexible point (fx_j,fy_j), $1 \leq j \leq ft_i$,
3.1 $(x_f,y_f) = (x_i',y_i') + (fx_j,fy_j)$,
3.2 $\mathsf{L}(x_f,y_f) = \mathsf{L}(x_f,y_f) + \mathsf{C}(fx_j,fy_j) \times |\mathsf{G}(x_c,y_c)|$

During the clustering of shapes in the training set, the algorithm defines an uncertainty region on an average shape. Using this information, the proposed algorithm updates first an average point (1. and 2.) and then corresponding flexible points (3.).

After normalizing the local accumulator, it compares each point in the local accumulator with the corresponding points in the global accumulator and stores the larger value between both.

4.5 Segmenting Shapes

For each new image, the proposed algorithm is performed using the existing set of templates. After plateaus in the accumulator are removed with a peak-sharpening step, the highest value in the global accumulator is found. This represents the best match of any template to a shape in the image. The appropriate template is overlaid onto the image at the appropriate rotation angle. This process is repeated until a user-defined number of objects has been located.

A snake is initialized for each object using the template points and runs to convergence. The user can then edit the resulting outline by dragging the boundary points to their desired location. The user may also remove an incorrect boundary, or draw a boundary on an undetected object by hand, using the mouse to initialize the snake points.

After an object has been correctly outlined by a snake, the best-matched template, $\bar{\mathbf{v}}$, is updated using this segmented shape, $\mathbf{u}$, to make a new flexible template, $\bar{\mathbf{v}}'$,

$$\bar{\mathbf{v}}' = \frac{N\bar{\mathbf{v}} + \mathbf{u}}{N+1}.$$

The standard deviation of the new template, $\bar{\mathbf{v}}'$, is :

$$\mathbf{std}(\bar{\mathbf{v}}') = \sqrt{\frac{N\mathbf{w} + (\bar{\mathbf{v}}' - \mathbf{u})^2}{N+1}} \ ,$$

where $\mathbf{w} = \mathbf{std}^2(\bar{\mathbf{v}}) + (\bar{\mathbf{v}}' - \bar{\mathbf{v}})^2$.

5 Experimental Results

This algorithm was tested on a set of cytologic images, each of which is gray-scaled with 640×480 pixel spatial resolution.[1] The images contain various numbers of nuclei with radius ranging from 8 up to 32 pixels.

Since the proposed algorithm uses IGHT, the memory requirement is the same with that of IGHT, $O(|w||h|)$, where $|w|$ is the width and $|h|$ is the height of the accumulator. We note that memory requirement for GHT covering the same space of templates, $O(|w||h||S||\Theta|n_t)$, would require more than 1250 times the memory used by the proposed algorithm, where $|S|$ and $|\Theta|$ represents the cardinality of the parameters and n_t is the number of templates searched.

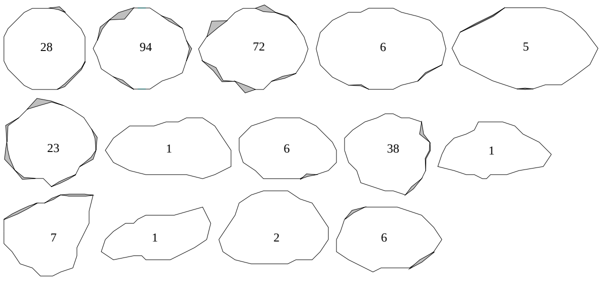


Figure 4: Result - Flexible templates searched

Figure 4 shows the set of flexible templates constructed after training 304 cells on twenty images. The first five shapes are modifications of the predefined templates in Figure 2. The following 9 new templates were created during the experiment. The number inside each template is the number of cells matching that template with various scale factors .

[1]These images are available at `ftp://dollar.biz.uiowa.edu/pub/street/images/`.

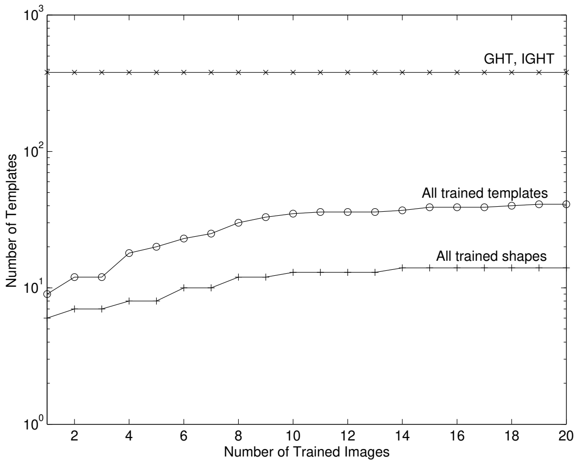
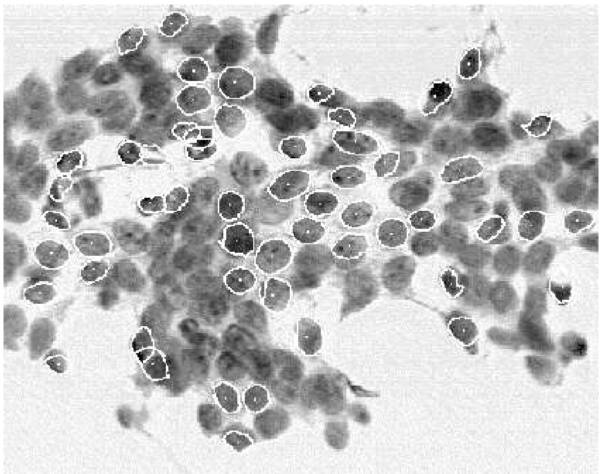


Figure 5: Result - Number of templates searched

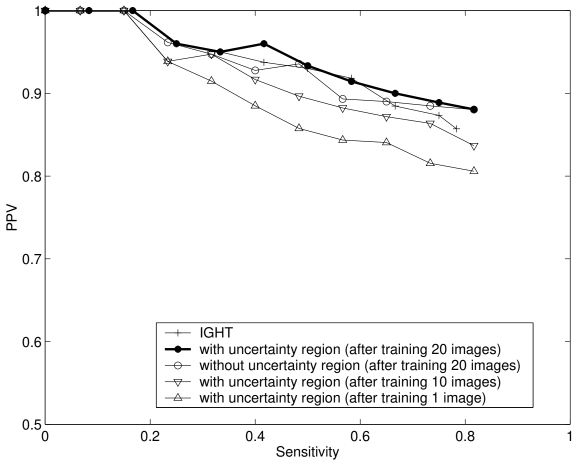
Figure 5 shows number of templates that had to be searched for the test images on a logarithmic scale. The search was carried out in two different ways. The original GHT or IGHT was performed with a constant set of elliptical templates. These vary significantly in size and shape in an attempt to capture the basic shape of all nuclei that might be encountered. Our system builds a set of flexible templates as it is used; the curve labelled "All trained templates" shows the number of these templates at each point. It is counted based on the scale factor. After an initial "ramping up" period, this number grows very slowly at a level nearly an order of magnitude below the original set of templates. The curve labelled "All trained shapes" shows the number of shapes, ignoring the scale factor. As with the curve "All trained templates" curve, after creating most of the new templates in the initial training period of Figure 4, it shows a stable number of learned templates.

The test proceeded as follows. Two test images were set aside and used for all tests. The system was trained on a sequence of training images and tested on both test images after each training image.

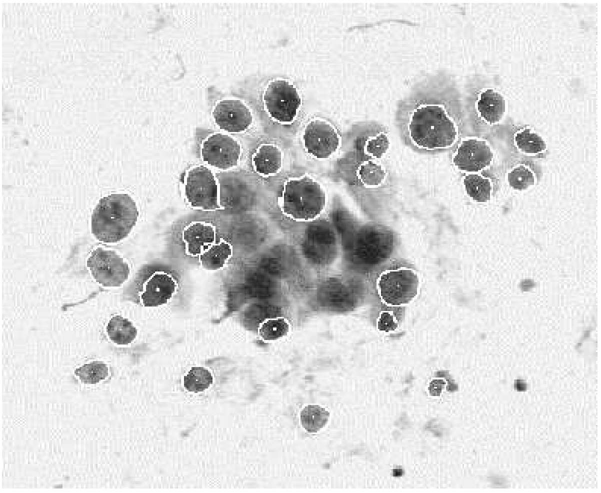
We used two measures to evaluate the performance of the algorithm: sensitivity and positive predictive value (PPV). Sensitivity is the likelihood that a nucleus will be detected in



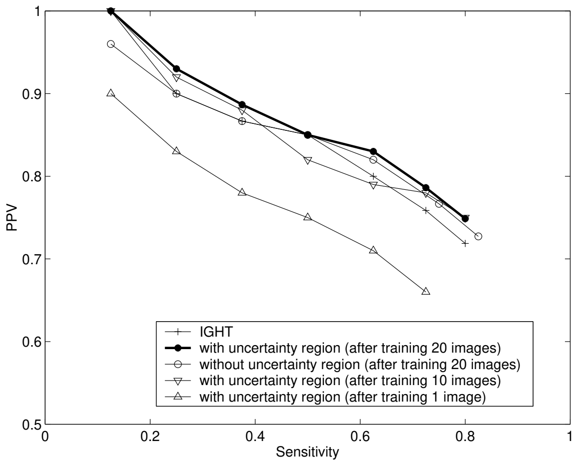
(a)



(a) Figure 6 (a)



(b)



(b) Figure 6 (b)

Figure 6: Result - test images reduced by 50%

Figure 7: Result - Sensitivity vs. Positive Predictive Value (PPV) on test images

case it has been marked by a expert. PPV is defined as the likelihood that the segmentation of a nucleus is actually associated with a nucleus segmented by an expert. As Figure 7 shows, as the system is trained on more and more images, its ability to correctly segment the nuclei in the test images increases. The accuracy of the new system reaches that of the exhaustive GHT after twenty training images.

6 Summary

This paper introduces a new GHT algorithm with an on-line learning scheme to aid the location and segmentation of objects in large-scale image analysis tasks. The system demonstrates dramatic improvements in the time needed to perform template-matching search while obtaining a higher quality result. Future work will focus on the improvement of the on-line, one-pass clustering algorithm and the incorporation of a classification algorithm for heterogeneous object population.

References

[1] D. H. Ballard. Generalizing the Hough transform to detect arbitrary shapes. *Pattern Recognition*, 13(2):111–122, 1981.

[2] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. Active shape models-their training and application. *Computer Vision and Image Understanding*, 61(1):38–59, 1995.

[3] A. Garrido and N. Perez De La Blanca. Physically-based active shape models initialization and optimization. *Pattern Recognition*, 31(8):1003–1017, 1998.

[4] P. V. C. Hough. Method and means for recognizing complex patterns. U.S. Patent 3069654, 1962.

[5] S.-C. Jeng and W.-H. Tsai. Scale- and orientation-invariant generalized Hough transform - a new approach. *Pattern Recognition*, 24(11):1037–1051, 1991.

[6] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, pages 321–331, 1988.

[7] K.-M. Lee and W. N. Street. A fast and robust approach for automated segmentation of breast cancer nuclei. In *Proceedings of the IASTED International Conference on Computer Graphics and Imaging*, pages 42–47, 1999.

[8] M. Lee, J. Kittler, and K. C. Wong. Generalized Hough transform in object recognition. In *Proceedings of the International Conference on Pattern Recognition*, volume 3, pages 285–289, 1992.

[9] D. Ma and Xing Chen. Hough transform using slope and curvature as local properties to detect arbitrary 2D shapes. In *Proceedings of the International Conference on Pattern Recognition*, pages 511–513, 1988.

[10] O. L. Mangasarian, W. N. Street, and W. H. Wolberg. Breast cancer diagnosis and prognosis via linear programming. *Operations Research*, 43(4):570–576, 1995.

[11] H. Oh and W. N. Street. A memory-efficient generalized Hough transform for segmenting cytological images. Master's thesis, Oklahoma State University, 1997.

[12] P. K. Ser and W. C. Siu. A rotation invariant generalized Hough transform for image recognition. *IEE Processing - Computers and Digital Techniques*, pages 231–235, 1992.

[13] P. K. Ser and W. C. Siu. Non-analytic object recognition using the Hough transform with matching technique. *IEE Processing - Computers and Digital Techniques*, 141(1):11–16, 1994.

[14] W. N. Street. Xcyt: A system for remote cytological diagnosis and prognosis of breast cancer. In L. C. Jain, editor, *Soft Computing Techniques in Breast Cancer Prognosis and Diagnosis.* CRC Press, 2000. In press.

[15] W. H. Wolberg, W. N. Street, and O. L. Mangasarian. Machine learning techniques to diagnose breast cancer from image-processed nuclear features of fine needle aspirates. *Cancer Letters*, 77:163–171, 1994.

[16] W. H. Wolberg, W. N. Street, and O. L. Mangasarian. A comparison of computer-based nuclear analysis versus lymph node status for staging breast cancer. *Clinical Cancer Research*, 5:3542–3548, 1999.