2003 International Symposium on Advanced Intelligent Systems
September 25-28, 2003, Jeju, Korea

A Possibilistic *C*-Means Approach to the Hough Transform for Line Detection

Frank Chung-Hoon Rhee+ and Eun-A Shim++
+Computation Vision and Fuzzy Systems Laboratory
Department of Electronic Engineering
Hanyang University
Ansan, Korea
frhee@fuzzy.hanyang.ac.kr

++LG Electronics
eashim@lge.co.kr

***Abstract*- The Hough transform (HT) is often used for extracting global features in binary images, for example curve and line segments, from local features such as single pixels. The HT is useful due to its insensitivity to missing edge points and occlusions, and robustness in noisy images. However, it possesses some disadvantages, such as time and memory consumption due to the number of input data, and the selection of an optimal and efficient resolution of the accumulator space can be difficult. Another problem of the HT is in the difficulty of peak detection due to the discrete nature of the image space and the round off in estimation. In order to resolve the problem mentioned above, a possibilistic *C*-means approach to clustering [1] is used to cluster neighboring peaks. Several experimental results are given.**

I. INTRODUCTION

An important task in intermediate level computer vision, which bridges low and high-level tasks, is to extract global features, e.g. curve and line segments, from local features such as single pixels. One of the tasks involves edge detection. However, an edge image obtained by edge detection seldom form closed and connected boundaries due to noise and breaks. Thus, edge detection algorithms typically are followed by edge linking [2],[3]. In particular, the Hough transform (HT) is often used for performing this task.

The HT is a mapping from image space into a parameter space (i.e., Hough space). The mapping determines the parameters of the detected lines. The HT is useful since it is insensitive to missing edge points and occlusions, and robust in noisy images. However, the HT possesses some problems, such as time and memory consumption, which is dependent on the number of input data, and the selection of an optimal and efficient resolution of the accumulator space can be complicated. To overcome these problems several new HT-based methods have been proposed recently ; 1) the fast Hough transform and the adaptive Hough transform, which uses an iterative "coarse to fine" exploration of the parameter space, and 2) the randomized Hough transform and the probabilistic Hough transform which uses the idea of random sampling of the data [3].

HT-based methods are composed of three steps. First is the transformation of each point in the image space into the Hough space (HS). Next is the detection of the peak points in the HS. In this step, the difficulty of the peak detection due to the discrete nature of the image space can lead to round off errors in the corresponding HS. In order to resolve the problem mentioned, the possibilistic *C*-means approach to clustering is used to cluster neighboring peaks.

The remainder of this paper is organized as follows. In Section II, we discuss some basic concepts of the Hough transform pointing out the strong and weak points. In Section III, we discuss the basic ideas behind possibilistic clustering. In Section IV, we discuss how possibilistic clustering is applied to the HT. In Section V, several experimental results are given. It is shown that a more accurate position of the peaks in the HS can be detected. Finally, Section VI gives the conclusions.

II. HOUGH TRANSFORM (HT)

This section introduces the basic principles of the HT pointing out the strong and weak points.

A. The Hough Transform Algorithm

Suppose that, for *n* points in an image, we want to locate subsets of these points that lie on straight lines. One possible solution is to first find all the lines determined by every pair of points and then find all subsets of points that are close to some particular line. The problem with this procedure is that it involves finding $C_2^n \approx n^2$ lines and then performing $nC_2^n \approx n^3$ comparisons of every point to all the lines. This approach can be computationally expensive in all but the most trivial applications [2]. As an alternative, a method commonly referred as the Hough transform (HT) was developed. The method is described as follows.

Consider the normal representation of a line [2] as

$$x\cos\theta + y\sin\theta = \rho\,, \tag{1}$$

where (ρ,θ) defines a vector from the origin to the nearest point on the line, as shown in Fig. 1(a). This vector is perpendicular to the line. We now consider a 2-D space defined by the two parameters ρ and θ. Any point in the *x-y* plane plots to a curve in the ρ-θ space. Thus, the HT of a straight line that pass through these points corresponds to a point in the ρ-θ space. That is the intersection of all the curves in the ρ-θ space.

For example, consider a particular point (x_1,y_1) in the x-y plane. There exist an infinite number of straight lines that pass through this point, and each of these lines corresponds to a point in the ρ-θ space. In addition, these points must satisfy equation (1) with x_1 and y_1 as constants. Thus, the locus of all such lines in the x-y space is a sinusoid in the parameter space. Hence, any point in the x-y plane (see Fig. 1(b)) corresponds to a sinusoidal curve in the ρ-θ space (see Fig. 1(c)).

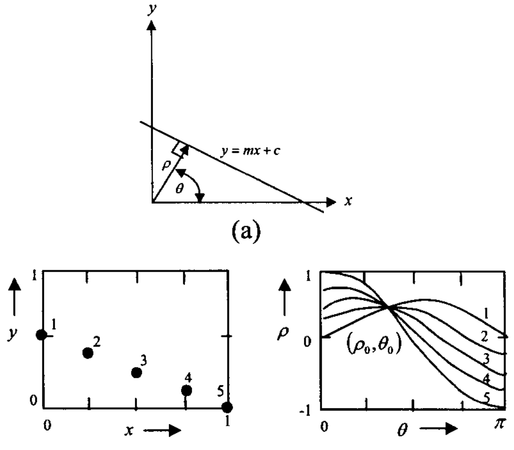


Fig. 1. Polar coordinate expression of (a) a straight line, (b) five collinear points in the x-y plane, and (c) corresponding points expressed in the ρ-θ plane.

If we have a set of edge points (x_i,y_i) that lie on a straight line having parameters ρ_0 and θ_0, then each edge point plots to a curve in the ρ-θ space. However, all these curves must intersect at the point (ρ_0,θ_0), since they all lie on the same line (Fig. 1(c)). Thus, to find the straight-line segment that the points all fall upon, we can consider a 2-D histogram in the ρ-θ space.

For each edge point (x_i,y_i), we increment all the histogram bins in the ρ-θ space that correspond to the HT (sinusoidal curve) for that point. When we have completed this for all the edge points, the bin containing (ρ_0,θ_0) is considered as a local maximum. Thus, our objective is to search the ρ-θ space histogram for local maxima and therefore obtain the parameters of the linear boundary segments.

B. Disadvantages of the Standard Hough Transform

The standard approach of implementing the HT involves several computational difficulties: 1) the HT process can be time and memory consuming depending on the number of input data, and 2) the selection of an optimal and efficient resolution of the accumulator space can be difficult.

The local maxima, peak points in the Hough space (HS), can be detected according to given threshold value. Due to the discrete nature of the image and HS, the peak corresponding to a curve in the image space can be distributed among several neighboring bins in the HS. This can make peak detection difficult.

The following simple example illustrates the problems associated with the distribution of peaks. Fig. 2(a) shows an example consisting of five image points where three are collinear. Fig. 2(b) shows the ρ-θ curves in the HS. Fig. 2(c) shows the peaks detected using a suitable threshold value (i.e., 40) and Fig. 2(d) shows the reconstructed image using the peaks in Fig. 2(c).

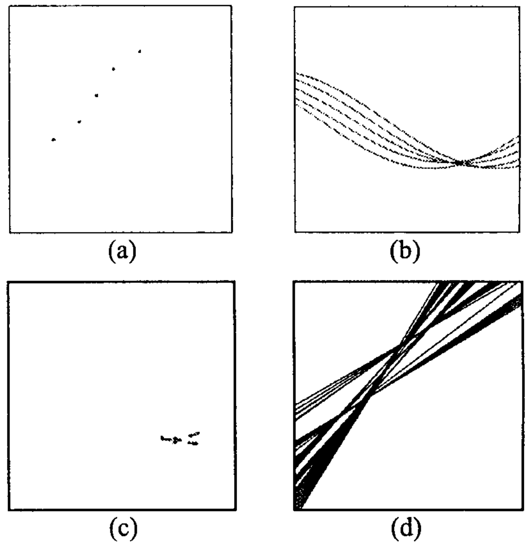


Fig. 2. Example of (a) three collinear data points, (b) ρ-θ curves in the HS, (c) peak points detected by threshold value, and (d) result of the inverse HT of (c).

As shown in above example, there exist several spurious peaks due to the discrete nature of the image and HS. As a result incorrect line detection in the image can occur. This problem is resolved by clustering the neighboring bins in the accumulator cells as explained in the following section.

III. POSSIBILISTIC *C*-MEANS CLUSTERING

Most analytic fuzzy clustering approaches are derived from Bezdek's fuzzy *C*-means (FCM) algorithm [4] and are developed under the probabilistic constraint that the memberships of a data point across clusters must sum to 1. This constraint came from generalizing a crisp *C*-partition of a data set, and was used to generate the membership update equations for an iterative algorithm based on the minimization of a least-squares type of criterion function. The constraint on the memberships used in the FCM algorithm is meant to avoid the trivial solution of all memberships being equal to 0, and it gives meaningful results in applications where it is appropriate to interpret memberships as probabilities or degrees of sharing [1].

The following simple example illustrates the problems associated with the probabilistic constraint used in the FCM algorithm as related to the interpretation of the resulting memberships. Consider the data sets shown in Fig. 5. The constraint on the memberships mentioned above would enforce the two cluster points A and B to have the same degree of membership. That is, a membership $u = 0.5$ for clusters 1 and 2.

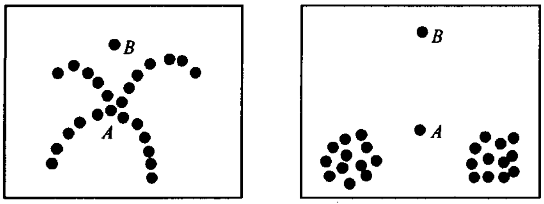


Fig. 3. Two cluster data sets.

The memberships may then be considered as a kind of "relative membership" for the clusters (i.e., the membership of point B in cluster 1 compared to the membership of point B in cluster 2). From an observer's point of view it might, however, be inappropriate to assign the same degrees of membership to points A and B since he interprets those as (absolutes) degrees of membership. For example, degrees to which points A or B belong to clusters 1 or 2, respectively. Krishnapuram *et al.* [1] proposed a possibilistic approach to the FCM algorithm in order to compute the latter type of degrees of membership for elements in clusters by modifying the definition of a fuzzy c-partition. As a consequence, the objective function of the FCM algorithm needs to be modified.

A. Basic Ideas of the Possibilistic C-Means Algorithm

For a given set of data points $X = \{x_1,\ldots,x_n\} \subset R^s$ the possibilistic clustering problem can be formulated as the following optimization problem [1].

Minimize

$$J_m(U,C) = \sum_{i=1}^{n}\sum_{j=1}^{c} u_{ij}^m d_{ij}^2 + \sum_{j=1}^{c}\eta_j \sum_{i=1}^{n}\left(1-u_{ij}\right)^m \quad (2)$$

subject to $\max_j u_{ij} > 0$,

where n is the number of data points to be clustered, c is the number of clusters ($1 \le c \le n$), u_{ij} ($0 \le u_{ij} \le 1$) is the membership grade for the ith data point in the jth cluster, m is the fuzzification factor ($m>1$), and $d_{ij}=\|x_i-c_j\|$ is the Euclidean distance between point x_i and the center c_j. The partition $U=\{u_{ij}\}$ is an $n\times c$ matrix and $C=[c_1,\ldots,c_c]$ is an $s\times c$ matrix. A family of possibilistic clustering algorithms can now be expressed as follows [1].

Possibilistic *C*-Means (PCM) Algorithm

Fix number of clusters, c;
Fix m, $1<m<\infty$;
Initialize Possibilistic C-partition $U^{(1)}$;
Set $k=1$;
Estimate η_j using

$$\eta_j = K\sum_{i=1}^{n} u_{ij}^m d_{ij}^2 \Big/ \sum_{i=1}^{n} u_{ij}^m \, ; \quad (3)$$

REPEAT
Compute $C^{(k)}$ using

$$\mathbf{c}_j(k+1) = \sum_{i=1}^{n} u_{ij}^m(k)\mathbf{x}_i \Big/ \sum_{i=1}^{n} u_{ij}^m(k)\, ; \quad (4)$$

Compute $U^{(k+1)}$ using

$$u_{ij} = 1\Big/\left[1+\left(d_{ij}^2/\eta_j\right)^{1/(m-1)}\right]; \quad (5)$$

Set $k=k+1$;
UNTIL $\left(\left\|U^{(k)}-U^{(k+1)}\right\| < \varepsilon\right)$

In the above algorithm, the value of η_j determines the distance at which the membership value of a point in a cluster becomes 0.5 (i.e., "the 3 dB point"). In general, it is desirable that η_j relate to the overall size and shape of the jth cluster. Also, it is to be noted that η_j determines the relative degree to which the second term in the objective function (2) is important compared with the first. Typically K is chosen to be 1.

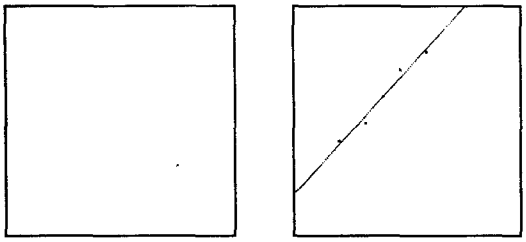
IV. HOUGH TRANSFORM USING PCM

As mentioned in the previous section, due to the discrete nature of the image and Hough space, the peak corresponding to a curve in the image space can be distributed among several neighboring bins in the HS. This may allow peak detection to become difficult. Therefore, the PCM algorithm is used to cluster the neighboring bins in the accumulator cells. The new method of the HT algorithm incorporating the PCM algorithm is described as follows.

PCM based Hough Transform for Line Detection

Create set of all edge points (x_i,y_i) in the binary image;
Quantize parameter space between appropriate maximum and minimum values for ρ and θ;
Form accumulator array $A(\rho,\theta)$;
FOR all points in $A(\rho,\theta)$ **DO**
    Set $A(\rho,\theta)=0$;
END FOR
FOR all points in $A(\rho,\theta)$ along the appropriate line **DO**
    Set $A(\rho,\theta)=A(\rho,\theta)+1$;
END FOR
FOR all points in $A(\rho,\theta)$;
  IF $A(\rho,\theta)$ is greater than some threshold value **THEN**
    Set $A(\rho,\theta)$ to local maxima;
  END IF
END FOR
Create data set of peak points (local maxima) in HS;
FOR all peak points in $A(\rho,\theta)$ **DO**
    Execute PCM clustering algorithm;
END FOR
Extract line corresponding to centers of clusters.

In above algorithm, each center of the clusters corresponds to a line in the image space. Fig. 4(a) shows the result of clustering of the peaks for the local maxima in Fig. 2(c). Fig. 4(b) shows the result of the inverse HT. As seen in the figure, incorrect line detection in the image can be eliminated by incorporating the PCM algorithm.



(a) (b)

Fig. 4. Clustering result of (a) peaks of the local maxima in Fig. 2(c) and (b) Inverse HT result of (a).

Next, we use the following method for determining the number of clusters c, for the PCM based HT.

1. Plot the histogram of the HS to choose a proper threshold value as shown in Fig. 5(a).
2. Choose the maximum value that contains most of the peaks in the histogram as the threshold value.
3. Create the data set of the Hough cells whose cell counts exceed the threshold value.
4. Cluster the data set and estimate the scatter matrices for different values of c.
5. Choose the value of c such that the trace of the within-scatter and between-scatter matrices begins to saturate as the number of clusters is increased.

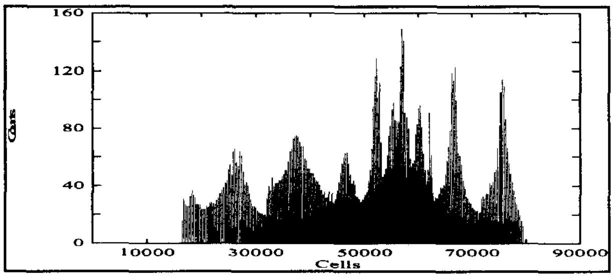


Fig. 5. Histogram example of HS.

V. EXPERIMENTAL RESULTS

We now present an example and compare the results with the standard Hough transform (SHT). Fig. 6(a) shows a 256×256 edge image with added uniformly distributed Gaussian noise. Fig. 6(b) shows the histogram plot. As seen in the figure, there exist many spurious peaks. Fig. 6(c) shows peak points detected by the PCM algorithm. Here, the threshold value used is 40 and the number of clusters obtained by using the scatter matrices is 11. By clustering the spurious peaks, the spurious peaks are removed, leaving only the accurate peak points. Fig. 6(d) shows the reconstructed image. Comparing with the original image, one can see that the resulting edge image has reconstructed well. It is shown in Fig. 6(e) that the result of the SHT shows gaps between some points and incorrect line segments.

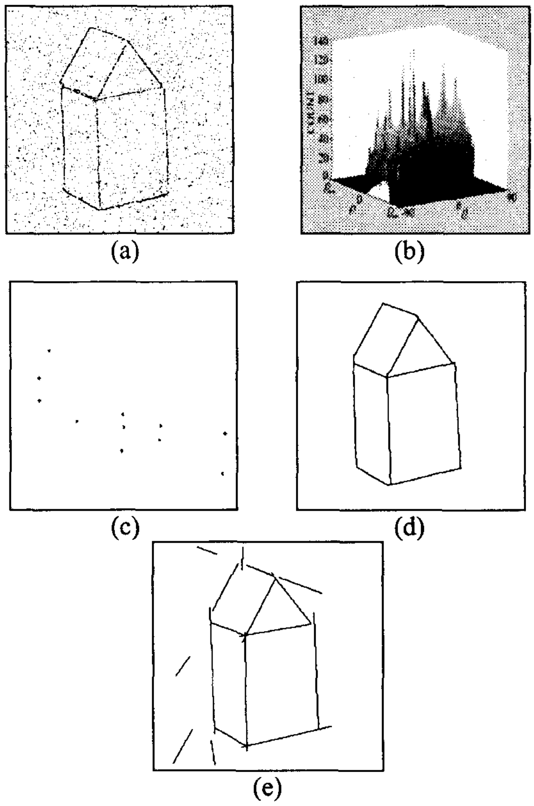


Fig. 6. Edge image (a) with added Gaussian noise, (b) histogram plot of the HS, (c) clustering result, (d) reconstructed image obtained by proposed method, and (e) reconstructed image obtained by SHT.

VII. CONCLUSIONS

In this paper, a new method of the HT was proposed to find an accurate peak position using the PCM algorithm. In order to remove spurious peaks, we clustered several neighboring bins in the accumulator cells. Results of our experiments show that our method can detect an accurate peak position in the presence of noise better than that of the SHT. Results show that the proposed algorithm did not include gaps and unnecessary lines as in the SHT. Although the examples presented consisted of straight lines, the proposed method can be extended for more complex shapes. This is currently under investigation.

REFERENCES

[1] R. Krishnapuram and J. Keller, "A possibilistic approach to clustering," *IEEE Trans. Fuzzy Systems*, vol. 1, no. 2, pp. 98-110, 1993.

[2] R. Gonzalez and R. Woods, *Digital Image Processing*. Addison-Wesley Publishing Co., Inc., 1993.

[3] H. Kalviainen, "Randomized hough transform: New extensions," Unpublished, Ph.D. dissertation, University of Technology, Lappeenranta, Finland, 1994

[4] J. Bezdeck, *Pattern Recognition with Fuzzy Objective Function Algorithms*. New York, NY: Plenum Press, 1981.