PERGAMON

Pattern Recognition 35 (2002) 1193–1196

PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/patcog

Rapid and Brief Communication

# An unsupervised generalized Hough transform for natural shapes

Noël Bonnet *

*INSERM Unit 514, UMRS (IFR 53 "Biomolecules"), LERI (University of Reims) Hôpital Maison Blanche, 45 rue Cognacq Jay, 51092 Reims Cedex, France*

Received 9 October 2001; accepted 30 October 2001

**Abstract**

The Hough transform was originally designed to recognize artifical objects in images. A Hough transform for natural shapes (HTNS) was subsequently proposed, but necessitates the supervised learning of the class of shapes. Here, we extend HTNS to unsupervised pattern recognition, the variability of the object class being coded with tools originating from mathematical morphology (erosion, dilation and distance functions). 

*Keywords:* Hough transform; Natural shapes; Mathematical morphology

## 1. Introduction

The Hough transform (HT) is one of the most powerful approaches for the detection and recognition of patterns with known simple geometrical shapes in images [1–3]. The generalized Hough transform (GHT) was developed to cope with arbitrary shapes [4]. While GHT is able to deal with *artificial* objects characterized by their rigid structures, it is not very well suited to *natural* objects, which are characterized by much more flexibility, i.e., similarity but not identity, within one class of objects.

In order to cope with such objects, a variant of the HT was proposed in Ref. [5] and was called Hough transform for natural shapes (HNS). The HNS has proved to be effective in detecting and classifying natural objects. However, one drawback of the method is that it is a supervised procedure: a set of references of the object class must be collected in order to define the object model. Although this may not be a problem for some (supervised) pattern recognition applications, such reference objects may not be available in some cases. Consider, for instance, the problem of tracking a deformable object. We have only one object available (the one to track) and the HNS method cannot be applied.

In this note, it is proposed to extend the HNS method to situations where only one object of the class is available as a reference. A suitable acronym for this method is HTNS-SR, for "HT for natural shape, based on a single realization (or a single reference)". Two variants of the method are suggested: one relies on binary mathematical morphology (BMM) and the other on gray-level mathematical morphology (GLMM).

The paper is organized as follows. In Section 2, the HNS method suggested by Samal and Edwards is summarized briefly. Section 3 describes two variants (HTNS-SR1 and HTNS-SR2) for performing HTNS with a single reference object. Comments are also offered on the way natural variability is taken into account by these two variants. Finally, an experimental example illustrating the method is presented in Section 4.

## 2. Summary of the HNS method

The aim of a HNS algorithm is essentially to code the shape variability within an object class, that is, to define a

* Tel.: +33-3-26-78-77-71; fax: +33-3-26-06-58-61.
*E-mail address:* noel.bonnet@univ-reims.fr (N. Bonnet).


PII: S0031-3203(01)00219-9

model of the object class. The method suggested in Ref. [5] consists in estimating this variability by obtaining a series of different prototypes for the object class. In that sense, it is a supervised learning approach: an expert has to tell the pattern recognition system regarding the similar but different objects that belong to the class. From binary silhouettes of these prototypes, called *interiors* of the objects, *min* and *max* areas of the class are defined as the intersection and union of all silhouettes. Then the potential boundary points (PBPs) are defined as pixels enclosed within the area obtained by subtracting the *min* from the *max* area. In the following, object contours whose pixels are PBPs of the object class are considered as belonging to this class. The remainder of the procedure is similar to the mapping procedure of the GHT. For all the selected pixels of the studied image (i.e., pixels with a gradient magnitude greater than a selected threshold), a mapping procedure is performed, i.e., the class PBPs are drawn in the parametric space, around the selected pixels considered as the potential centers of mass of the shape. If many pixels are boundary points of an object belonging to the class, then the different PBPs overlap each other at a point which is in fact the center of mass of the found shape, see Fig. 6 in Ref. [5].

## 3. Extensions of the HNS procedure involving one single reference shape

The HNS procedure involves several different reference shapes, which are assumed to represent the variability of natural shapes. Arguments are now presented to show that HNS may also be applied with one single reference of the object class, i.e., without supervision.

### 3.1. First variant (HTNS-SR1)

Since we consider only one reference, we cannot use the concepts of *min* and *max* areas for evaluating the PBPs of the object class. We suggest that *min* and *max* may be replaced by the silhouettes obtained after *erosion* and *dilation* of the single reference silhouette. These operations belong to the category of BMM tools and they possess all the properties needed for our purpose. Thus, the PBPs are the set of pixels enclosed in the area defined by

$$\mathrm{PBP}(\varepsilon) = (S_{\mathrm{ref}} \oplus \varepsilon) - (S_{\mathrm{ref}} \ominus \varepsilon), \tag{1}$$

where $S_{\mathrm{ref}}$ is the reference silhouette, $\oplus$ and $\ominus$ are the Minkowski addition and subtraction operators and $\varepsilon$ is a structuring element, the size of which defines the amount of variability.

Once the PBPs are defined as a model of the natural shape, the procedure follows exactly the same line as the HNS: during the voting step, the PBPs are mapped for each boundary pixel detected in the image space. Then, peaks are searched in the parameter space. Peaks with a magnitude higher than a fixed threshold correspond to centers of gravity of detected shapes.

### 3.2. Second variant (HTNS-SR2)

An alternative way to code the variability in the shape is to define a PBP region whose points possess continuous values (defining potential boundary values: PBVs), instead of binary ones. Such a non-binary model can be obtained using other concepts originating from GLMM, such as internal and external distances. For any pixel in the interior of a shape, the internal distance is a measure of its distance to the closest boundary point. For pixels outside the object, the external distance has an equivalent definition. The overall distance of any pixel to an object can be expressed as

$$\begin{aligned} D(x,y) &= D_{\mathrm{int}}(x,y) \quad \text{if the pixel } (x,y) \text{ is inside the object} \\ &= D_{\mathrm{ext}}(x,y) \quad \text{if the pixel is outside the object.} \end{aligned} \tag{2}$$

Once this overall distance is evaluated, it can be converted into a PBV, which decreases as a function of distance to the boundary pixels of the reference. One possibility consists in choosing

$$\mathrm{PBV}(x,y) = \mathrm{e}^{-k \cdot D(x,y)}, \tag{3}$$

where $k$ is a coefficient which reflects the amount of acceptable variability. The remaining part of the procedure is still the same, except that the PBVs are mapped instead of PBPs.

## 4. Illustration of the procedure

Fig. 1a represents one shape reference. The aim is to detect similar shapes within scenes. For doing this, we first have to code the accepted variability in the reference shape, according to one of the two procedures described above. Fig. 1b and c displays the shape reference after dilation and erosion, respectively, by a structuring element with a size of $7 \times 7$ pixels. Fig. 1f shows the difference between these two images. White pixels represent the PBPs, i.e., pixels that are allowed to be considered as boundary pixels of objects of the same class. Fig. 1d and e displays the internal and external Euclidean distances, respectively. Brighter pixels are situated farther from the boundary of the reference shape. Fig. 1g was computed from these distances using Eq. (3). It represents a function (the PBV) that decreases when the pixel is farther from the boundary of the reference shape, and thus can be considered as expressing the probability that a pixel belongs to the boundary of any shape in the class of similar shapes. Any of these two templates can be used for performing the detection of a shape similar to that represented in Fig. 1a. Fig. 1h represents a scene composed of several shapes, one of which is similar, but not identical, to the reference shape. Fig. 1i displays the boundary pixels detected in the scene. Using a mapping procedure similar to

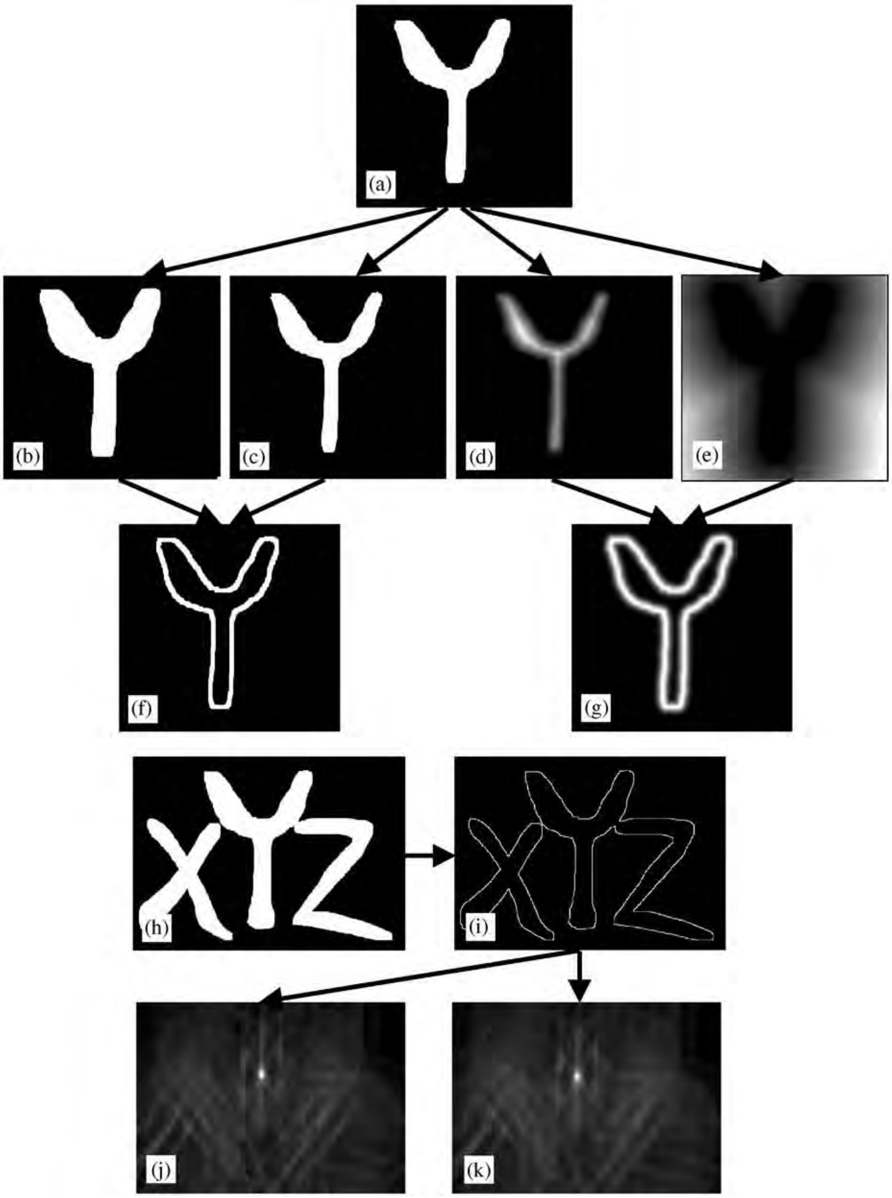


Fig. 1. Illustration of the detection of a shape in a scene, on the basis of templates coding the variability of the reference object class: (a) template binary shape, (b) binary shape dilated by a filled structuring element, (c) binary shape eroded by the same structuring element, (d) internal Euclidean distance computed on the binary shape, (e) external Euclidean distance, (f) PBPs for the class of shapes similar to that in (a), (g) potential boundary values for the class of shapes, computed according to Eq. (3), (h) a binary scene containing several shapes, (i) the boundary points of objects present in scene (h), (j) Hough parameter space obtained after mapping the template in Fig. 1f on the boundary points in Fig. 1i and (k) Hough parameter space obtained after the mapping of template in Fig. 1g onto the boundary points in Fig. 1i. In both cases, the center of gravity of the 'Y' shape is detected as a well-defined peak against a background.

that used for the GHT, with the templates in Fig. 1f and g, we get the results displayed in Fig. 1j and k, respectively. In both cases, a peak results from the accumulation of templates. Its position corresponds to the center of gravity of the detected shape ('Y'). No spurious peak with high amplitude occurs for shapes that do not belong to the class, i.e., 'X' and 'Z'.

## 5. Conclusion

In this paper, two extensions of the proposal by Samal and Edwards are suggested. The main advantage is that while the previous approach requires supervision, our method remains unsupervised. In this new approach, the variability of the class is fixed by the user, through parameters entering the definition of the potential boundary region: size ($\varepsilon$) of the structuring element in variant HTNS-SR1 and parameter ($k$) in variant HTNS-SR2. It is believed that this type of approach can be helpful for the pattern recognition (or tracking) of deformable objects, when only one prototype of the object class is available.

A more rigorous presentation of the method, together with a discussion of its limitations compared to the limitations of alternative pattern recognition methods (such as template matching), will be given in a forthcoming extended version of this paper.

## References

[1] P.V.C. Hough, Method and means for recognizing complex patterns, U.S. patent 3 069 654, 1962.
[2] J. Illingworth, J. Kittler, A survey of the Hough transform, Comp. Vision Graph. Im. Proc. 44 (1988) 87–116.
[3] E.R. Davies, Machine Vision: Theory, Algorithms, Practicalities, Academic Press, London, 1990.
[4] D.H. Ballard, Generalizing the Hough transform to detect arbitrary shapes, Pattern Recognition 13 (1991) 111–122.
[5] A. Samal, J. Edwards, Generalized Hough transform for natural shapes, Pattern Recognition Letters 18 (1997) 473–480.