A New Accumulator-Based Approach to Shape Recognition

Karthik Krish and Wesley Snyder

Department of Electrical and Computer Engineering, North Carolina State University

Abstract. An algorithm is presented which uses evidence accumulation to perform shape recognition. Because it uses accumulators, noise and isotropic measurement errors tend to average out. Furthermore, such methods are intrinsically parallel. It is demonstrated to perform better than any competing technique, and is particularly robust under partial occlusion. Its performance is demonstrated in applications of silhouette and face recognition using only edges and in solving the correspondence problem for image registration. The method uses only biologically-reasonable computations.

1 Introduction

The use of accumulator arrays for evidence accumulation in image analysis is not new, dating back to its use for finding straight lines in the original Hough transform[8]. In an accumulator-based algorithm, noise and isotropic distortions tend to average out. This paper uses the concept of accumulators to develop a very general approach to shape recognition. The strategy begins by making local measurements at salient points in an image. The application of the concept in our algorithm is most similar to that described in the Generalized Hough Transform[2], but the algorithm differs in several ways, including its use of feature vectors and its table lookup model for shape. The strategy provides a robust solution to the problem of making global decisions from a collection of local measurements, considered one of the fundamental problems of machine intelligence in general and machine vision in particular.

The method, which we call the "SKS algorithm", is invariant to rotation (in the camera plane), translation, and scale change, and very robust to partial occlusion and local variations in image brightness.

The algorithm was developed under the constraint that all operations must be reasonably computed by a biological neural network. That is, we constrain the operations to have the properties:

* May use a great deal of memory
* Must use simple computations
* Must use low precision arithmetic
* Should make use of a great deal of parallelism

In this paper, the recognition algorithm is presented first. Second, the strategy for learning models is presented. Finally, examples of recognizing shapes are presented.

G. Bebis et al. (Eds.): ISVC 2008, Part II, LNCS 5359, pp. 157–169, 2008.

Although the algorithm is motivated by a desire to understand mid-level recognition of shapes in biological brains, this paper does not pursue that motivation explicitly. In particular, neuron-level computations are not simulated here, nor do we directly incorporate known neural architectural components such as receptive fields or striate interactions, leaving those to a planned later paper.

2 Recognition

The recognition process begins with identification of salient points in an image. This process produces an unordered collection of $[x, y]$ pairs, denoted by $\gamma_i = [x_i, y_i]$, each indicating the coordinates of a particular salient point. From the neighborhood of each salient point, say γ_i, one may then extract a feature vector $\boldsymbol{\theta_i} = [\theta_{i1}, \theta_{i2}, \ldots, \theta_{id}]$, and a special feature, r_i which is the distance[1] to some particular reference point, Ω.

Each shape model also has a reference point. The objective of the recognition process is to find all instances of the reference point in the observed image, if any exist. The choice of the feature vector $\boldsymbol{\theta}$ is described in section 4.

Define a shape model, ϕ, as a function $\phi : \mathbb{R} \times \mathbb{R}^d \rightarrow [0, 1]$ such that $\phi_m(r, \boldsymbol{\theta})$ reports the likelihood[2] that the feature $\boldsymbol{\theta}$ occurs in the m^{th} model at a distance r from the reference point. Construction of a model for a shape is discussed in section 3.

Given a collection of models, $\{\phi_1, \phi_2, \ldots \phi_M\}$, we seek a method to determine if there are instances of a particular model in image I. Without loss of generality, assume the images are sampled and square, $n \times n$, $I : \mathbb{Z} \times \mathbb{Z} \rightarrow \mathbb{R}$, and let an image I be described by a set of salient points and their corresponding feature vectors.

To determine the quality of possible matches between image I, which has J salient points and model m, construct an $n \times n$ accumulator array $A_m(x, y)$. Then, for the pixel at coordinates x, y, compute

$$A_m(x, y) = \frac{1}{J} \sum_{j=1,J} \phi_m(||(x, y) - \gamma_j||, \boldsymbol{\theta}_j) \tag{1}$$

where γ_j denotes the (x, y) coordinates of salient point j in the image and $\boldsymbol{\theta}_j$ is the feature vector measured at that point.

Thus, at a pixel (x, y) in the image, we accumulate the likelihood that (x, y) is the reference point of model m, at a distance $||(x, y) - \gamma_j||$ from the salient point, and the salient point has particular feature vector $\boldsymbol{\theta}_j$.

We observe that since ϕ is precomputed, and J is small (typically a few hundred), the matching process is very fast.

[1] Euclidian distance in the image plane.

[2] Some readers may find it helpful to think of the likelihood as the log of the probability, but that interpretation is not necessary to understand the algorithm. Instead, think of it as some general confidence.