Accurate and Efficient Curve Detection in Images: 
The Importance Sampling Hough Transform *

Daniel Walsh
Pennsylvania State University

Adrian E. Raftery
University of Washington

Technical Report No. 388
Department of Statistics
University of Washington
February 2001

*Daniel Walsh is Postdoctoral Research Associate, Department of Anthropology, Pennsylvania State University, University Park, PA 16802. Corresponding author e-mail: dcw11@psu.edu. Adrian E. Raftery is Professor of Statistics and Sociology, Department of Statistics, University of Washington, Box 354322, Seattle, WA 98115-4322. E-mail: raftery@stat.washington.edu; Web: www.stat.washington.edu/raftery.
Abstract

The Hough transform is a well known technique for detecting parametric curves in images. We place a particular group of Hough transforms, the probabilistic Hough transforms, in the framework of importance sampling. This framework suggests a way in which probabilistic Hough transforms can be improved: by specifying a target distribution and weighting the sampled parameters accordingly to make identification of curves easier. We investigate the use of clustering techniques to simultaneously identify multiple curves in an image. We also use probabilistic arguments to develop stopping conditions for the algorithm. The resulting methodology is called the Importance Sampling Hough Transform (ISHT). We apply our method to both simulated and real data, and compare its performance with that of two much used versions of the Hough transform: the standard Hough transform and the randomized Hough transform. In our experiments, it is more accurate than either of these common methods, and it is faster than the randomized Hough transform.

KEY WORDS: Clustering, Importance sampling, Hough transform, Probabilistic Hough transform, Target distribution.
# Contents

1 Introduction ................................................. 4
   1.1 Notation and Definitions ............................... 4
   1.2 The Standard Hough transform ......................... 4
   1.3 The Randomized Hough transform ....................... 5

2 The Importance Sampling Hough Transform ................. 7
   2.1 Importance Sampling .................................... 7
   2.2 The Algorithm .......................................... 8
   2.3 Importance Sampling Distribution ...................... 9
   2.4 Target Distribution .................................... 9
   2.5 Peak Detection in the Parameter Space via Clustering 10
   2.6 Stopping Conditions .................................... 11

3 Examples ..................................................... 13
   3.1 Simulated Data .......................................... 13
      3.1.1 Implementation .................................... 14
      3.1.2 Results ........................................... 14
      3.1.3 Stopping Conditions ............................... 16
   3.2 Blood Cell Data ......................................... 17
      3.2.1 Implementation .................................... 17
      3.2.2 Results ........................................... 19

4 Discussion .................................................. 21
1 Introduction

The Hough transform is a well known technique for detecting parametric curves in images. We place a particular group of Hough transforms, the probabilistic Hough transforms, in the framework of importance sampling. This framework suggests a way in which probabilistic Hough transforms can be improved: by specifying a target distribution and weighting the sampled parameters accordingly to make identification of curves easier. We investigate the use of clustering techniques to simultaneously identify multiple curves in the image. We also use probabilistic arguments to develop stopping conditions for the algorithm. Results from applying our method and two popular versions of the Hough transform to both simulated and real data are shown.

1.1 Notation and Definitions

The Hough transform (Hough 1962), hereafter HT, is typically used to detect object boundaries in images. Before applying the HT to a particular image, the image must be transformed by an edge detection algorithm into a binary edge detected image (also referred to as the image space). An edge detection algorithm assigns each pixel to be a foreground pixel (edge pixel), or a background pixel. In all the examples presented here the foreground pixels are black and the background pixels are white. We will use the term points to refer to foreground pixels and the term pixels to refer to foreground and background pixels. We define \( N \) to be the total number of points in the image space.

In order to detect instances of a particular parametric curve in an image, one must decide upon a parameterization of the curve. We will denote the curve parameterization by \( \Theta \). The dimension of the curve will be denoted by \( p \). For instance, if a line is parameterized in Cartesian coordinates, \( p = 2 \), and \( \Theta = \{ m, c \} \), where \( m \) and \( c \) are the slope and intercept of the line, respectively.

Several of the HTs we discuss use an accumulator array to detect curves. An accumulator array, denoted by \( \mathcal{A} \), is a discretization of the parameter space, \( \Theta \). Each cell of \( \mathcal{A} \) is used to store votes for curves whose parameters are contained within the cell.

1.2 The Standard Hough transform

The standard HT (SHT) is a popular method for detecting parametric curves (lines, circles, ellipses etc) in binary image data. Good introductions to the HT and surveys of the literature can be found in Illingworth and Kittler (1988) and Leavers (1993). We will briefly describe the SHT for line detection. Polar coordinates \( (p = x \cos \theta + y \sin \theta) \), are most often used to parameterize lines (Duda and Hart 1972). Given this parameterization, the parameters of all lines passing through a particular point \((x', y')\) in the image space form a sinusoid in the parameter space. The standard HT for detecting lines proceeds by mapping each point in
the image space to its sinusoid in the discretized parameter space (the accumulator array, $\mathcal{A}$). Each cell in the accumulator array is incremented once for each sinusoid that passes through it. A peak detection method, sometimes a simple threshold, is used to locate local peaks in the accumulator array. The location of each peak gives the parameters of each detected line.

1.3 The Randomized Hough transform

The main problems of the standard HT are its long computation times and large storage requirements. The long computation times are caused by the fact that the HT increments the cells in the accumulator array corresponding to all curves that pass through all points in the image. Thus, much of the computation time is spent storing votes for curves with very little support from the data. The storage requirements of the standard Hough transform are a bigger problem though, since the size of the accumulator array is exponential in the number of parameters. When the number of parameters is greater than two, the storage space requirements become excessive.

A new class of HTs called probabilistic Hough transforms (PHTs) attempted to address these problems by using random sampling of edge points, and many-to-one mapping of edge points from the image space to the parameter space. The simplest PHT is the randomized HT (RHT) due to Xu, Oja, and Kultanen (1990). Below we outline the RHT algorithm for the general case of detecting all instances of a particular $p$-dimensional curve in a binary image.

1. Create the set $E$ of all edge points in the binary edge detected image.
2. Select $p$ points, $(e_1, \ldots, e_p)$, at random from $E$.
3. Solve for the parameters ($\theta$) of the curve in the image space, defined by the selected points.
4. Increment the appropriate cell, $\mathcal{A}(\theta)$, in an accumulator array.
5. If the $\mathcal{A}(\theta)$ exceeds a predefined threshold $t$, then the curve parameterized by $\theta$ is detected. When this happens, the points that lie on the curve are removed from $E$ and the accumulator array is re-initialized.
6. Regardless of whether a curve is detected, check whether or not a stopping condition is satisfied. If it is not, return to step 2.

The two main ingredients in this algorithm (and all PHTs) are a sampling mechanism (steps 2-3) and a peak detection method (steps 4-5).
The sampling mechanism defines a sampling distribution on the continuous parameter space, $\Theta$. However, the sampling mechanism is discrete in nature, so we can view the sampling mechanism as defining a discrete sample space $\Omega \subset \Theta$, and a sampling distribution (probability mass function), $g(\cdot)$, on $\Omega$. In the RHT, the sampling mechanism is the selection of $p$ points at random. Thus, if there are $K = \binom{N}{p}$ $p$-tuples of points and $\theta_i$ is the parameter for the $i^{th}$ $p$-tuple, then $\Omega = \{\theta_1, \ldots, \theta_K\}$ and the sampling distribution can be written as:

$$g(\theta^*) = \frac{1}{K} \sum_{i=1}^{K} 1_{\{\theta^* = \theta_i\}} \quad \theta^* \in \Omega.$$ 

If all the $\theta_i$'s are unique, then $g(\cdot)$ is a uniform distribution on $\Omega$, i.e.

$$g(\theta^*) = \frac{1}{K} \quad \theta^* \in \Omega.$$ 

Curves that are present in the image should correspond to regions in $\Theta$ that have relatively large probability under $g$. The peak detection method of the RHT achieves this by grouping sampled parameter values into cells in the accumulator array and comparing with a threshold.

Applying the RHT is not straightforward if the curve is nonlinear with respect to its parameters, in which case selecting $p$ points does not necessarily uniquely define a $p$-dimensional curve. An ellipse is an example of a curve that is nonlinear with respect to its parameters. A solution to the problem of detecting ellipses within the framework of PHTs has been explored by McLaughlin (1998) (based on work by Yuen, Illingworth, and Kittler (1989) in a non-PHT framework). In his method, three points are selected at a time. Estimates of the tangents to the (possible) ellipse at these points are then calculated and used to interpolate the center of the ellipse. Given the location of the ellipse center the remaining parameters are easily determined.

A greater problem affecting the RHT is that its performance is poor when the image is noisy or complex. New PHTs have been proposed to improve the performance of the RHT. The simplest modification to the RHT is the RHT with point distance criterion (RHT-D). In this HT, at each iteration, one point is sampled at random and then $p - 1$ points are sampled uniformly from all points that are within certain a distance (greater than $d_{\text{min}}$ and less than $d_{\text{max}}$) of the first point. Sensible choices of $d_{\text{min}}$ and $d_{\text{max}}$ lead to an increase in the probability of sampling points on a curve present in the image.

Comparisons of various probabilistic (and non-probabilistic) HTs can be found in Kälviäinen, Hirvonen, Xu, and Oja (1995) and Kälviäinen and Hirvonen (1997). These methods include the RHT-D mentioned above, the Dynamic RHT, the Random Window RHT, the Window RHT, the Connective RHT, and the Dynamic Combinatorial HT. Most of these modifications to the RHT attempt to improve the sampling distribution in various ways so that it is more peaked around the parameters corresponding to the curves in the image, thus making
the subsequent process of peak detection easier. In some cases the distinction between the sampling mechanism and peak detection method is blurred, e.g. in the DRHT where curves are detected by an iterative process involving two RHTs.

While all of these methods differ in the sampling distribution employed they all share two common traits: (i) they all detect curves sequentially, and (ii) curves are detected when the number of votes in a cell in the accumulator array exceeds a certain threshold, or in other words, when a curve has been sampled a certain number of times. We see these as areas where improvements can be made. When detecting curves sequentially, every time a curve is detected the accumulator array is reset, thus discarding other curves that have already been accumulated. When accumulating a curve in the accumulator array, typically no effort is made to assess the “quality” of the entire curve. A practical consequence of this is that a curve must be sampled many times to be detected.

In our approach we will use a criterion for judging the “quality” of a curve, and introduce a technique that allows detection of multiple curves simultaneously. The form of this criterion is suggested by considering the following question: “What distribution would I ideally wish to sample from?” If we can define an ideal sampling distribution and obtain a sample from \( g(\cdot) \), then we can obtain a sample from the ideal distribution via the technique of importance sampling.

In the following sections we introduce importance sampling and then show how it relates to the HT. We present some simple rules-of-thumb for determining how many samples from the sampling distribution are needed and suggest using clustering techniques to simultaneously identify curves. Results from applying these ideas to both simulated and real data are shown, and our method is compared to two standard HTs.

## 2 The Importance Sampling Hough Transform

### 2.1 Importance Sampling

Importance sampling is a technique that can be useful when a sample from a particular target distribution is desired but simulation from that distribution is not straightforward. We will restrict our attention to the case where the target distribution is discrete and known only up to a multiplicative scale factor.

Consider a discrete random variable, \( \theta \), with probability mass function \( \{ \pi(\theta) : \theta \in \Omega \} \) where \( \Omega \) is the sample space of \( \theta \). We wish to obtain a sample \( \theta_1, \ldots, \theta_T \) from \( \pi(\cdot) \). We know the form of \( \pi(\cdot) \) up to a scale factor, i.e.:

\[
\pi(\theta) = \frac{f(\theta)}{c}, \quad \theta \in \Omega,
\]

where the form