FEATURE DETECTION IN HIGHLY NOISY IMAGES USING RANDOM SAMPLE THEORY

Ali Cafer Gurbuz, James H. McClellan and Waymond R. Scott, Jr.

School of Electrical & Computer Engineering
Georgia Institute of Technology, Atlanta, GA USA

ABSTRACT
A novel feature detection algorithm utilizing random sample theory is proposed for 2D and 3D images. The proposed method works on both binary and gray-scale images and yields faster results than standard feature detection algorithms, such as the Hough Transform (HT), while keeping the performance level of HT. The proposed method creates random hypothesis features and tests them to select candidate features in the image. The selected candidate features are then re-estimated within a smaller search space around the candidate feature. The proposed algorithm has been tested on both simulated and experimental subsurface seismic and GPR images to locate linear features like pipes or tunnels. Results show that the proposed algorithm can detect features accurately and much faster than conventional methods.

Index Terms— RANSAC, Hough Transform, Fast line detection, Subsurface imaging

1. INTRODUCTION
The Hough Transform (HT) [1], its variants and their generalizations [2–5] are the most commonly used methods capable of detecting lines [3], circles [4], or other parameterized curves [6]. The HT uses a parameterized model of each feature to transform the feature in the original image space into a single mesh point in the HT parameter space. The better a feature matches the model, the higher its value will be at a mesh point. Features having values above a predetermined threshold are labeled as a detection. Although the HT is an effective algorithm even in very noisy images, it is not easily implementable because of its high computation time and large memory requirements. These problems are exacerbated when moving from two to three dimensions.

Various methods have been proposed to decrease the computational requirements of the HT, the primary ones being the Probabilistic Hough Transform (PHT) [7], the Randomized Hough Transform (RHT) [8], and Line Detection using Random Sample Consensus (RANSAC) [9–11]. The PHT uses a randomly selected subset of edge points in the image as input for the HT; however, this technique can lead to erroneous results if the subset is too small, and selection of the optimum subset size requires a priori knowledge of the image.

The RHT randomly selects \( n \) pixels, solves for the feature parameters and then increases the value of the parameter space cell by one. Using this many-to-one mapping together with randomization saves on memory requirements as well as computation time. The RHT is suitable for low noise images [12]. For highly noisy images, PHT outperforms RHT, but neither algorithm works very well in detecting buried features in extremely noisy subsurface images.

To improve the robustness of feature detection, the RANSAC [9] algorithm was proposed. In RANSAC \( n \) edge points are randomly selected, and the features of a line are determined from the \( n \) points. The data lying within a defined distance from the line are classified as “inliers” while the remaining data are marked as “outliers.” If the number of inliers is larger than a predetermined threshold, the feature parameters are re-estimated using only the inliers. In this way, the effect of misleading outliers is mitigated. Furthermore, the memory requirements are much less for the RANSAC algorithm since an accumulator array is not utilized. RANSAC has been shown to perform line detection faster than the HT [10, 13].

Although the RANSAC algorithm robustly detects features, it can only be applied to binary images, not gray-scale subsurface images. Adapting the subsurface images to RANSAC using binarization may seem like a viable solution, but in fact the loss of important data degrades the performance of the algorithm, especially in high noise images. The goal of the proposed algorithm is to detect features in highly noisy gray-scale images more robustly and faster than the HT without the degradation of performance suffered with the PHT or RHT algorithms.

2. PROPOSED METHOD
The basic idea of the new algorithm is to first find rough areas or volumes in the image that possibly include features, and then search only these rough regions with a more accurate algorithm like the HT. Reducing the search space of the HT decreases the computation time, while still maintaining detection performance at a level comparable to that of the HT.

2.1. Algorithm Steps
The proposed method has two stages; the first stages searches for feature, while the second stage refines the estimate of the features.

Stage I (Candidate Model Selection)
(i) The feature to be detected is denoted by a set of parameters \( P = (p_1, p_2, \ldots, p_n) \), where each parameter \( p_i \) has limits defined by \( R_i \). For example, a line in 2D is denoted by two parameters, \( (\rho, \theta) \), because it can be expressed as \( \rho = x \cos \theta + y \sin \theta \). Line parametrization in 3D and parameter ranges can be found in [6].

(ii) Randomly generate a feature by selecting its parameters with a random variable uniformly distributed over its range. In other words, \( p_i \sim U[R_i] \) with \( i = 1, 2, \ldots, n \). The set of parameters \( P \) makes it possible to instantiate the feature in the image.

In RANSAC and RHT, random features are selected by randomly choosing \( n \) points from the image and solving for the parameters \( (p_1, p_2, \ldots, p_n) \). While this might be time efficient for binary images where edge points do not constitute a major part of the image, for gray-scale and higher dimensional images, this requires a
2.2. Selection of Algorithm Parameters

Selection of the parameter $\sigma$ is important because it determines the number of trials, $N$, in Stage I, as well as the size of the search space, $\Delta P$, in Stage II. The relative size of $\sigma$ with respect to the size of the image, $s_1$, is a crucial parameter. Selecting a very small $\sigma/s_1$ ratio will increase $N$ and reduce $\Delta P$. In the limit $\sigma \to 0$, the algorithm approaches the RHT. Conversely, when the ratio $\sigma/s_1$ increases, $\Delta P$ will increase, while the value of $N$ will decrease. In the limit $\sigma \to s_1$, the number of iterations $N$ in Stage I will be 1, making the candidate model selection stage useless and reducing the proposed method to the HT applied in Stage II to the whole image.

Once $\sigma/s_1$ is selected, $\Delta P$ can be computed from $\sigma/s_1$ as follows:

$$\Delta \theta = \arctan \left( \frac{\Delta y}{\Delta x} \right) \quad \Delta \phi = \arctan \left( \frac{\Delta y}{\Delta x} \right)$$

$$\Delta \rho = \sigma \quad \Delta u = \sigma \quad \Delta v = \sigma$$

(1)

Let $N$ be the number of trials required to have at least one feature within $\Delta P$ of the true feature parameter with probability $q$. If the selected parameters fall beyond $\Delta P$, the feature cannot be correctly detected since Stage II only searches for features within $\Delta P$. The number of trials $N$ for picking a random feature within $\Delta P$ with probability $q$ is

$$N = \frac{\log (1-q)}{\log \left( \prod_{i=1}^{N} R(p_i) - 2^k \prod_{i=1}^{N} \Delta p_i \right)}$$

(2)

The threshold $T_s$ is used to compare the sum of the image pixel values within a width $\sigma$ of the selected feature. The noise statistics $(\mu_s, \sigma_s^2)$ of the image can be estimated from the image itself. To detect features of an SNR level greater than $m_{SNR}$, the threshold $T_s$ should be selected as

$$T_s = m_{SNR} \sigma_s^2 s_1 + N_{pix} \mu_s$$

(3)

where $N_{pix}$ is the number of pixel points added in the summation region of the selected feature.

3. RESULTS

A 101 x 101 image with two linear structures having the parameter values $\rho_1 = 70$, $\theta_1 = 65°$ and $\rho_2 = 20$, $\theta_2 = 120°$ is shown in Fig. 1(a). The image is formed as sum of Gaussian distributions expressed by $f(i,j) = \sum G(i,j|x_t, y_t, \sigma_{x,y})$ where $x_t, y_t$ are the true object positions in the image and $\sigma_{x,y}$ are the standard deviations on $x$ and $y$ axes. Zero-mean white Gaussian random noise (WGN) is added to the image. The signal-to-noise ratio (SNR) is defined as the ratio of the maximum absolute value of the object signal to the average noise power. A noisy image with SNR = 0 dB is shown in Fig. 1(b). Note that the linear features cannot be seen because they are masked by the noise.

The parameter $\sigma$ is chosen as 10, i.e., $\sigma/s_1 = 0.1$. Using (1) $\Delta \rho = 10$ and $\Delta \theta = 11.2°$. Using (2) for a probability of detection $q = 0.99$, the number of trials $N$ is found to be $N = 527$. The lines estimated from Stage I are shown in Fig. 2. The selected lines are re-estimated in Stage II by doing a HT within a vicinity $(\Delta \rho, \Delta \theta)$ of the parameters of the selected lines. The true parameters of the target lines and the detected parameters are listed in Table 1.

3.1. Performance of the algorithm for different $\sigma$ values:

The effect of the parameter $\sigma$ on the detection performance and run time of the algorithm is analyzed using a Monte Carlo simulation. A 2D image containing a linear structure with parameters $(\rho, \theta) = (70, 65°)$ with SNR = 10 dB is generated. For various $\sigma/s_1$ values, the algorithm was run 200 times and the elapsed times and the detected parameters were recorded. For varying values of $\sigma/s_1$, the average run times of Stage I, Stage II and the average total run time of the algorithm are shown in Fig. 3.

Increasing \( \sigma/s_i \) enlarges the search space \((\Delta \rho, \Delta \theta)\) according to (1), and also raises the computation time of the HT in Stage II. On the other hand, enlarging the search space will make \( N \) smaller (2) resulting in a lower average run time for Stage I. The combination of these two effects yields the total run time seen in Fig. 3(c). For the total run time of the algorithm a \( \sigma/s_i \) value that minimizes the average run time can be found, and \( \sigma/s_i = 0.15 \) is selected as the optimum parameter.

The probability of detection, \( P_D \), with changing \( \sigma/s_i \) values is shown in Fig. 3(d). Even though the detection performance decreases slightly when \( \sigma/s_i > 0.3 \), it is nearly constant for \( \sigma/s_i < 0.3 \). So the optimum \( \sigma/s_i \) which minimizes run time and also maintains the performance level is still \( \sigma/s_i = 0.15 \) for 2D images.

### 3.2. Performance of the proposed method for varying SNR

In this part, the detection performance and average running time of the proposed algorithm for varying levels of SNR is compared to the HT, RHT and PHT for 2D images. Two versions of the RHT and PHT algorithms with low (RHTL, PHTL) and high (RHTH, PHTH) trial numbers and data percentage are used for comparison.

At each SNR value, the proposed algorithms are run 100 times with random noise added to the original signal each time. The detected target parameters and the run time of the algorithms are noted. To fairly compare the detection performance and the run times for all algorithms the same parameter resolution is used. That is, \( \rho_p = 1 \) and \( \theta_p = 2^\circ \). The proposed algorithm uses a \( \sigma/s_i = 0.1 \) ratio. The PHT uses 50\% and 5\% of the data for the PHHT and PHTL results; RHT uses \( 10^4 \) and \( 10^5 \) random point selections for RHTH and RHTL, respectively.

The probability of detection \( (P_D) \) for all six algorithms is given in Fig. 4. These probability of detection curves show that the proposed method (PM) and the HT have nearly the same performance for all SNR values while the PHT and RHT algorithms have much lower \( P_D \). Here the advantage of the proposed method lies in the average run time. The average run times of the algorithms are given in Table 2. It can be seen that while the proposed method (PM) has the same performance level as the HT, the average run time is nearly ten times less than the HT. The algorithms RHTL and PHTL can run faster than proposed method, but they have much worse detection performance. Even the PHHT and RHTH algorithms which have higher average run times than proposed method have worse detection performance. So the proposed method combines the fast running time of a random selection method with the best possible detection performance.

---

**Table 1. True and Detected Target Parameters**

<table>
<thead>
<tr>
<th>2D Line Detection Results</th>
<th>Targets</th>
<th>Target 1</th>
<th>Target 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameters</td>
<td>( \rho )</td>
<td>( \theta (\circ) )</td>
<td>( \rho )</td>
</tr>
<tr>
<td>True Parameters</td>
<td>70</td>
<td>65</td>
<td>20</td>
</tr>
<tr>
<td>Random Selection</td>
<td>71.3</td>
<td>68.3</td>
<td>30.6</td>
</tr>
<tr>
<td>Corrected</td>
<td>69.7</td>
<td>68.3</td>
<td>20.6</td>
</tr>
</tbody>
</table>

**Table 2. Average Run Times (secs) of the Algorithms in 2D**

<table>
<thead>
<tr>
<th></th>
<th>PM</th>
<th>HT</th>
<th>RHTH</th>
<th>RHTL</th>
<th>PHHT</th>
<th>PHTL</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.87</td>
<td>18.75</td>
<td>3.26</td>
<td>0.33</td>
<td>9.02</td>
<td>0.88</td>
<td></td>
</tr>
</tbody>
</table>

*Fig. 2. Detected lines from Stage I (random selection) and Stage II (corrected by the HT) of the proposed method.*

*Fig. 3. (a) Average run time of stage I of the algorithm versus \( \sigma/s_i \). (b) Average run time of stage II of the algorithm, (c) Average run time for the whole algorithm, (d) Probability of detection for varying \( \sigma/s_i \) values.*

*Fig. 4. Probability of detection \( (P_D) \) for all six algorithms vs. SNR.*
Table 3. Experimental Results with 3-D Data Comparing the Proposed Method (PM) to the HT

<table>
<thead>
<tr>
<th>Sensors</th>
<th>GPR</th>
<th>SEISMIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td>PM</td>
<td>HT</td>
</tr>
<tr>
<td>$\theta (^\circ)$</td>
<td>2.87</td>
<td>0</td>
</tr>
<tr>
<td>$\phi (^\circ)$</td>
<td>-1.09</td>
<td>0</td>
</tr>
<tr>
<td>$u$</td>
<td>-14.90</td>
<td>-15</td>
</tr>
<tr>
<td>$v$</td>
<td>61.48</td>
<td>62</td>
</tr>
<tr>
<td>Time (s)</td>
<td>2580</td>
<td>5.33 $\times 10^6$</td>
</tr>
</tbody>
</table>

3.3. Experimental Data Results in 3D

An experimental system to collect co-located ground penetrating radar (GPR) and seismic data was available to investigate the problem of detecting shallow tunnels [14–16]. In the experiments, a scale model of a tunnel was buried in the sandbox, which has a 1.8 m by 1.8 m scanning area. The tunnel is 10 cm in diameter and was buried 58 cm deep. The collected data from the GPR and seismic sensors were backprojected by a migration algorithm to form the 3D subsurface images [14, 17].

The proposed algorithm and the HT have been applied to the GPR and seismic 3D subsurface images. Figure 5 shows isosurface images of the GPR and seismic data along with the detected lines found by the proposed method.

![Isosurface image and detected feature in 3-D migrated data for (a) GPR and (b) Seismic sensors.](image)

The detected parameters of the features in both the proposed method and the HT along with the run times of the algorithms are summarized in Table 3. In higher dimensional images the advantage of the proposed method over the HT with respect to run times will be substantial.

4. CONCLUSIONS

A fast, robust and effective approach for detecting parameterized features in images was demonstrated. The method utilizes random sample theory. Results from simulated and experimental data sets show that the proposed method can keep the same performance level as the HT, while computing the feature detection much faster. The proposed method also outperforms faster HT variants such as the RHT and PHT algorithms. The proposed algorithm is well suited for applications such as detecting pipes, tunnels or other features in 3D subsurface images from seismic and GPR sensors.

5. REFERENCES