Directional edge detection using the logical transform for binary and grayscale images

Ethan E. Danahya, Sos S. Agaian, Karen A. Panetta

Department of Electrical and Computer Engineering, Tufts University
Department of Electrical Engineering, University of Texas/San Antonio

Overview

Edge detection, the process of determining edge pixels within an image, is an important task in feature-based image processing. Most techniques currently used for edge detection are based on identifying abrupt changes in the pixel values, often through analysis of the first or second derivative of the image data.

This paper presents a novel method for edge detection within 2D signals. A binary edge map is generated using Boolean partial derivatives calculated quickly through a logical transform. Computer simulations demonstrate the procedure for three classes of signals: synthetic images (where actual edge maps are known), natural images, and cell-phone images (those taken by a low-resolution, low-quality camera). Results are compared quantitatively (when possible with Pratt’s figure of merit) and visually with six common edge detection techniques: Sobel, Prewitt, Roberts, Laplacian of Gaussian, zero-cross and Canny methods.

Partial Derivatives of Boolean Functions

Given binary function \( f(x) \), there exists a Boolean representation \( x = (x_1, x_2, \ldots, x_n) \):

\[
\frac{\partial f}{\partial x_j} = \begin{cases} 
1 & \text{if } f(x) = 1 \text{ and } f(x_1, \ldots, x_{j-1}, 1, x_{j+1}, \ldots, x_n) = 0 \text{ and } f(x_1, \ldots, x_{j-1}, 0, x_{j+1}, \ldots, x_n) = 0 \\
0 & \text{otherwise}
\end{cases}
\]

Partial derivative of the Boolean function with respect to the variable \( x_j \).

Fast implementation:

\[
\frac{\partial f}{\partial x_j} = \left( \left( \frac{f(x_1, \ldots, x_{j-1}, 1, x_{j+1}, \ldots, x_n)}{2} \right) \oplus \left( \frac{f(x_1, \ldots, x_{j-1}, 0, x_{j+1}, \ldots, x_n)}{2} \right) \right)
\]

Examples:

- Forward transform
- Inverse (implicant expansion)

The Logical Transform

Calculating the partial derivative for Boolean functions via Logical Transform

Algorithm Details

The algorithm for binary images

Synthetic Results

<table>
<thead>
<tr>
<th>Image</th>
<th>proposed method</th>
<th>Sobel</th>
<th>Prewitt</th>
<th>Roberts</th>
<th>Log</th>
<th>zero-cross</th>
<th>Canny</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.939</td>
<td>0.929</td>
<td>0.929</td>
<td>0.929</td>
<td>0.973</td>
<td>0.973</td>
<td>0.935</td>
</tr>
<tr>
<td>B</td>
<td>0.903</td>
<td>0.955</td>
<td>0.937</td>
<td>0.942</td>
<td>0.820</td>
<td>0.820</td>
<td>0.783</td>
</tr>
<tr>
<td>C</td>
<td>0.946</td>
<td>0.794</td>
<td>0.801</td>
<td>0.945</td>
<td>0.820</td>
<td>0.820</td>
<td>0.783</td>
</tr>
<tr>
<td>D</td>
<td>0.905</td>
<td>0.893</td>
<td>0.893</td>
<td>0.890</td>
<td>0.903</td>
<td>0.903</td>
<td>0.903</td>
</tr>
<tr>
<td>E</td>
<td>0.922</td>
<td>0.934</td>
<td>0.932</td>
<td>0.952</td>
<td>0.539</td>
<td>0.539</td>
<td>0.916</td>
</tr>
<tr>
<td>F</td>
<td>0.919</td>
<td>0.819</td>
<td>0.843</td>
<td>0.947</td>
<td>0.838</td>
<td>0.838</td>
<td>0.769</td>
</tr>
</tbody>
</table>

Computer Simulations: Synthetic Images

- Original
- Algorithm output
- Roberts
- zero-cross
- Canny

Computer Simulations: Natural Images

- Original
- Algorithm output
- Roberts
- zero-cross
- Canny

Computer Simulations: Cell-Phone Images

- Original
- Algorithm output
- Roberts
- zero-cross
- Canny

Conclusion

Presented is a new edge detection technique using partial derivatives of Boolean functions, as generated via the logical transform. Applicable to both binary and grayscale images, this method is able to capture the main image features, while not obscuring the finer details. A variety of both binary and grayscale images were tested, and the method presented here was found to be competitive with the comparison techniques. While the comparison methods were strong in either detecting major edges or preserving finer details, none were seen to cover both aspects comprehensively. This method simultaneously maintained both these desired attributes.