Image Understanding

WS 01/02

Lecture 7/8: Edge Detection
1. Background ideas
2. Edge profiles
3. Edge Detection
4. Derivative Operators
5. Roberts, Prewitt, Sobel masks
6. Compass masks
7. Laplacian operators
8. Combined detection
9. Extended masks
Literature:

Lecture notes: Section 4

Background

1. Noise Smoothing
2. Edge Localisation
3. Edge Enhancement
   - Thinning wide edges
   - Thresholding
4. 

Questions:
1. high-frequency
2. gradient operator
Edge Types

Ideal

Step edges

Ramp edges

Ridge edges (line)

Roof edges

Noisy
Ideal vs. Real Edge

Intensity vs. Coordinates

Intensity vs. Coordinates

University of Stuttgart, Germany
Institute of Parallel and Distributed High-Performance Systems (IPVR)
Edge Descriptors

Edge direction: Tangent to the contour of the edge

Edge Normal: Direction of maximum intensity variation at the edge point

Edge Position: The image position at which the edge is located

Edge Strength: Measure of local contrast across the edge
Edge Detection Operators

Based on the idea that edge information in an image, is found by looking at the relationship between a pixel and its neighbours.

i.e. edge is found by discontinuity of grey level values.

An ideal edge detector should produce an edge indication localised to a single pixel located at the mid-point of the slope.
Edge Detection: Derivative Operators

\[ f(x) := \sin(x) \]

\[ \frac{\partial}{\partial x} \sin(x) \]
2 major classes of differential edge detection

1. **First order derivative**
   
   Some form of spatial first order differentiation is performed, and the resulting gradient is compared to a threshold value. An edge is judged present if the gradient exceeds the threshold.

2. **Second order derivative**
   
   An edge is judged present if there is a significant spatial change in the polarity of the second derivative.
Edge Detection: Derivative Operators

Original image

Gray-levels of image

First derivative of gray-level

\[ +\text{ve at leading edge of transition} \]
\[ -\text{ve at trailing edge of transition} \]
\[ \text{i.e. magnitude can detect presence of edge} \]

Second derivative of gray-level

\[ +\text{ve for dark side of edge} \]
\[ -\text{ve for light side of edge} \]
\[ \text{i.e. sign = side of edge} \]
The first derivative at any point in an image is obtained by using the magnitude of the gradient at that point.

A change of the image function can be described by a gradient that points in the direction of the largest growth of the image function.

Many are implemented with convolution masks.
Edge Detection: Derivative Operators

\[ \nabla f = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \quad \Rightarrow \quad \text{mag}(\nabla f) = \left\{ \left( \frac{\partial f}{\partial x} \right)^2 + \left( \frac{\partial f}{\partial y} \right)^2 \right\}^{1/2} \]

\[ \text{mag}(\nabla f) = |e - h| + |e - f| \]

<table>
<thead>
<tr>
<th>Image(x,y)</th>
</tr>
</thead>
<tbody>
<tr>
<td>a b c</td>
</tr>
<tr>
<td>d e f</td>
</tr>
<tr>
<td>g h i</td>
</tr>
</tbody>
</table>

|e-h| for x-direction
|e-f| for y-direction
Edge Detectors: Roberts

Image(x,y)

\[
\begin{bmatrix}
    a & b & c \\
    d & e & f \\
    g & h & i \\
\end{bmatrix}
\]

\[\begin{bmatrix}
    1 & 0 \\
    0 & -1 \\
\end{bmatrix} + \begin{bmatrix}
    0 & 1 \\
    -1 & 0 \\
\end{bmatrix}\]

\[|e*1+f*0+h*0-i*1| + |e*0+f*1-h*1+0*i| = |e-i| + |f-h|\]
Edge Detectors: Roberts

\[
\begin{bmatrix}
1 & 0 \\
0 & -1 \\
\end{bmatrix} \quad \begin{bmatrix}
0 & 1 \\
-1 & 0 \\
\end{bmatrix}
\]

Marks the edge points only
Works best with binary images
Use each in turn to produce two numbers: \( r, c \) for each pixel

\[
| I(r,c) - I(r-1, c-1) | + | I(r,c-1) - I(r-1,c) |
\]

very sensitive to the effects of noise
Edge Detection: Derivative Operators

\[
\nabla f = \begin{cases} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{cases}
\]

\[
\text{mag}(\nabla f) = \left\{ \left( \frac{\partial f}{\partial x} \right)^2 + \left( \frac{\partial f}{\partial y} \right)^2 \right\}^{1/2}
\]

with 3*3 neighbourhood

\[
\text{mag}(\nabla f) = \left| (g + h + i) - (a + b + c) \right| + \\
+ \left| (c + f + i) - (a + d + g) \right|
\]
Edge Detectors: Prewitt

\[
\begin{bmatrix}
-1 & -1 & -1 \\
0 & 0 & 0 \\
1 & 1 & 1
\end{bmatrix} \quad \text{Row}
\]

\[
\begin{bmatrix}
-1 & 0 & 1 \\
-1 & 0 & 1 \\
-1 & 0 & 1
\end{bmatrix} \quad \text{Column}
\]

Use each in turn to produce two numbers: $r$, $c$ for each pixel

Edge Magnitude $\sqrt{r^2 + c^2}$

Edge Direction $\tan^{-1} \left[ \frac{r}{c} \right]$  Note: This is the direction of the normal
Edge Detectors: Sobel

\[
\begin{bmatrix}
-1 & -2 & -1 \\
0 & 0 & 0 \\
1 & 2 & 1
\end{bmatrix}
\quad \begin{bmatrix}
-1 & 0 & 1 \\
-2 & 0 & 2 \\
-1 & 0 & 1
\end{bmatrix}
\]

Row \quad \text{Column}

Use each in turn to produce two numbers: \( r, c \) for each pixel

\[
\text{Edge Magnitude} \quad \sqrt{r^2 + c^2}
\]

\[
\text{Edge Direction} \quad \tan^{-1} \left( \frac{r}{c} \right)
\]

Note: This is the direction of the normal

also provides a smoothing effect so is good for reducing the effects of noise
What is happen if we will use only one of these masks:

\[
\begin{bmatrix}
-1 & -1 & -1 \\
0 & 0 & 0 \\
1 & 1 & 1 \\
\end{bmatrix}
\]

\[
\begin{bmatrix}
-1 & 0 & 1 \\
-1 & 0 & 1 \\
-1 & 0 & 1 \\
\end{bmatrix}
\]
Compass operators

\[
\begin{align*}
(N) & : \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix} \\
(NW) & : \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & -1 \\ 0 & -1 & -1 \end{bmatrix} \\
(W) & : \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix} \\
(SW) & : \begin{bmatrix} 0 & -1 & -1 \\ 1 & 0 & -1 \\ 1 & 1 & 0 \end{bmatrix} \\
(S) & : \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \\
(SE) & : \begin{bmatrix} -1 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} \\
(E) & : \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \\
(NE) & : \begin{bmatrix} 0 & 1 & 1 \\ -1 & 0 & 1 \\ -1 & -1 & 0 \end{bmatrix}
\end{align*}
\]
Robinson Compass masks

\[
\begin{bmatrix}
-1 & 0 & 1 \\
-2 & 0 & 2 \\
-1 & 0 & 1 \\
\end{bmatrix}
\quad \begin{bmatrix}
0 & 1 & 2 \\
-1 & 0 & 1 \\
-2 & -1 & 0 \\
\end{bmatrix}
\quad \begin{bmatrix}
1 & 2 & 1 \\
0 & 0 & 0 \\
-1 & -2 & -1 \\
\end{bmatrix}
\quad \begin{bmatrix}
2 & 1 & 0 \\
1 & 0 & -1 \\
0 & -1 & -1 \\
\end{bmatrix}
\quad \begin{bmatrix}
1 & 0 & -1 \\
2 & 0 & -2 \\
1 & 0 & -1 \\
\end{bmatrix}
\quad \begin{bmatrix}
0 & -1 & -2 \\
1 & 0 & -1 \\
2 & 1 & 0 \\
\end{bmatrix}
\quad \begin{bmatrix}
-1 & -2 & -1 \\
-1 & -2 & -1 \\
-2 & -1 & 0 \\
\end{bmatrix}
\quad \begin{bmatrix}
-2 & -1 & 0 \\
-1 & 0 & 1 \\
0 & 1 & 2 \\
\end{bmatrix}
\]

Edge magnitude is the maximum value found from all masks.

Edge direction by the mask with the maximum value for a pixel.
Kirsch Compass masks

Vertical edge

\[
\begin{bmatrix}
-3 & -3 & 5 \\
-3 & 0 & 5 \\
-3 & -3 & 5 \\
\end{bmatrix}
\begin{bmatrix}
-3 & 5 & 5 \\
-3 & 0 & 5 \\
-3 & -3 & -3 \\
\end{bmatrix}
\begin{bmatrix}
5 & 5 & 5 \\
5 & 0 & -3 \\
-3 & -3 & -3 \\
\end{bmatrix}
\begin{bmatrix}
5 & 5 & -3 \\
5 & 0 & -3 \\
-3 & -3 & -3 \\
\end{bmatrix}
\]

Diagonal NWSE

\[
\begin{bmatrix}
5 & -3 & -3 \\
5 & 0 & -3 \\
5 & -3 & -3 \\
\end{bmatrix}
\begin{bmatrix}
-3 & -3 & -3 \\
5 & 0 & -3 \\
5 & 5 & -3 \\
\end{bmatrix}
\begin{bmatrix}
-3 & -3 & -3 \\
-3 & 0 & -3 \\
5 & 5 & 5 \\
\end{bmatrix}
\begin{bmatrix}
-3 & -3 & -3 \\
-3 & 0 & 5 \\
-3 & 5 & 5 \\
\end{bmatrix}
\]

Edge magnitude is the maximum value found from all masks
Edge direction by the mask with the maximum value for a pixel
Edge Detection: Laplacian

The second derivative at any point in an image is obtained by using the Laplacian.

The second-order derivative of \( f(x, y) \)

\[
\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}
\]

in practice most frequently encountered in form:

\[
\nabla^2 f = 4e - (b + d + f + h)
\]
Laplacian Operators

Unlike compass masks, the Laplacian masks are rotationally symmetric, which means edges at all orientations contribute to the result.

They are applied by selecting one mask and convolving it with the image. The sign of the result ( +ve or –ve ) from 2 adjacent pixel locations provides directional information, and tells us which side of the edge is brighter.

very sensitive to noise
Masks – which to use

Note:

1. if we are only interested in edge information, the sum of the coefficients that make up the mask should be zero.

2. if we want to retain most of the information that is in the original image, the coefficients should sum to a number greater than 0. The larger this sum, the less the processed image is changed from the original image.
Edge Operator

a. Original image.
b. Sobel operator.
c. Prewitt operator.
d. Frei-Chen operator, edge subspace.
e. Laplacian operator.
f. Kirch operator.
g. Roberts operator.
h. Robinson operator.
Edge Operator + noise

a. Original image.
b. Image with added gaussian and salt-and-pepper noise.
c. Sobel with $3 \times 3$ mask.
e. Prewitt with $3 \times 3$ mask.
Extending the masks

We could pre-process the image with various spatial filters to remove, or hide some of the effects from the noise, or we could expand the edge detection operators themselves to reduce noise effects.

\[
\begin{bmatrix}
-1 & -1 & -1 & -2 & -1 & -1 & -1 \\
-1 & -1 & -1 & -2 & -1 & -1 & -1 \\
-1 & -1 & -1 & -2 & -1 & -1 & -1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 1 & 2 & 1 & 1 & 1 \\
1 & 1 & 1 & 2 & 1 & 1 & 1 \\
1 & 1 & 1 & 2 & 1 & 1 & 1
\end{bmatrix}
\]

Sobel 7x7 Row Mask

\[
\begin{bmatrix}
1 & 1 & 1 & 0 & -1 & -1 & -1 \\
1 & 1 & 1 & 0 & -1 & -1 & -1 \\
1 & 1 & 1 & 0 & -1 & -1 & -1 \\
1 & 1 & 1 & 0 & -1 & -1 & -1 \\
1 & 1 & 1 & 0 & -1 & -1 & -1 \\
1 & 1 & 1 & 0 & -1 & -1 & -1 \\
1 & 1 & 1 & 0 & -1 & -1 & -1
\end{bmatrix}
\]

Prewitt 7x7 Row Mask

Better performance than 3x3 masks with noisy images, but require more computations and tend to blur the edges slightly.
Edge Operator $7 \times 7 + \text{noise}$

a. Original image.
b. Image with added gaussian and salt-and-pepper noise.
d. Sobel with $7 \times 7$ mask.
f. Prewitt with $7 \times 7$ mask.
Content of this image: more lines or more edges?
Combined detection

Let $z = \begin{cases} z_1 \\ z_2 \\ \vdots \\ z_9 \end{cases}$ be an image, and $w = \begin{cases} w_1 \\ w_2 \\ \vdots \\ w_9 \end{cases}$ be a mask, and

Application (convolution)

$$R = \sum_{i=1}^{9} w_i z_i = w^T z$$
Combined detection

In orthogonal basis \( \mathbf{w}_1, \mathbf{w}_2 \) product:

\[
\mathbf{w}_1^T \mathbf{z} = \|\mathbf{w}_1\| \|\mathbf{z}\| \cos \Theta
\]

If:

\( \|\mathbf{w}_1\| = 1 \)

then:

\[
\mathbf{w}_1^T \mathbf{z} = \|\mathbf{z}\| \cos \Theta
\]

projection of \( \mathbf{z} \) onto \( \mathbf{w}_1 \)
Combined detection

Let $\mathbf{w}_1$, $\mathbf{w}_2$ for lines, $\mathbf{w}_3$ for edges detection

\[
\mathbf{w}_2^T \mathbf{z} = \| \mathbf{z} \| \cos \Theta
\]
\[
\mathbf{w}_3^T \mathbf{z} = \| \mathbf{z} \| \cos \Theta
\]

The angel between $\mathbf{z}$ and each of these two projection indicates whether $\mathbf{z}$ is closer to line or the edge subspace.
Frei-Chen masks

**Edge subspace**
\[
\begin{bmatrix}
1 & 2 & 1 \\
0 & 0 & 0 \\
-1 & -\sqrt{2} & -1 \\
\end{bmatrix}
\begin{bmatrix}
\frac{1}{2}\sqrt{2} & \sqrt{2} & 0 \\
\sqrt{2} & 0 & -2 \\
1 & 0 & -1 \\
\end{bmatrix}
\begin{bmatrix}
0 & -1 & \sqrt{2} \\
1 & 0 & -1 \\
-\sqrt{2} & 1 & 0 \\
\end{bmatrix}
\begin{bmatrix}
\frac{1}{2} & -1 & 0 \\
-1 & 0 & 1 \\
0 & -1 & -2 \\
\end{bmatrix}
\]

**Line subspace**
\[
\begin{bmatrix}
1 & -2 & 1 \\
-2 & 4 & -2 \\
1 & -2 & 1 \\
\end{bmatrix}
\begin{bmatrix}
\frac{1}{2} & -1 & 0 \\
0 & 0 & 0 \\
1 & 0 & 1 \\
\end{bmatrix}
\begin{bmatrix}
1 & -2 & 1 \\
-2 & 4 & -2 \\
1 & -2 & 1 \\
\end{bmatrix}
\begin{bmatrix}
-2 & 1 & -2 \\
1 & 4 & 1 \\
-2 & 1 & -2 \\
\end{bmatrix}
\]

**Average subspace**
\[
\begin{bmatrix}
1 & 1 & 1 \\
1 & 1 & 1 \\
1 & 1 & 1 \\
\end{bmatrix}
\begin{bmatrix}
\frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\
\frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\
\frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\
\end{bmatrix}
\begin{bmatrix}
\frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\
\frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\
\frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\
\end{bmatrix}
\begin{bmatrix}
\frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\
\frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\
\frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\
\end{bmatrix}
\]

University of Stuttgart, Germany
Institute of Parallel and Distributed High-Performance Systems (IPVR)
Figure 7.13  (a) Original image; (b)–(f) projections onto $w_1$, $w_2$, $w_3$, $w_4$, and $w_5$ subspaces, respectively. (From Hall and Frei [1976].)

Figure 7.13  (Continued) (g)–(j) projections onto $w_6$, $w_7$, $w_8$, and $w_9$ subspaces; (k) magnitude of projection onto edge subspace; (l) magnitude of projection onto line subspace. (From Hall and Frei [1976].)
Template matching

Given is an image. You should extract the edges in the most represented direction. How to do it?

- Apply a mask \([i]\)
- Calculate intensity \(a[i]\)

Repeat \(n\)-times

- Find \(a[i] == \text{max}\)
- Apply a mask \([i]\)
Questions that you have to answer after this lecture

1. How to extract edge information?
2. Explain the underlying principles of edge detection
3. Describe type of edges
4. Explain the underlying principles of derivative operators
5. Explain the difference between Roberts, Prewitt and Sobel masks
6. How to extract an oriented edge?
7. Explain the underlying principles of Laplacian operators
8. How to perform a combined detection
Point of the next lecture

Edge linking, edge following, etc.