Computer Vision-IT416

Dinesh Naik
Department of Information Technology,
National Institute of Technology Karnataka, India

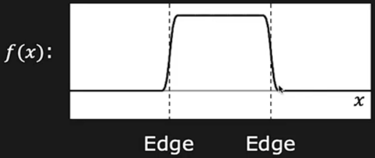
April 5, 2022

Edge detection Using Gradients

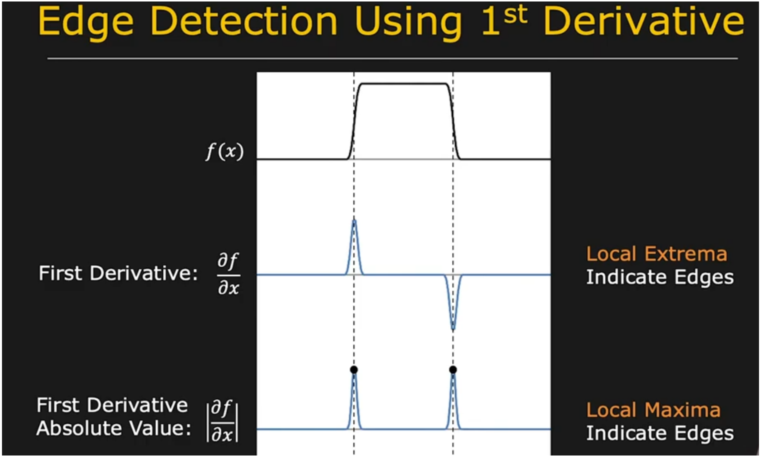
- An image gradient is a directional change in the intensity or color in an image.
- The gradient of the image is one of the fundamental building blocks in image processing.
- Mathematically, the gradient of a two-variable function (here the image intensity function) at each image point is a 2D vector with the components given by the derivatives in the horizontal and vertical directions.
- The most common way to approximate the image gradient is to convolve an image with a kernel, such as the Sobel operator or Prewitt operator.

1D Edge Detection

Edge is a rapid change in image intensity in a small region



Basic Calculus: Derivative of a continuous function represents the amount of change in the function.


Edge Detection Using 1st Derivative
$f(x)$
First Derivative: $\frac{\partial f}{\partial x}$
Local Extrema Indicate Edges
First Derivative Absolute Value: $\left|\frac{\partial f}{\partial x}\right|$
Local Maxima Indicate Edges

2D Edge Detection

$I(x, y)$:

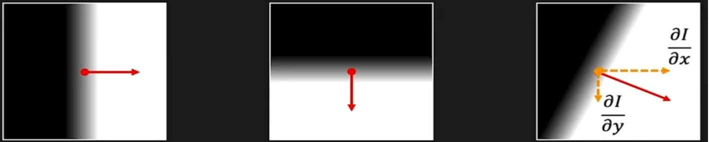
Basic Calculus: Partial Derivatives of a 2D continuous functi
represents the amount of change along each dimension.

Gradient (∇)

Gradient (Partial Derivatives) represents the direction of most rapid change in intensity

$$\boxed{\nabla I = \left[\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y}\right]}$$

Pronounced as "Del I"

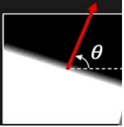


$$\nabla I = \left[\frac{\partial I}{\partial x}, 0\right] \qquad \nabla I = \left[0, \frac{\partial I}{\partial y}\right] \qquad \nabla I = \left[\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y}\right]$$

Gradient (∇) as Edge Detector

Gradient Magnitude $S = \|\nabla I\| = \sqrt{\left(\frac{\partial I}{\partial x}\right)^2 + \left(\frac{\partial I}{\partial y}\right)^2}$

Gradient Orientation $\theta = \tan^{-1}\left(\frac{\partial I}{\partial y} \Big/ \frac{\partial I}{\partial x}\right)$



Discrete Gradient (∇) Operator

Finite difference approximations:

$$\frac{\partial I}{\partial x} \approx \frac{1}{2\varepsilon}\left(\left(I_{i+1,j+1} - I_{i,j+1}\right) + \left(I_{i+1,j} - I_{i,j}\right)\right)$$

$$\frac{\partial I}{\partial y} \approx \frac{1}{2\varepsilon}\left(\left(I_{i+1,j+1} - I_{i+1,j}\right) + \left(I_{i,j+1} - I_{i,j}\right)\right)$$

$I_{i,j+1}$	$I_{i+1,j+1}$
$I_{i,j}$	$I_{i+1,j}$

ε

Can be implemented as Convolution!

$$\frac{\partial}{\partial x} \approx \frac{1}{2\varepsilon} \begin{array}{|c|c|} \hline -1 & 1 \\ \hline -1 & 1 \\ \hline \end{array} \qquad \frac{\partial}{\partial y} \approx \frac{1}{2\varepsilon} \begin{array}{|c|c|} \hline 1 & 1 \\ \hline -1 & -1 \\ \hline \end{array}$$

Comparing Gradient (∇) Operators

Gradient	Roberts	Prewitt	Sobel (3x3)	Sobel (5x5)
$\frac{\partial I}{\partial x}$	$\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$	$\begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} -1 & -2 & 0 & 2 & 1 \\ -2 & -3 & 0 & 3 & 2 \\ -3 & -5 & 0 & 5 & 3 \\ -2 & -3 & 0 & 3 & 2 \\ -1 & -2 & 0 & 2 & 1 \end{bmatrix}$
$\frac{\partial I}{\partial y}$	$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$	$\begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$	$\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$	$\begin{bmatrix} 1 & 2 & 3 & 2 & 1 \\ 2 & 3 & 5 & 3 & 2 \\ 0 & 0 & 0 & 0 & 0 \\ -2 & -3 & -5 & -3 & -2 \\ -1 & -2 & -3 & -2 & -1 \end{bmatrix}$


Gradient (∇) Using Sobel Filter
I.3
Image (I)

Edge Thresholding

Standard: (Single Threshold T)

$\|\|\nabla I(x,y)\|\| < T$	Definitely Not an Edge
$\|\|\nabla I(x,y)\|\| \geq T$	Definitely an Edge

Hysteresis Based: (Two Thresholds $T_0 < T_1$)

$\|\|\nabla I(x,y)\|\| < T_0$	Definitely Not an Edge
$\|\|\nabla I(x,y)\|\| \geq T_1$	Definitely an Edge
$T_0 \leq \|\|\nabla I(x,y)\|\| < T_1$	Is an Edge if a Neighboring Pixel is Definitely an Edge

Canny Edge Detection Algorithm

- Step 1 - Grayscale Conversion.
- Step 2 - Gaussian Blur.
- Step 3 - Determine the Intensity Gradients.
- Step 4 - Non Maximum Suppression.
- Step 5 - Double Thresholding.
- Step 6 - Edge Tracking by Hysteresis.
- Step 7 - Cleaning Up.