Canny Edge Detection

Canny edge detection is a method for detecting image edges. It involves several steps including smoothing with a Gaussian filter, finding the gradient magnitude and direction, applying non-maximum suppression to weak the edges, and finally, using hysteresis thresholding to detect and link edges. The Canny algorithm aims to accurately detect the edges while minimizing false positives and suppressing noise, making it widely used in applications such as object detection, image segmentation, and feature extraction.

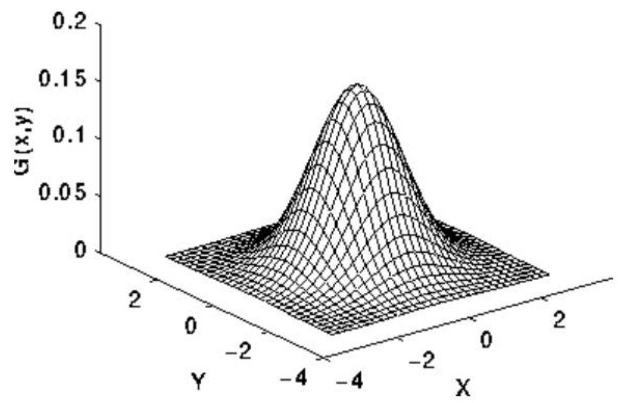
Steps to implement Canny:

1-Remove Noise: Importing libraries of open cv, Dimension of image: 3d image, Shape of image: 3 values in each and every cell, i.e. (width, height, values of RGB). Grayscale : 2d image, have values from 0 to 255 black and white image with variance of Gray.

Gaussian Filter used for removing the noise and make the image smooth for image processing. A 2-D convolution operator called the Gaussian smoothing operator is used to ` blur' pictures and eliminate noise and detail. It is comparable to the mean filter in this regard, but it makes use of a different kernel that depicts a Gaussian (or "bell-shaped").

In 2-D,Gaussian has the form:

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$



2- Sobel Edge Detection: To compute approximations of the derivatives, the operator convolves two 3x3 kernels, one for horizontal changes(sobelx) and one for vertical ones(sobely), with the processed imaged(norm) i.e convert it to gray and applied gaussian filter then normalized the image. The vertical axis is rising upward rather than downward, flipping the vertical kernel in the process to 90degree. Multiply the kernel by the overlapping area of the image element by element at each point.

The calculations are as follows if we designate (norm) as the source picture and Gx and Gy as two images that, at each point, include the approximations for the horizontal and vertical derivatives, respectively:

$$\mathbf{G}_x = \begin{bmatrix} +1 & 0 & -1 \\ +2 & 0 & -2 \\ +1 & 0 & -1 \end{bmatrix}$$ ***Image (norm) =**

$$\mathbf{G}_y = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$ ***image(norm) =**

Here, we define the y-coordinate as increasing in the "down" direction and the x-coordinate as rising in the "right" direction. The gradient magnitude and the angle obtained at each place in the image by combining the generated gradient and direction approximations using the following formula:

$$\mathbf{G} = \sqrt{{\mathbf{G}_x}^2 + {\mathbf{G}_y}^2}$$

$$\mathbf{\Theta} = \operatorname{atan2}(\mathbf{G}_y, \mathbf{G}_x)$$

<u>3-Non maximum and supression</u>: Compute the gradient magnitude and direction using Sobel edge detection. Dividing the gradient orientation into four angles like 0,45,90,135 degree. The algorithm goes through all the points on an image and finds the pixel with maximum value of gradient in the edge directions. The angle of gradient is 3pi/4. Now should check all the front and back pixels in 3x3 matrix. If one of the pixels is greater than the judged pixel keeps the magnitude unchanged else supressed it to 0.

<u>4-Double thresholding</u>: Compute gradient magnitude using Sobel. Using the double thresholding approach, we classify pixels as strong and weak edge pixels depending on the gradient magnitude. We need 2 threshold values, minimum Value and Maximum value. The edges which are below minimum value marked as weak edges and remove them. And values which are more than maximum value make them bright. The values between them are edges or non-edges based on their connectivity, and solve by historsis

<u>5-Edge tracking by hysteresis:</u> Use hysteresis to track edges and connect weak to strong edges. The objective is to create continuous edges by joining pixels with weak edges to those with strong ones.

Difference between canny and other operators.

	Canny edge detection	**Other operators**
Process	The multi-stage approach for canny edge detection consists of smoothing the image using a Gaussian filter, determining the gradient's magnitude and direction, thinning the edges with non-maximum suppression, and detecting and linking edges using hysteresis thresholding.	Operators such as Sobel, Prewitt, and Roberts use a straightforward convolution with a predetermined kernel to directly compute the gradient of the image intensity. Usually, they use the gradient's magnitude to identify edges.
Handling the Noise	Before edge detection, Canny implements a Gaussian smoothing step to assist lower image noise. This makes it more robust to noise compared to other operators.	These operators use the blurring techniques but requires to normalize the image specially in Sobel, Robert, etc.
Thickness	Canny uses non-maximum suppression to create edges with a width of just one pixel. This ensures that edges are represented as thin lines, which is ideal for several applications like object identification and segmentation of images.	Thicker edges may be produced by several alternative edge detection operators, particularly more basic ones like Sobel and Prewitt. This may result in a less accurate edge map, particularly in areas where there are several neighbouring edges.
Thresholding	Hysteresis thresholding is used by canny to categorize pixels as edge or non-edge based on the gradient magnitudes of those pixels. Two thresholds are set: a low threshold and a high threshold. Adaptive thresholding facilitates the handling of images with different contrast and noise levels.	Fixed thresholding techniques, in which a single threshold value is applied uniformly to every pixel in the image, are used by certain other operators. Fixed thresholding techniques, although simpler, could not be as effective in processing images with varying properties and noise levels.