Canny Edge Detection

Canny edge detection is a multi-step algorithm that can detect edges with noise supressed at the same time.

1. Smooth the image with a Gaussian filter to reduce noise and unwanted details and textures.

$$g(m,n) = G_\sigma(m,n) * f(m,n)$$

where

$$G_\sigma = \frac{1}{\sqrt{2\pi\sigma^2}} exp\left(-\frac{m^2+n^2}{2\sigma^2}\right)$$

2. Compute gradient of $g(m,n)$ using any of the gradient operatiors (Roberts, Sobel, Prewitt, etc) to get:

$$M(n,n) = \sqrt{g_m^2(m,n) + g_n^2(m,n)}$$

and

$$\theta(m,n) = tan^{-1}[g_n(m,n)/g_m(m,n)]$$

3. Threshold M:

$$M_T(m,n) = \begin{cases} M(m,n) & \text{if } M(m,n) > T \\ 0 & \text{otherwise} \end{cases}$$

where T is so chosen that all edge elements are kept while most of the noise is suppressed.

4. Suppress non-maxima pixels in the edges in M_T obtained above to thin the edge ridges (as the edges might have been broadened in step 1). To do so, check to see whether each non-zero $M_T(m,n)$ is greater than its two neighbors along the gradient direction $\theta(m,n)$. If so, keep $M_T(m,n)$ unchanged, otherwise, set it to 0.

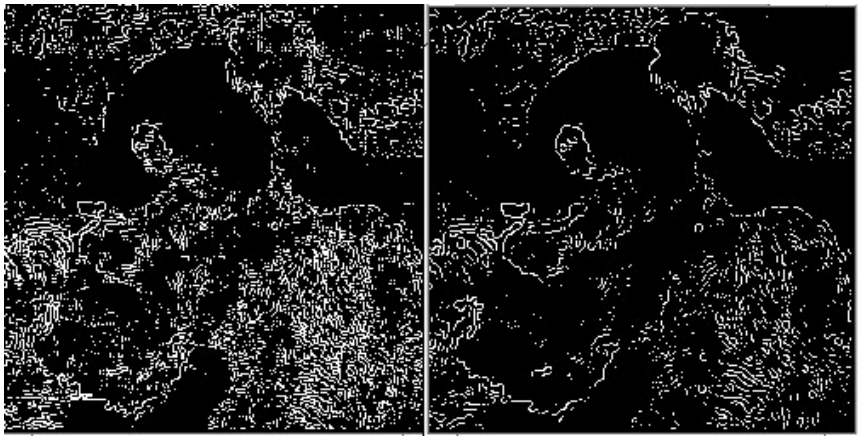
5. Threshold the previous result by two different thresholds τ_1 and τ_2 (where $\tau_1 < \tau_2$) to obtain two binary images T_1 and T_2. Note that T_2 with greater τ_2 has less noise and fewer false edges but greater gaps between edge segments, when compared to T_1 with smaller τ_1.

6. Link edge segments in T_2 to form continuous edges. To do so, trace each segment in T_2 to its end and then search its neighbors in T_1 to find any edge segment in T_1 to bridge the gap untill reaching another edge segment in T_2.

Example 1:

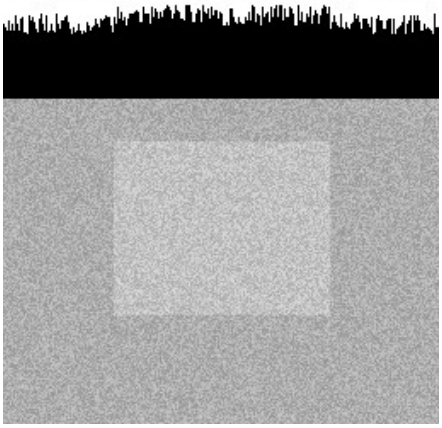
Edge detection by gradient operators (Roberts, Sobel and Prewitt):

Edge detection by LoG and DoG:



Edge detection by Canny method ($\sigma = 1, 2, 3,\ \tau_1 = 0.3,\ \tau_2 = 0.7$):

Example 2:



Edge detection results by Sobel, Prewitt gradient operators, by DoG method and by Canny's method ($\sigma = 5$, $\tau_1 = 0.8$, $\tau_2 = 0.95$):