Programming Assignment-2 report

Aditya Pulekar
CSCI 631: Foundations of Computer Vision

February 29, 2016

Problem 1. Implement the canny edge detection algorithm.

Solution: The Canny Edge Detection Algorithm is a combination of 4 steps:

1. Smoothing:
   An image taken by a camera usually contains some added noise. In order to avoid the false detection of edges in images, it is important to remove noise from the images. A Gaussian filter is applied to the image for smoothing it. A Gaussian kernel of size 3x3 (i.e 9) and standard deviation (σ) of 1.4 is used for this purpose.

2. Finding the X-gradient and Y-gradient for the image:
   Convert the image from the RGB to Gray color space. Using two kernels Gx and Gy,

$$Gx = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

$$Gy = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -2 \end{bmatrix}$$

   image gradients along the x and y directions is calculated for every pixel of the image. These two gradients are a result of convolution of the two kernels with every neighborhood window over the image. Knowing the gradients along the x and y directions about a pixel enables the calculation of gradient angle and the net gradient value for every pixel. The gradient angle is approximated to 0° if the angle is between 0° and 22.5° & between 157.5° and 180°. If the angle is between 22.5° and 67.5°, the angle is approximated to 45°. If the angle is between 67.5° and 112.5°, the angle is approximated is 90°. And, if the angle is between 112.5° and 157.5°, the angle is approximated to 135°.

3. Non-maximum suppression:
   Every pixel, now, has a net gradient value and a gradient angle (i.e. orientation). On traversing through the image, if the current pixel has the maximum value among its neighbors along its orientation, then the value of the current pixel is retained. Else, the value of the current pixel is suppressed to zero. This yields a new non-zero image matrix.

4. Double Thresholding for edge formation:
   The newly obtained image matrix is traversed. Every pixel value is compared to threshold_1 and threshold_2 set for the image. A pixel value is suppressed to zero if the pixel value is lesser than threshold_1.The pixel value is raised to 255 if it exceeds threshold_2. If the pixel value lies between threshold_1 and threshold_2, then the pixel's neighborhood is checked for the presence of a strong pixel (i.e. a pixel having a value greater than threshold_2). This way strong and weak edges in the image are determined. the final image is the canny edge detected image.

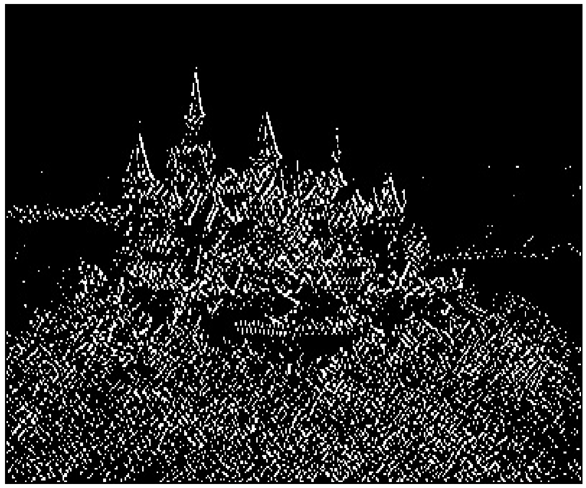

Figure 1: Canny Edge Detected Image

□

Problem 2. Implement Bilateral Filtering to smooth the image.

Solution: Bilateral filter is a noise reducing and edge preserving filter. For this purpose, the intensity value of every pixel is replaced by the weighted average of all the nearby pixels. These weights depend not only on the pixel's color intensity differences but also on the euclidean differences between the pixel coordinates in the image.Note that these weights are Gaussian kernels. This is how the edges in the images are retained.The equation for the bilateral filter is as follows:

$$I_{filtered}(x) = \frac{1}{W_{pixel}} \times \sum_{x_k \in N} I(x_i) \times (R_k \times ||I(x_i) - I(x)||) \times (S_k \times ||x_i - x||) \tag{1}$$

where,

$$W_{pixel} = \sum_{x_k \in N} (R_k \times ||I(x_i) - I(x)||) \times (S_k \times ||x_i - x||) \tag{2}$$

R_k, S_k = Range (or Color Intensity) kernel , Spatial Kernel respectively.
The resultant image obtained after updating the pixel intensities, we get the bilateral filter output image.

Figure 2: Bilateral Filter Output

□

Problem 3. Combine the bilateral filter with the edge detector to obtain a painting or cartoon like image.

Solution: The cartoon image is a combination of the bilateral filter output image and the edge detected image (canny edge detected image, in this case). A bit-wise AND operation represents the combination of these two images.

Figure 3: Cartoon Image

□

https://www.overleaf.com/read/bvhtnfnztcgm