Detecting contours of human organs in CT images using the Canny edge detector

Biljana Vitanova
BSIP 2024/25 , FRI, UL
bv7063@student.uni-lj.si

***Abstract*—This project implements the Canny edge detector and uses it for edge extraction from CT images of human body parts. It also implements the Otsu method for automatically estimating the threshold during the hysteresis step.**

I. Introduction

Medical imaging, particularly CT scans, plays a crucial role in diagnosing and understanding the internal structures of the human body. A key step in processing these images is contour detection. An efficient method for this is the Canny Edge detector [1], which uses multiple stages to identify edges. This project outlines the steps to fully implement the Canny edge detector and tests its effectiveness on 2D images.

II. Methodology

The Canny edge detector consists of three main stages:

1) Smoothing and Gradient Calculation
2) Applying Non-Maxima Suppression
3) Hysteresis Thresholding

A. Smoothing and Gradient Calculation

The first step involves smoothing the image to reduce noise caused by random variations in pixel intensity. A Gaussian kernel is applied to the image for this purpose. The Gaussian smoothing operation is given by:

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

After smoothing, the image is convolved with derivative filter such as Sobel, Roberts, or Prewitt to compute the gradients in the x- and y-directions (G_x and G_y).

The magnitude of the gradient, which determines the strength of potential edges, is calculated as:

$$|G| = \sqrt{G_x^2 + G_y^2} \tag{2}$$

The direction of the gradient, indicating the orientation of potential edges, is computed as:

$$\theta = \arctan\left(\frac{G_y}{G_x}\right) \tag{3}$$

This step prepares the image by identifying areas with significant intensity changes, which are likely to correspond to edges.

B. Applying Non-Maxima Suppression

In this step, only the strongest intensity pixels in a neighborhood are retained. For each pixel, the algorithm compares its magnitude with the magnitudes of the two neighboring pixels along the gradient direction. If the pixel does not have the highest magnitude, it is suppressed. This process thins the edges to a single-pixel thickness, ensuring a cleaner and more accurate edge map.

C. Hysteresis Thresholding

The final step is hysteresis thresholding, which determines which edges are kept in the final output. Two thresholds are used: a high threshold and a low threshold.

- Pixels with a gradient magnitude above the high threshold are considered strong edges and are kept.
- Pixels with magnitudes between the two thresholds are considered weak edges. These are retained only if they are connected to strong edges, ensuring continuity of edge structures.

This step helps in eliminating noise while preserving significant edges.

D. Otsu Method

The thresholding process in the hysteresis step requires careful selection of the threshold values. Setting thresholds too low may include noise, while thresholds that are too high may skip important edges. To automate the selection of these thresholds, the Otsu method can be applied.

The Otsu method determines the optimal threshold by analyzing the distribution of pixel intensity values in the image. It maximizes the between-class variance, which increases the separation between the foreground and background, which is equivalent to minimizing the within-class variance, ensuring that pixels within each class (foreground and background) are as similar as possible in intensity.

III. Experiments

For this project, I used the CTMRI [2] database , which contains CT images of human organs. There were three parameters that needed to be estimated:

- Kernel size for smoothing: To smooth the images, I used a Gaussian kernel with a size of 5x5 and $\sigma = 1$, which resulted in capturing more detailed and smaller edges.
- Derivative kernel: I tested the Prewitt and Sobel derivative kernels for edge detection. Both resulted in similar edge extraction, but in the end, I decided to use the Sobel kernel.
- Threshold ratio: I estimated the low threshold ratio as 0.1 times the high threshold, ensuring that smaller edges were detected.

IV. Results

Below is the application of the Canny edge detector for extracting edges from four different parts of the human body. The first column represents the original image, the second column shows the image after applying non-maximum suppression, and the third column shows the final image after performing hysteresis thresholding.

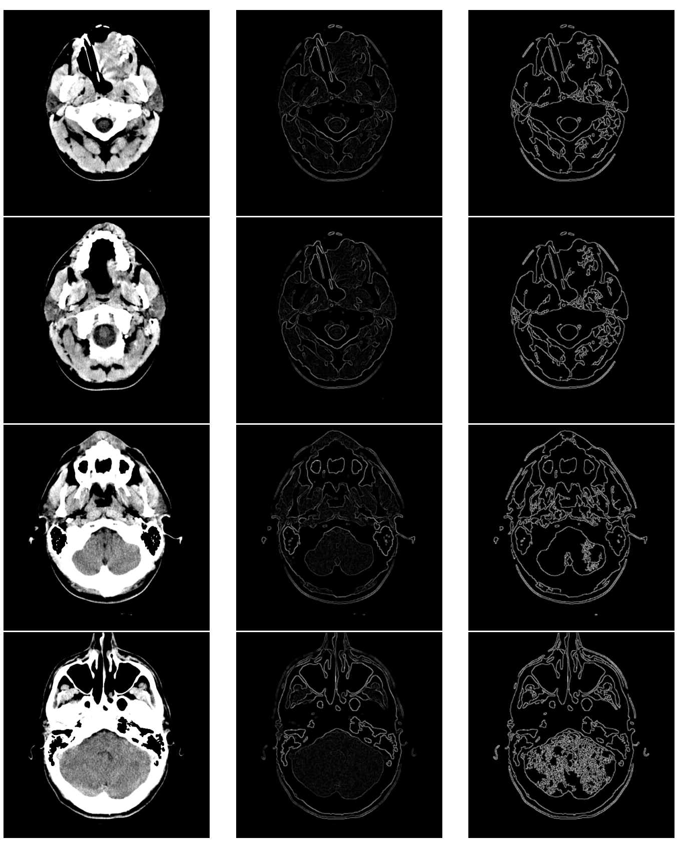

Fig. 1. Images of Patient 1

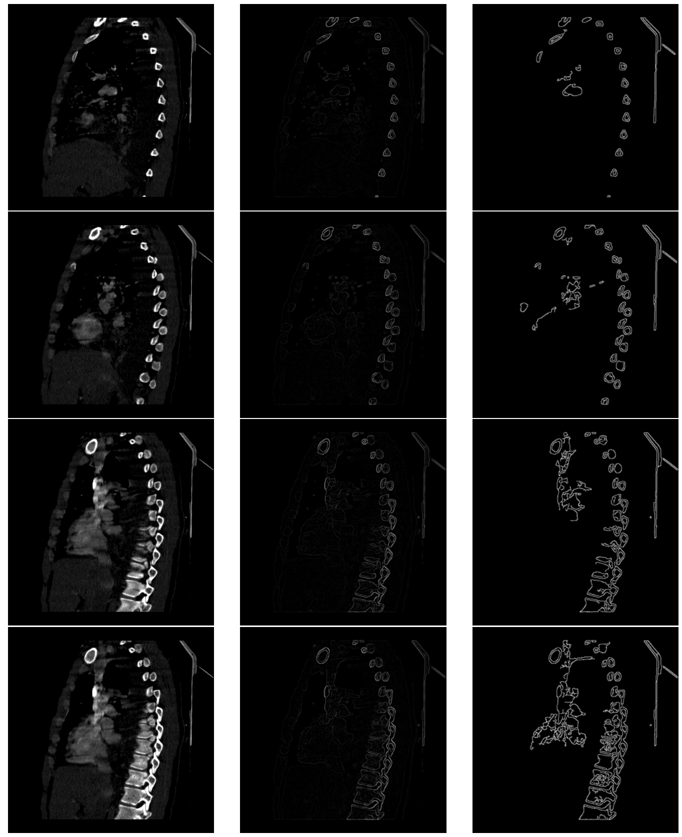

Fig. 2. Images of Patient 2

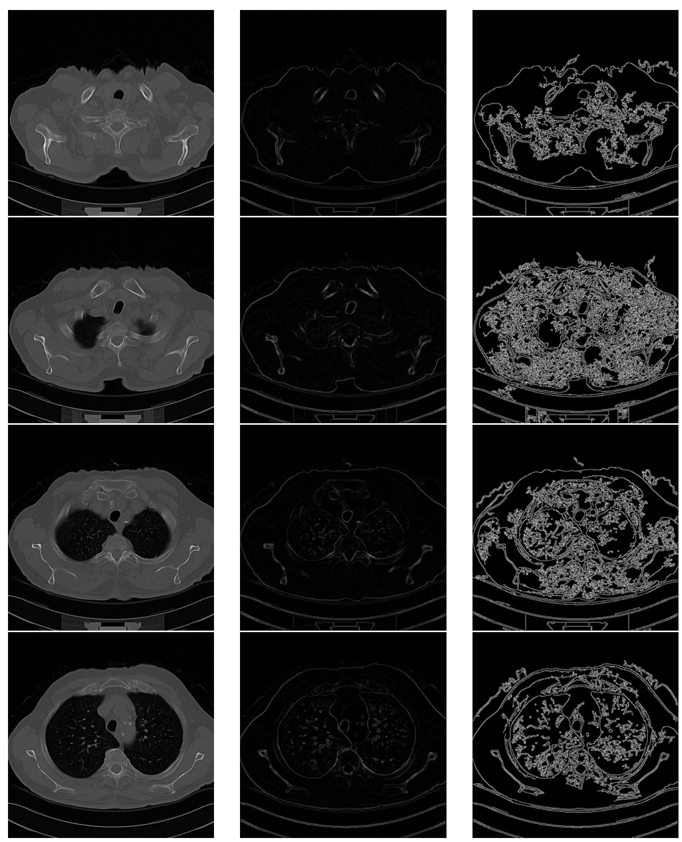

Fig. 3. Images of Patient 3

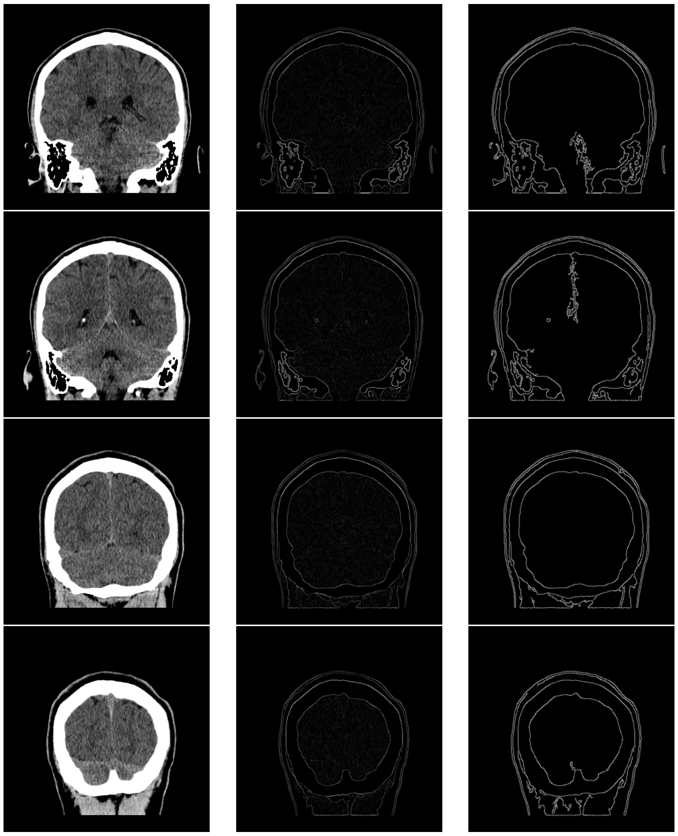

Fig. 4. Images of Patient 4

V. Conclusion

This project demonstrated the application of the Canny edge detector. It was successfully implemented, yielding satisfactory results for edge extraction. For future work, it is recommended to explore the application of the Canny edge detector on 3D images and further implementing the 24-connectivity step.

References

[1] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 6, pp. 679–698, 1986.

[2] A. Taddei, F. Jager, A. Smrdel, K. Bezlaj, and M. Zadnikar, "Ct-mri database." [Online]. Available: https://lbcsi.fri.uni-lj.si/OBSS/Data/CTMRI/