Analysis of computed tomography (CT) images

Blaž Erzar

Biomedical Signal and Image Processing 2023/24
Second section report

Abstract

This report describes the implementation of Canny's edge detector for 3D computed tomography images. The detector computes the gradient magnitude and angle images for the input image smoothed using a Gaussian. On top of these images, non-maxima suppression is used to thin the edges down to single pixel and hysteresis thresholding creates the final binary output image. We present some improvements to the basic algorithm, show the results of the algorithm on 14 CT slices from CTMRI databases together with a 3D point cloud visualisation and propose some potential future improvements.

1 Introduction

In the second section of the course Biomedical Signal and Image Processing, we have been analysing computed tomography images using edge detection. In this report, we present the implementation of edge detection using Canny's algorithm [1].

This algorithm incorporates gradient images of the smoothed image to detect the edges. Edges obtained from the gradient magnitude image are thinned using gradient orientations and a special thresholding procedure extracts strong pixels to obtain the final edges. To obtain the gradients in x and y directions, we can convolve the image with different kernels. We present Prewitt [2] and Sobel [3] operators for this. When using them, we need to first convolve the whole image with a Gaussian kernel to smooth it, meaning we need to do two convolution steps. As an improvement to this approach, we implement a Gaussian derivative kernel, that allows us to convolve the image once to obtain the derivative of the smoothed image.

In our work with 3D CT images – essentially, a series of 2D image slices – we find it crucial to connect edges between consecutive images. This is essential for a thorough analysis of the 3D dataset. To achieve this, we employ the 24-connectivity method. This technique allows us to link edges between adjacent images, enhancing our ability to examine structural details throughout the entire 3D CT scan.

The whole pipeline has been developed using the CTMRI database [4]. The final results for selected images of a specific patient's head CT are shown, to observe both the performance of edge detection as well as edge linking algorithms. We also include a 3D visualisation of the point cloud, obtained when using all CT slices as the input, as it can provide an easier understanding of the extracted contours.

2 Methods

2.1 Canny

Smoothing

The first step in Canny's algorithms is smoothing the input image to remove high-frequency noise. For this, we convolve the image with a **Gaussian kernel**:

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

For σ we take 0.5% of the smallest image dimension and $N = 2\sigma + 1$ for the size of the kernel, making sure its size is odd. The kernel is also normalised so its elements sum up to 1. Because this kernel is separable, we can achieve the same result using 1D convolution twice with 1D vertical and horizontal Gaussian kernels:

$$G(x) = e^{-\frac{x^2}{2\sigma^2}}.$$

Gradients

After the image is smoothed, we compute its gradients in x and y directions. We can do this by convolving the smoothed image using Prewitt,

$$P_x = \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix} \qquad P_y = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix},$$

or Sobel,

$$S_x = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \qquad S_y = \begin{bmatrix} 1 & 2 & -1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix},$$

operators.

The same result can be obtained by convolving the *original* image with the partial derivative of the 2D Gaussian kernel

$$\frac{\partial}{\partial x}G(x,y) = -xe^{-\frac{x^2+y^2}{2\sigma^2}}$$

and analogous for y, normalised so their absolute values of elements sum up to 1.

After computing gradients g_x and g_y, we can obtain **gradient magnitude**

$$m(x,y) = \sqrt{g_x(x,y)^2 + g_y(x,y)^2}$$

and **gradient angle**

$$a(x,y) = \arctan\frac{g_y(x,y)}{g_x(x,y)}$$

images.

Non-maxima suppression

The gradient magnitude image already contains edges, but they are too wide. To make them thin, we compare every gradient magnitude value to two values in the 8-neighbourhood. The two neighbours are selected in the direction of the gradient angle and the opposite direction. We set the gradient magnitude pixels to 0 if any of the two neighbours have a higher value.

Hysteresis thresholding

As output we want to have a binary image and some sort of thresholding needs to be used to achieve this. This process uses two thresholds, **low threshold** t_L and **high threshold** t_H. We used $t_L = 0.15$ and $t_H = 0.30$.

Every pixel with a gradient magnitude after non-maxima suppression higher than t_H is a strong edge point. Other pixels with values higher than t_L are weak edge points. The final edges consist of strong edge points and weak edge points that are connected to strong ones.

2.2 Edge linking

After obtaining edges for all slices of the 3D scan, we need to connect the edges of consecutive images. For this we use 24-connectivity. Let I_n and I_{n+1} be two consecutive Canny edge images for $n = 1, \ldots, N$, where N is the total number of input images.

We go over all edge pixels (x,y) in I_n. They are connected to all edges pixels (a,b) in 5×5 neighbourhood of I_{n+1} if 3×3 neighbourhood of I_{n+1} is empty. Connection is done using shortest path between (x,y) and (a,b), where we update x and y coordinates by 1 until they equal a and b.

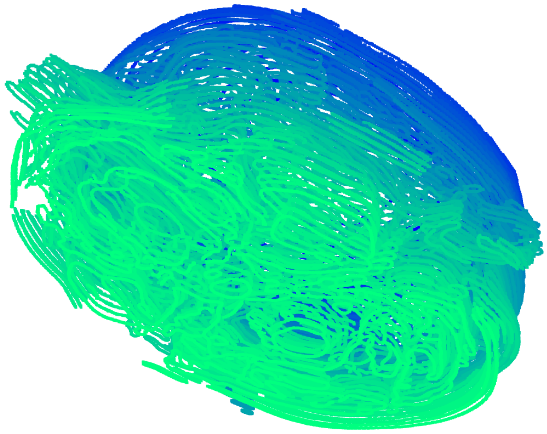

Figure 1: *3D visualisation of point cloud obtained from implemented algorithm. The point cloud contains the upper head CT viewed from the lower front.*

3 Results

The results of running the implemented algorithm on a small subset of 14 images from the CTMRI database can be seen in Figure 2. Edges added in the 3D linking stage are additionally coloured in yellow.

For easier interpretation, a 3D visualisation has also been made and is shown in Figure 1. In this case, we took CT images of the same patient, ran slices from the index 30 to 120 through the algorithm and visualised points after edge linking.

4 Discussion

Looking at the results we can conclude that the detector does detect edges quite consistently, but there is still room for improvement. Currently, some edges might be missed due to constant thresholds for hysteresis thresholding. The next sensible step would be to add a mechanism for setting these thresholds automatically per image.

At the same time, some performance improvements could be made. The detector takes around a second per image, but having a lot of CT slices can drastically increase the computation time. It would be beneficial to run detections in parallel since they can be computed independently.

References

[1] John Canny. "A Computational Approach to Edge Detection". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-8.6 (1986), pp. 679–698. DOI: 10.1109/TPAMI.1986.4767851.

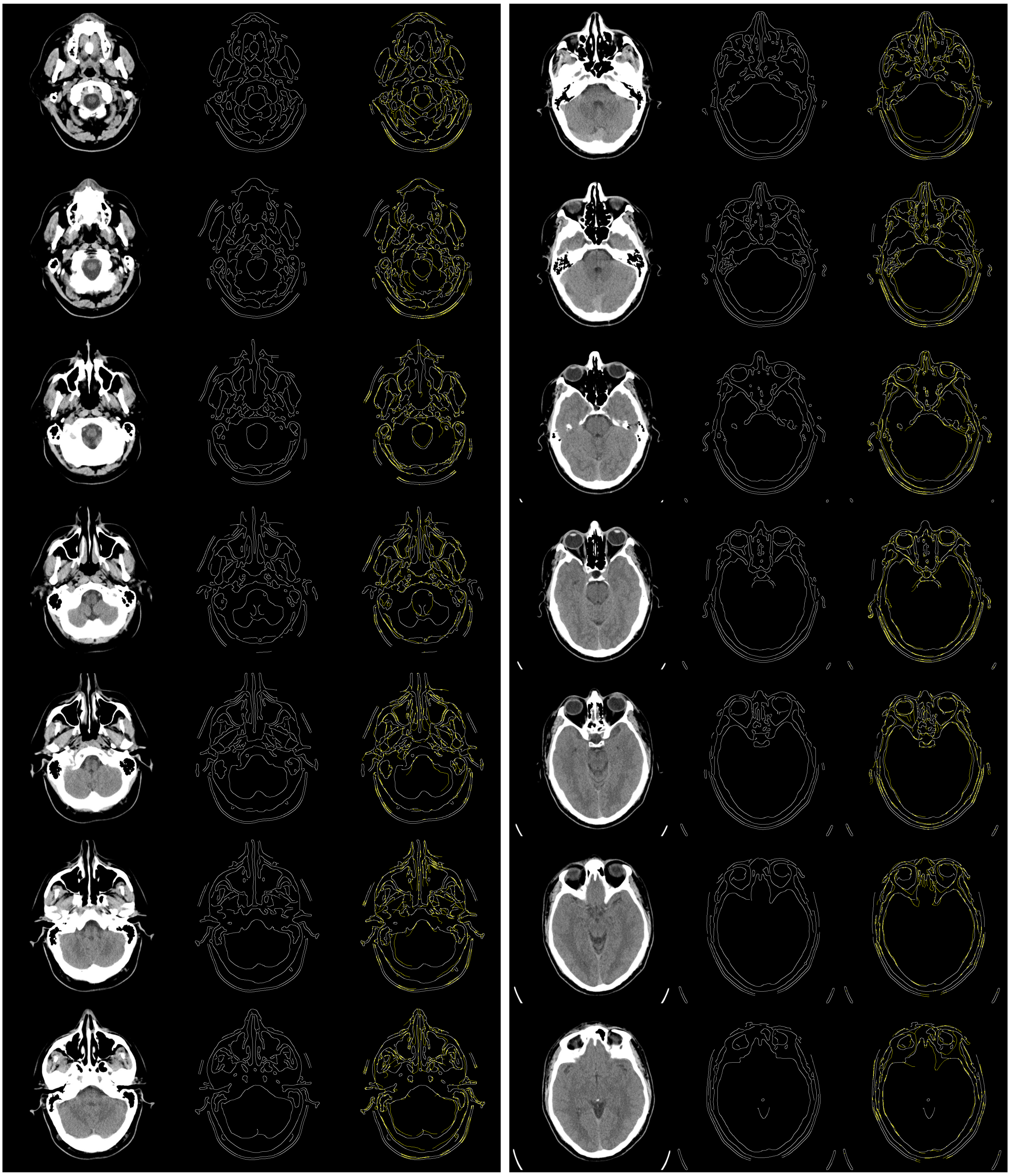

Figure 2: *Canny edge detection and edge linking results for 14 head CT images (the first seven in the first column and others on the right). Each row contains the original grey-scale image, edges from Canny's algorithm and 3D linked edges coloured in yellow.*

[2] Judith MS Prewitt et al. "Object enhancement and extraction". In: *Picture processing and Psychopictorics* 10.1 (1970), pp. 15–19.

[3] Irwin Sobel, Gary Feldman, et al. "A 3x3 isotropic gradient operator for image processing". In: *a talk at the Stanford Artificial Project in* (1968), pp. 271–272.

[4] Alessandro Taddei, Franc Jager, Ales Smrdel, Karmen Bezlaj, and Maja Zadnikar. *CT-MRI database*.