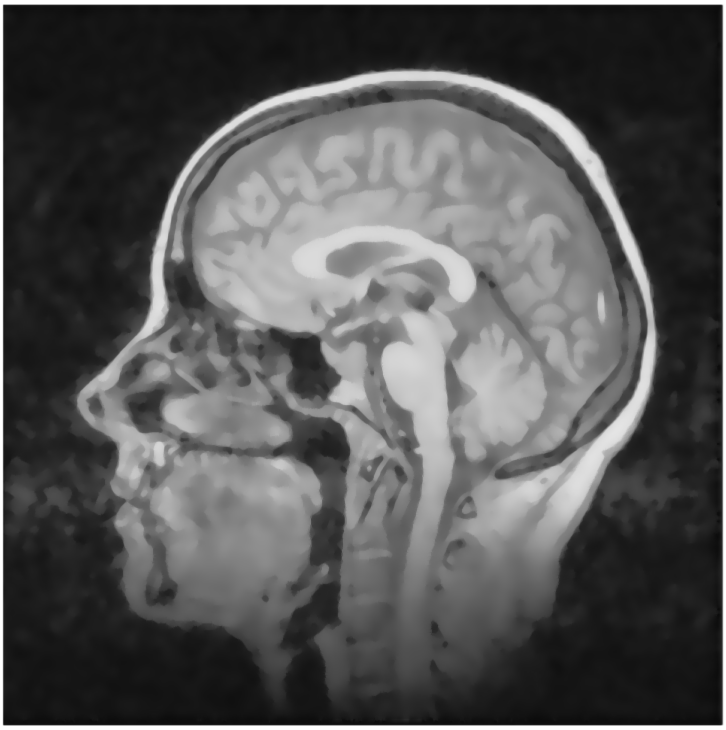
Computer Vision and Pattern Recognition Assignment - 2

Ufuk Dogan

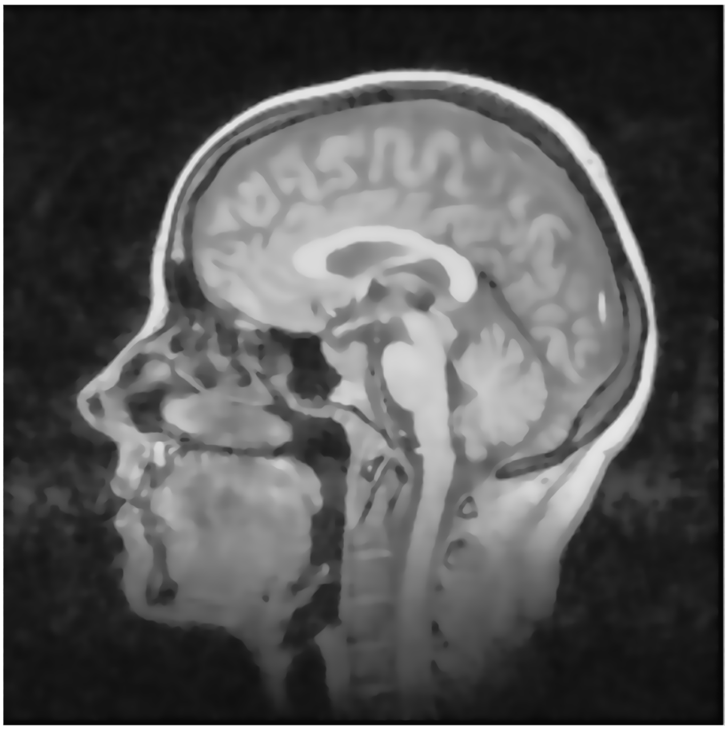
09 June 2019

1 Exercise - 1

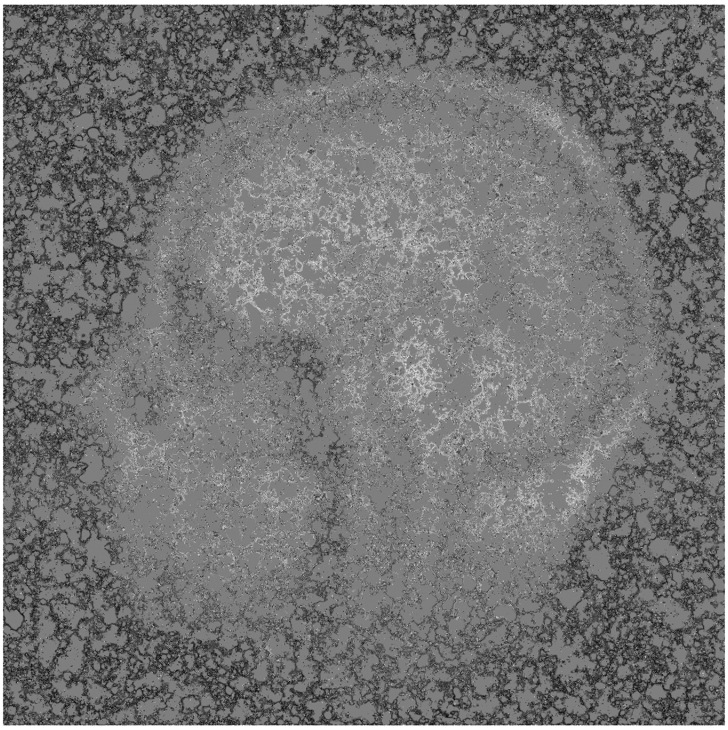
- Part A



- Part B



- Part C



- Part D

- What happens at point (c)? And why?

  The value of t affects the stability of this method. If the t value is between 0 and 0.25 the scheme converges otherwise, it does not converge. Von Neumann stability analysis method can be used in order to check the stability.

2 Exercise - 2

- Part A

$$\int_{\Omega\Omega_0} \left|f(x,y) - g(x,y)\right|^2 dx\,dy + \lambda \int_{\Omega} \|\nabla f(x,y)\|_2 \, dx\,dy$$

$$= \frac{dF}{df} - \left(\frac{d}{dx} \cdot \frac{dF}{df_x} + \frac{d}{dy} \cdot \frac{dF}{df_y} \right)$$

$$\frac{1}{2} \cdot \frac{2 f_x}{\sqrt{f_x^2 + f_y^2}} = \frac{f_x}{\sqrt{f_x^2 + f_y^2}}$$

$$\frac{1}{2} \cdot \frac{2 f_y}{\sqrt{f_x^2 + f_y^2}} = \frac{f_y}{\sqrt{f_x^2 + f_y^2}}$$

$$2\left(f(x,y) - g(x,y)\right) - \sqrt{f_x^2 + f_y^2} = 2\left(f(x,y) - g(x,y)\right) - \frac{f_x + f_y}{\sqrt{f_x^2 + f_y^2}}$$

$$= 2\left(f(x,y) - g(x,y)\right) - \mathrm{div}\left(\frac{\nabla f}{\|\nabla f\|_2} \right)$$

- Part B

$$\mathrm{div}\left(\frac{\nabla f(x,y)}{\sqrt{\varepsilon^2 + \|\nabla f\|_2^2}} \right) = \frac{d}{dx}\left(- \frac{f_x}{\sqrt{\varepsilon^2 + \|\nabla f\|_2^2}} \right) + \frac{d}{dy}\left(\frac{f_y}{\sqrt{\varepsilon^2 + \|\nabla f\|_2^2}} \right)$$

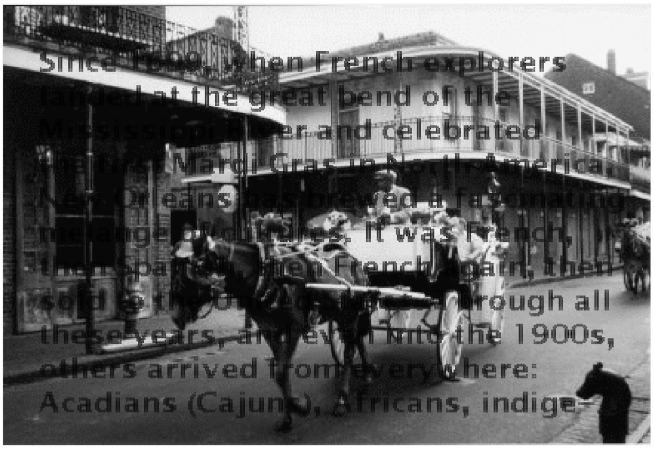
$$\frac{d}{dx}\left(\frac{f_x}{\sqrt{f_x^2 + f_y^2 + \varepsilon^2}} \right) = \frac{f_{xx} \cdot (f_x^2 + f_y^2 + \varepsilon^2)^{1/2} - f_x \left(\frac{d}{dx} (f_x^2 + f_y^2 + \varepsilon^2)^{1/2} \right)}{f_x^2 + f_y^2 + \varepsilon^2}$$

$$= \frac{f_{xx}(f_x^2 + f_y^2 + \varepsilon^2) - f_x (f_x \cdot f_{xx} + f_y \cdot f_{xy})}{(f_x^2 + f_y^2 + \varepsilon^2)^{3/2}} = \frac{f_{xx} \cdot f_y^2 + \varepsilon^2 f_{xx} - f_x f_y \cdot f_{xy}}{(f_x^2 + f_y^2 + \varepsilon^2)^{3/2}}$$

$$\frac{d}{dy}\left(\frac{f_y}{\sqrt{f_x^2 + f_y^2 + \varepsilon^2}} \right) = \frac{f_{yy} \cdot (f_x^2 + f_y^2 + \varepsilon^2)^{1/2} - f_y \cdot \frac{1}{2} \cdot (f_x^2 + f_y^2 + \varepsilon^2)^{-1/2} \cdot (2 \cdot f_x \cdot f_{xy} + 2 f_y \cdot f_{yy})}{f_x^2 + f_y^2 + \varepsilon^2}$$

$$= \left[f_{xx} \cdot f_y^2 - 2 f_{xy} f_x f_y + f_{yy} \cdot f_x^2 + \varepsilon^2 \cdot (f_{xx} + f_{yy}) \right] \Big/ (f_x^2 + f_y^2 + \varepsilon^2)^{3/2}$$

- Part C

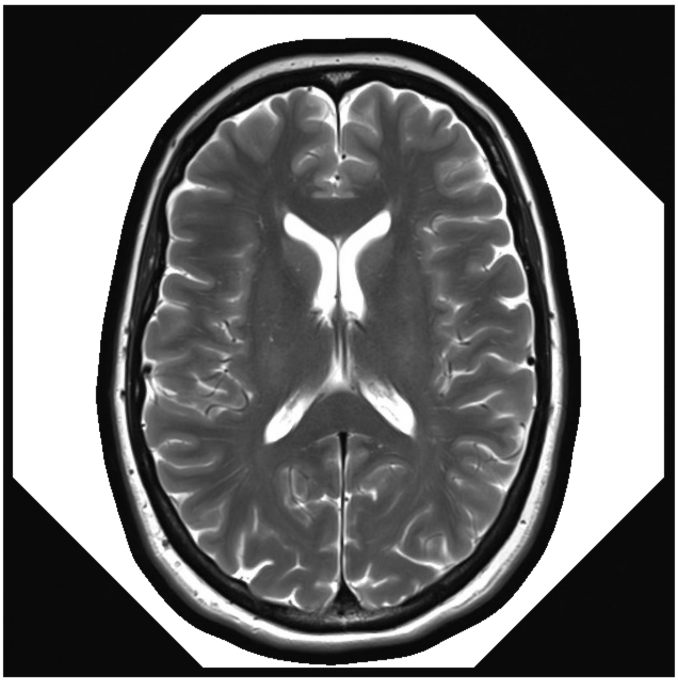


It is not stable since, with less time steps I have received images with texts. In order to improve the stability The Von Neumann stability can be applied.

- Part D

3 Exercise - 3

- Part A

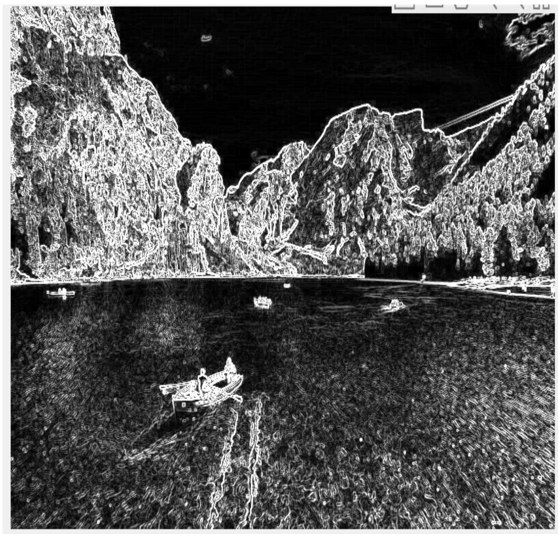


- Part B
  By the Fast Geodesic Active Contours paper, we can say that the efficiency of the AOS based implementations are high because the implementation works slowly. AOS based implementations should be upgraded to the narrow band. However, this method is tricky because the standard static run length implementation is not sufficient. Because of that the narrow band is rebuilt every iteration using the fast marching algorithm, which generates narrow band pixels in arbitrary order. Creating the run length encoding from such a stream of pixels can be done either off-line, by first constructing and then scanning a map of the whole image (which is clearly inefficient), or online using a dynamic data structure.

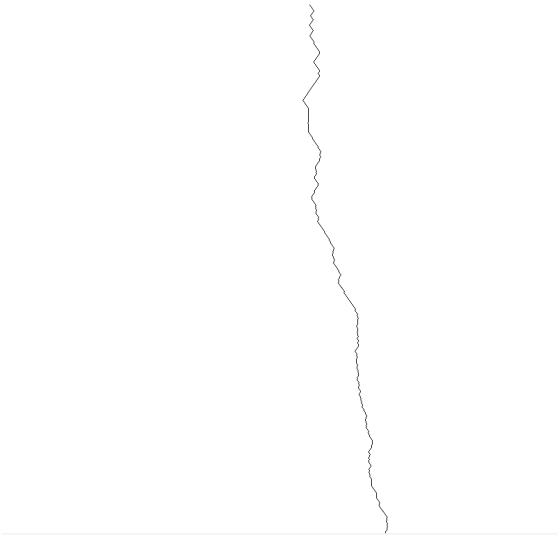
4 Bonus Exercise 1

- Part A

(a) Original Image



(b) Energy Map at First Iteration



(a) Seam at First Iteration

(b) After Vertical Seam Algorithm

- Part B

Seam carving is a way to resize an image for both reduction and extension using the content aware approach. A seam is a 8-connected path of pixels from top to bottom or left to right on the image. Seam can be found by many energy functions. In order to create the seam, in the code I have used the Sobel gradient operator which is basically gradient of the image. After creating the seam, the only thing should be done is removing the seam which is has the lowest energy in a for loop. At the end of this iteration and removing process, the output will be the new picture with desired size.

So, in my code i have followed the same steps I mentioned above, in part A you can see the pictures I captured.

5 Reference

- https://www.researchgate.net/publication/233910857_Fast_Geodesic_Active_Contours
- https://karthikkaranth.me/blog/implementing-seam-carving-with-python/
- http://graphics.cs.cmu.edu/courses/15-463/2007_fall/hw/proj2/imret.pdf