Quantification of fibre width in biological images

Lake Como School "Computational methods for inverse problems in imaging"

Mathilde Galinier

PhD student at the Department of Physics, Informatics and Mathematics
Università degli studi di Modena e Reggio Emilia

Wednesday, May 23rd 2018

INSTITUT NATIONAL DES SCIENCES APPLIQUÉES TOULOUSE

LISBP Ingénierie des Systèmes Biologiques et des Procédés

UNIMORE UNIVERSITÀ DEGLI STUDI DI MODENA E REGGIO EMILIA

Introduction

- **Aim :** Enabling a better understanding of microbial ecosystems in order to improve waste-water treatment methods.
- Project based on the analysis of sewage pictures.
- Algorithm associating Canny edge detector-related methods and statistical analysis.

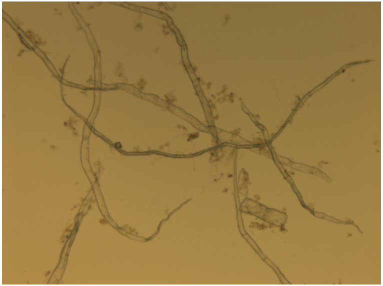


Figure: Raw image u, taken with a fluorescence microscope.

Step 1 : Convolution with Gaussian filters

- Image convolved by two Gaussian filters :

$$\begin{cases} u_1 = K_\sigma * u \\ u_2 = K_{\beta\sigma} * u, \quad \beta < 1 \end{cases}$$

with

$$K_\sigma(k,l) = \frac{1}{2\pi\sigma^2} exp\left(-\frac{k^2+l^2}{2\sigma^2}\right)$$

- For example, a $N \times M$ Gaussian filter K_σ applied to the image u provides :

$$u_{1\,ij} = \sum_{k=0}^{N-1} \sum_{l=0}^{M-1} K_\sigma(k,l) u(i-k, j-l)$$

which can be computed thanks to the Fourier transform :

$$u_1 = \mathcal{F}^{-1}\left(\mathcal{F}(K_\sigma)\ .\ \mathcal{F}(u)\right)$$

Step 2 : Computation of the gradient

- Detection of the edges of an image $f(x,y)$ based on the computation of the gradient.
- The vector :

$$\nabla f(x,y) = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}(x,y) = \begin{bmatrix} g_x \\ g_y \end{bmatrix}(x,y)$$

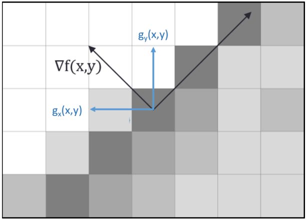
points in the direction of the greater rate of change of f at (x,y).

- Norm of $\nabla f(x,y)$:

$$\|\nabla f(x,y)\|_2^2 = g_x^2 + g_y^2$$

- Direction of $\nabla f(x,y)$:

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



Step 2 : Computation of the gradient

- In our case, the matrix of interest is written, for each pixel :

$$C = \frac{1}{2}\left(\nabla u_1 \nabla u_2^T + \nabla u_2 \nabla u_1^T\right) = \begin{pmatrix} G_{xx} & G_{xy} \\ G_{xy} & G_{yy} \end{pmatrix}$$

  where :

$$\begin{cases} G_{xx} = \partial_x u_1 . \partial_x u_2 \\ G_{yy} = \partial_y u_1 . \partial_y u_2 \\ G_{xx} = \frac{1}{2}\left(\partial_x u_1 . \partial_y u_2 + \partial_y u_1 . \partial_x u_2\right) \end{cases}$$

- **Topological gradient :** largest eigenvalue of C, noted λ_1.

Step 2 : Computation of the gradient

- In our case, the matrix of interest is written, for each pixel :

$$C = \frac{1}{2}\left(\nabla u_1 \nabla u_2^T + \nabla u_2 \nabla u_1^T\right) = \begin{pmatrix} G_{xx} & G_{xy} \\ G_{xy} & G_{yy} \end{pmatrix}$$

  where :

$$\begin{cases} G_{xx} = \partial_x u_1 . \partial_x u_2 \\ G_{yy} = \partial_y u_1 . \partial_y u_2 \\ G_{xx} = \frac{1}{2}\left(\partial_x u_1 . \partial_y u_2 + \partial_y u_1 . \partial_x u_2\right) \end{cases}$$

- **Topological gradient :** largest eigenvalue of C, noted λ_1.
- *Remark :* In the case $u_1 = u_2$:

$$C = \begin{pmatrix} \partial_x u_1^2 & \partial_x u_1 \partial_y u_1 \\ \partial_x u_1 \partial_y u_1 & \partial_y u_1^2 \end{pmatrix} \qquad \text{and} \qquad \lambda_1 = \partial_x u_1^2 + \partial_y u_1^2 = \|\nabla u_1\|_2^2$$

Step 2 : Computation of the gradient

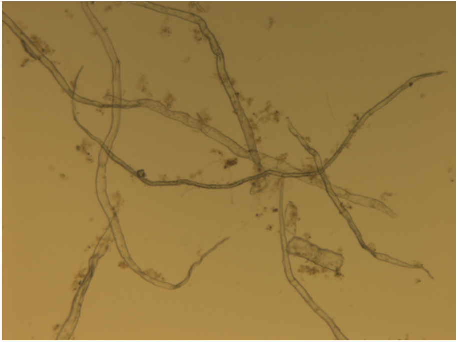


Figure: Raw image u.

Step 2 : Computation of the gradient

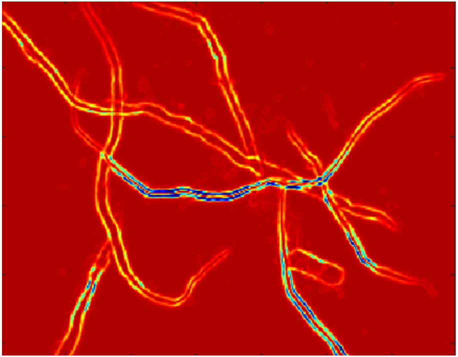
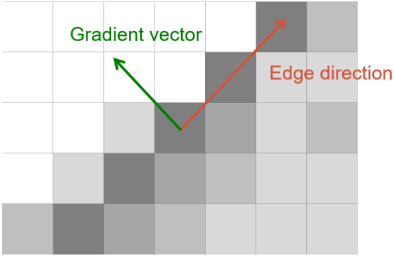
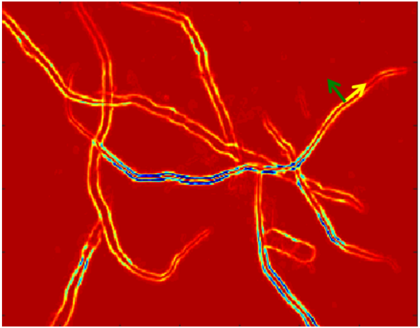


Figure: Topological gradient of u.

Step 3 : Non local maxima suppression

- 1. Find the 2 closest pixels along the edge normal.
- 2. Retain the pixel with maximum magnitude value.

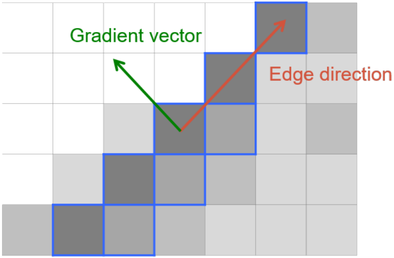


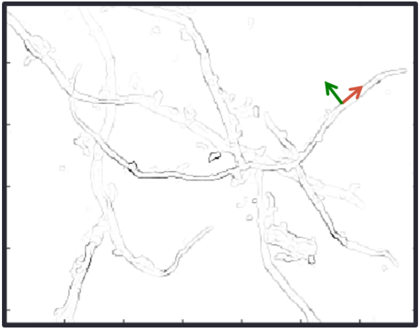


Topological gradient of u.

Step 3 : Non local maxima suppression

- 1. Find the 2 closest pixels along the edge normal.
- 2. Retain the pixel with maximum magnitude value.

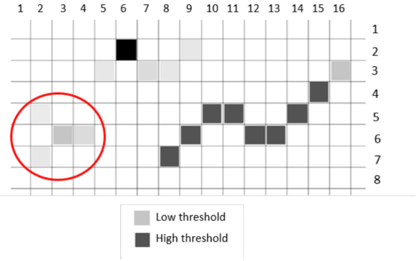


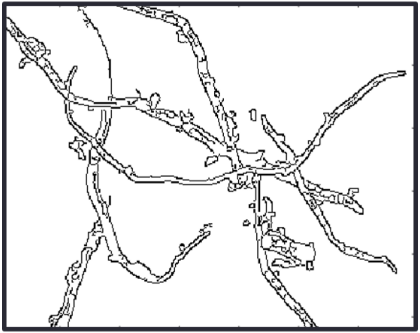


Non local maxima suppression

Step 4 : Hysteresis thresholding

- Keep pixels with a low intensity only if they are connected to a 'strong' pixel.

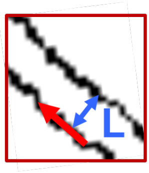


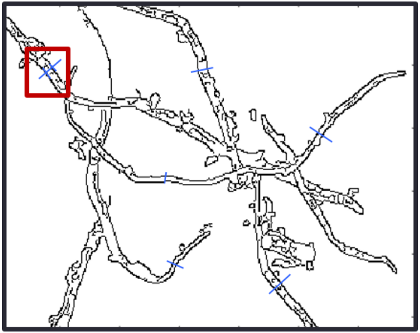


Binary image after hysteresis thresholding

Step 5 : Computation of fiber width

- For each non-zero pixel, the algorithm searches for another edge in the direction of the edge normal.



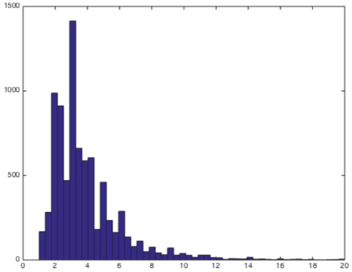


Binary image after hysteresis thresholding

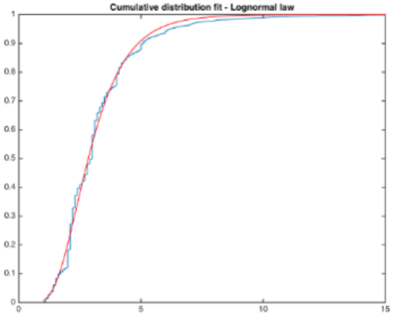
Statistical analysis

- Gauss-Newton algorithm for the fitting of the cumulative distribution functions :

$$\min_p \|G(p)\|^2_{L^2} = \|F_{p_1,\cdots,p_K} - F_{data}\|^2_{L^2}$$



Histogram of fiber widths. Abscissa : fiber width (pixels) ; Ordinate : number of fibers by category.



In red : Data cumulative distribution function. In blue : Fitting with lognormal cumulative distribution function.

Comparison of the cumulative distribution functions over several days

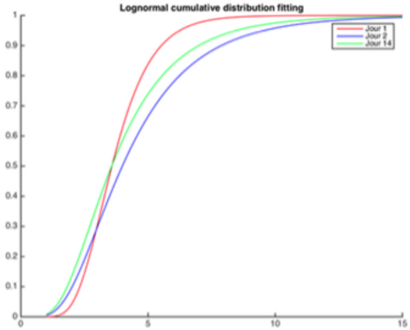


Figure: Lognormal cumulative distribution functions of a sample, for days 1,2 and 14.

References

Samuel Amstutz and Jérôme Fehrenbach.
Edge detection using topological gradients: A scale-space approach.
Springer Science+Business Media, Jan 2015.

John Canny.
A computational approach to edge detection.
IEEE Transactions on pattern analysis and machine intelligence, PAMI-8(6), Nov 1986.

Nick Efford.
Digital Image Processing: A Practical Introduction Using Java.
Pearson Education, 2000.