

Using Clustering and Controlled Regression to Analyze the Orientation of Rod-Like Objects

*Original*

Using Clustering and Controlled Regression to Analyze the Orientation of Rod-Like Objects / Sparavigna, Amelia Carolina; Marazzato, Roberto. - In: INTERNATIONAL JOURNAL OF SCIENCES. - ISSN 2305-3925. - 8:05(2019), pp. 16-18. [10.18483/ijSci.2055]

*Availability:*

This version is available at: 11583/2732966 since: 2019-05-11T20:55:44Z

*Publisher:*

Alkhaer Publications

*Published*

DOI:10.18483/ijSci.2055

*Terms of use:*

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

*Publisher copyright*

(Article begins on next page)

# Using Clustering and Controlled Regression to Analyze the Orientation of Rod-Like Objects

R. Marazzato<sup>1,2</sup>, A. C. Sparavigna<sup>3</sup>

<sup>1</sup>Dipartimento di Scienze Matematiche "G. L. Lagrange", Politecnico di Torino, Torino, Italy

<sup>2</sup>Dipartimento di Management, Università degli Studi di Torino, Torino, Italy

<sup>3</sup>Dipartimento Scienza Applicata e Tecnologia, Politecnico di Torino, Torino, Italy

**Abstract:** This work proposes the use of the R language for the analysis of the orientation of rod-like objects. First, the objects are separated from the background by means of a segmentation based on a thresholding. DBSCAN, a density based clustering algorithm, is used on the thresholded image. After the clustering, a linear regression is applied to determine the orientation of rod-like objects. The orientation is displayed by means of a radar diagram.

**Keywords:** Image Processing, Image Segmentation, Orientation of Textures

## Introduction

A part of the image processing is devoted to the analysis of oriented textures. The analysis consists, as told in [1], of the extraction of an orientation field. In [1], for instance, an algorithm was given to represent a flow-like texture, using an oriented filter - the gradient of Gaussian - and performing manipulations on the resulting gradient vector field. A sparse visualization, as proposed in [1], can be replaced by means of a visualization of dense vector fields, using algorithms such as Thick Oriented Stream-Line (TOSL) algorithm [2-4], or the Line Integral Convolution (LIC) and fastLIC algorithms [5-7].

Besides the use of vector fields, the orientation of objects in digital images can be determined using the tensors of inertia that reflect the mean shape of the grains [8-11]. An example of the use of the tensor of inertia to determine the orientation of rod-like objects is given in [12]. In the given reference, the orientation is determined for each object detected after a segmentation of the image [13], based on the thresholding of the grey tones of the pixels. After measuring the directions of the axes of the objects in

the image frame, the distribution of the orientation can be given.

Here, our aim is that of proposing a method to determine the distribution of the orientation, which is based on the use of a linear regression implemented by R language. R is a programming language and free software environment for statistical computing and graphics supported by the R Foundation for Statistical Computing. It is widely used among statisticians. In this framework, we will consider sets of rod-like objects displayed in an image frame. It can be applied to images like the one shown in Fig. 1, where a certain number of linear or quasi-linear shapes represent the content to be extracted and analyzed. The input is the grey-scale version of the image itself, while the output of the process is a radar chart showing the counts of objects for each orientation  $\theta_i \in (-\pi/2, \pi/2)$  taken by at least one of them.

## Method

A greyscale image of width  $w$  and height  $h$  can be represented by a set of triples

$$\hat{I} = \{(x, y, v_{xy}) : x \in I_w \wedge y \in I_h \wedge v_{xy} \in [0, 255]\}.$$

As a first step to detect objects, under the hypothesis that a set of rod-like ones exists, we extract the set of points of the image satisfying a threshold constraint  $B: = v_{xy} < v_T$  or  $B: = v_{xy} > v_T$  for a given threshold  $v_T$ :

$$\hat{I}_T = \{(x, y) : (x, y, v_{xy}) \in \hat{I} \wedge B\} \subseteq I_w \times I_h.$$

This set is suitable for a clustering partition; we use a DBSCAN algorithm [14], which involves the use of the two parameters  $\epsilon$  and  $n$ , so to generate a set of subsets of  $\hat{I}_T$  connected by point density:

$$\hat{R}(\hat{I}_T, \epsilon, n) = DBSCAN_{\epsilon, n}(\hat{I}_T) = \{\hat{R}_k \subseteq \hat{I}_T : k \neq k' \Rightarrow \hat{R}_k \cap \hat{R}_{k'} = \emptyset \wedge \bigcup_k \hat{R}_k \subseteq \hat{I}_T\}_{\epsilon, n}$$

This article is published under the terms of the Creative Commons Attribution License 4.0

Author(s) retain the copyright of this article. Publication rights with Alkhaer Publications.

Published at: <http://www.ijsciences.com/pub/issue/2019-05/>

DOI: 10.18483/ijSci.2055; Online ISSN: 2305-3925; Print ISSN: 2410-4477

Each  $\widehat{R}_k$  is a rod-like graphical object. For each of them, we can use a linear model, choosing between the two options to compute the fitted values:

$$\hat{y} = \beta_{1k}x + \beta_{0k}$$

and

$$\hat{x} = \bar{\beta}_{1k}y + \bar{\beta}_{0k}.$$

We will use the first one when the rod-like object is roughly “more horizontal than vertical”, the second one in the other cases, implementing a controlled version of linear regression. This allows us to compute the orientation of the  $k$ -th rod:

$$\widehat{\theta}_k = \begin{cases} \arctan(\beta_{1k}) & \leftarrow V(x) > V(y) \\ \arctan(\bar{\beta}_{1k}^{-1}) & \leftarrow V(x) < V(y). \end{cases}$$

A basic subdivision / grouping of the  $\widehat{\theta}_k$  values into the classes  $\Theta_i$  yields the couples  $(\theta_i, n_i)$ , where  $n_i = \#\{\widehat{\theta}_i: \widehat{\theta}_i \in \Theta_i\}$

and  $\theta_i$  is the representative (mean) value of the class. Such couples can be graphically depicted by means of a radar chart, in which only the right side is used. We implemented the whole algorithm in R, using the packages jpeg, reshape2, fpc, fmsb. Here in the following an example of processing, with the outline of each phase.

#### Reading the image.

First, we need to read the image into the appropriate R data structure. All plots which are here shown are given just to visually describe the steps of the procedure. Let us consider, for instance, the image given in the Figure 1. Objects in the image frame are rod-like.



Fig.1: Original image.

#### Getting ready for segmentation

The image is converted (molten) to a dataframe. A threshold is used to select the points to analyze. Rows are assigned to  $x$ , while columns to  $y$ , which needs to be reversed so that the plot looks exactly like the original image. A boolean parameter is used in order to consider black or white points for processing. The result is shown in the Figure 2.

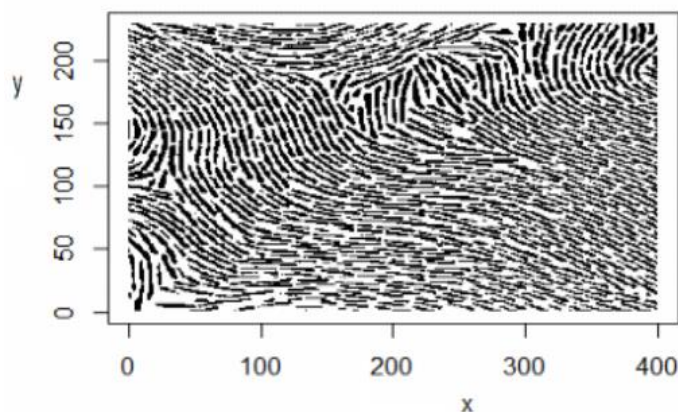


Fig.2: The image is obtained after the thresholding of the Figure 1.

#### Using DBSCAN to get the superpixels

DBSCAN is a density based clustering algorithm. It can be used to segment the given image. The maximum reachability distance and the minimum number of surrounding points for reachability are two parameters which must be set. No optimization of such parameters has been implemented up to now. The result of the segmentation of the Fig.2 is given in the Fig.3, where we can see the superpixels represented with different colors. Of course, it is possible to choose the colors according to some specific features of the superpixels, for instance, according to their length.

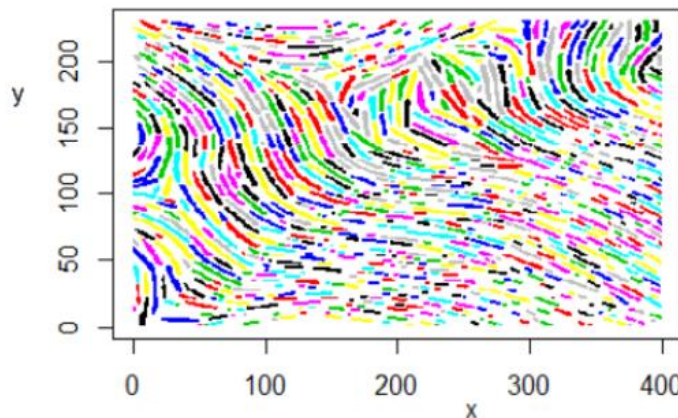


Fig.3: Colored superpixels of the rod-like objects in the Figure 2.

#### Linearizing the superpixels

A regular linear model algorithm, with the optimization of the relationship  $(y \sim x \text{ or } x \sim y)$  is used to perform the linearization. The graphical part of this step is not strictly necessary, but it helps to visualize the slopes. Their values and counts are grouped into a histogram, which is not displayed. The linearized superpixels are shown in the Figure 4.

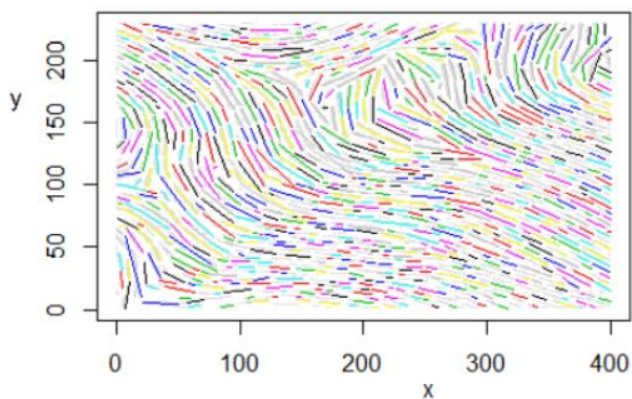


Fig. 4: Linearized superpixels.

### Plotting the results

The counts from the histogram are converted to a radar (spider) chart, in order to associate them to the corresponding orientation ( $-\pi/2$  to  $\pi/2$ ). The chart is given in the Figure 5.

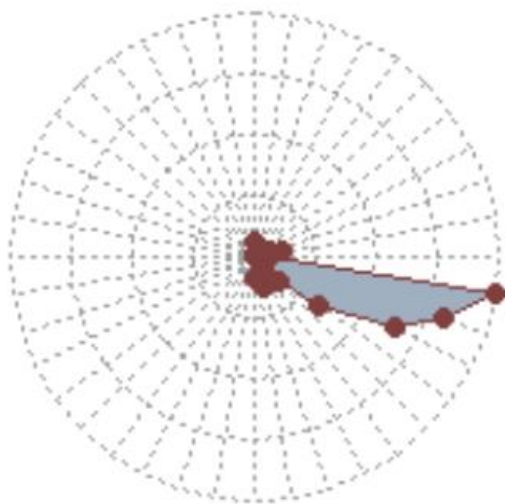


Fig. 5: The counts from histograms are converted to a radar (spider) chart.

### Discussion

The method proposed above, based on the use of R-language and related libraries can have several applications. For instance, as in the case of the image shown in the Figure 1, we could analyze the distribution of rod-like bacteria or cells. Like any automated image analysis software, such as that discussed in [15], we can perform an automated enumeration of bacterial cells and provide quantitative estimates of bacterial length. However, the use of the method is not limited to field of biological imaging. We can apply it in all the case where a rose of directions can be used to give information on certain features in the image. For instance we can determine the distribution of fibres, from the microscopic to the macroscale, without using Fourier analysis [16,17], or we can study the orientation of cracks, from the microcracks of the craquelure to the huge crevasses of glaciers [18], and so on. A future work will be devoted to illustrate the use of the method in the case of satellite imagery.

### References

1. Rao, A. R., & Schunck, B. G. (1991). Computing oriented texture fields. *CVGIP: Graphical Models and Image Processing*, 53(2), 157-185. DOI: 10.1016/1049-9652(91)90059-s
2. Sanna, A., Montrucchio, B., Montuschi, P., & Sparavigna, A. (2001). Visualizing vector fields: the thick oriented streamline algorithm (TOSL). *Computers & Graphics*, 25(5), 847-855. DOI: 10.1016/s0097-8493(01)00126-1
3. Sparavigna, A., Sanna, A., Montrucchio, B., & Strigazzi, A. (1999). Streamline image analysis: a new tool for investigating defects in nematic liquid crystals. *Liquid crystals*, 26(10), 1467-1478. DOI: 10.1080/026782999203797
4. Sparavigna, A., & Montrucchio, B. (2006). Vector field visualization with streamlines. *arXiv preprint cs/0610088*.
5. Cabral, B., & Leedom, L. C. (1993). Imaging vector fields using line integral convolution. *Proceedings of SIGGRAPH '93. Computer Graphics 1993*. p.263-72. DOI: 10.1145/166117.166151
6. Stalling, D., & Hege, H. C. (1995). Fast and resolution independent line integral convolution. *Proceedings of SIGGRAPH '95, Computer Graphics, 1995*. p. 249-56. DOI: 10.1145/218380.218448
7. Forssell, L. K., & Cohen, S. D. (1995). Using line integral convolution for flow visualization: Curvilinear grids, variable-speed animation, and unsteady flows. *IEEE Transactions on Visualization and Computer Graphics*, 1(2), 133-141. DOI: 10.1109/2945.468406
8. Mlekusch, B. (1999). Fibre orientation in short-fibre-reinforced thermoplastics II. Quantitative measurements by image analysis. *Composites Science and Technology*, 59(4), 547-560. DOI: 10.1016/s0266-3538(98)00101-8
9. Paquet, E., Rioux, M., Murching, A., Naveen, T., & Tabatabai, A. (2000). Description of shape information for 2-D and 3-D objects. *Signal processing: Image communication*, 16(1-2), 103-122. DOI: 10.1016/s0923-5965(00)00020-5
10. Cyganek, B. (2013). *Object detection and recognition in digital images: theory and practice*. John Wiley & Sons.
11. Csetverikov, D. (1999). *Basic algorithms for digital image analysis: a course*. Institute of Informatics, Budapest, Hungary.
12. Sparavigna, A. C. A method based on image segmentation for the analysis of the orientation of rod-like objects. *Zenodo*. DOI: 10.5281/zenodo.2567394
13. Sparavigna, A. C. (2017). Image Segmentation Applied to the Study of Micrographs of Cellular Solids. *International Journal of Sciences*, 6(02), 68-76. DOI: 10.18483/ijSci.1201
14. Martin Ester, Hans-Peter Kriegel, Joerg Sander, Xiaowei Xu (1996). A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. *Institute for Computer Science, University of Munich. Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining (KDD-96)*.
15. Selinummi, J., Seppälä, J., Yli-Harja, O., & Puhakka, J. A. (2005). Software for quantification of labeled bacteria from digital microscope images by automated image analysis. *Biotechniques*, 39(6), 859-863. DOI: 10.2144/000112018
16. Tunák, M., & Linka, A. (2007). Analysis of planar anisotropy of fibre systems by using 2D Fourier transform. *Fibres & Textiles in Eastern Europe*, 15(5-6), 64-65.
17. Redon, C., Chermant, L., Chermant, J. L., & Coster, M. (1999). Automatic image analysis and morphology of fibre reinforced concrete. *Cement and Concrete Composites*, 21(5-6), 403-412. DOI: 10.1016/s0958-9465(99)00025-6
18. Smith, B. E., Haran, T. M., Morin, P. J., Porter, C. C., & Scambos, T. A. (2013, December). Finding clouds, dunes, crevasses and subglacial valleys with surface-texture maps of Antarctica. *In AGU Fall Meeting Abstracts*.