Stochastic permutation ordering watershed

Olivier Lézoray

Stochastic permutation ordering watershed

Olivier Lézoray
Normandie Univ, UNICAEN, ENSICAEN, CNRS, GREYC
14000 Caen, France
olivier.lezoray@unicaen.fr

***Abstract*—The stochastic watershed is a morphological approach to segmentation that repeats the application of a seeded watershed from series of uniform random markers. The obtained watershed boundaries are combined to construct a probability density function. We propose an alternative approach called stochastic permutation ordering watershed. Our approach relies on the construction of several permutation orderings of the pixels of an image, starting from a random pixel and using a dedicated Hamiltonian path construction on a graph. From the permutation orderings, several seeded watersheds are applied and averaged. In contrast to the stochastic watershed, our approach enables to take into account any features associated to pixels, such as patches, of prime important to segment textured images. Experimental results show the benefit of the approach.**

***Index Terms*—Segmentation, stochastic watershed, Hamiltonian path.**

I. Introduction

For image segmentation, the watershed [1] is among the most popular methods that have been proposed so far. This approach belongs to the research field of Mathematical Morphology (MM) [2], a nonlinear approach to image processing. Starting from region seeds, the watershed performs a flooding on a gradient image to obtain a segmentation. Efficient algorithms [3] have been proposed to perform this flooding process. In an unsupervised segmentation process, region seeds are usually the minima of the gradient. However this is very sensitive to both noise and over-segmentation [1]. Typical approaches to cope with these problems do either remove local minima in a pre processing step or merge the final regions in a post processing step. Another solution is to consider, instead of the gradient image, an image that reduces the minima inside the objects and enhances the boundaries. With this objective, Angulo and Jeulin have proposed [4] the stochastic watershed. The principle is twofold. First, one repeats the application of a seeded watershed from series of uniform random markers using Monte Carlo simulations. Second, the obtained boundaries of the watershed segmentation are combined to construct a probability density function (pdf) for the boundaries in the image. The pdf computed by the stochastic watershed converges to the probability of seeds falling on both sides of each boundary and can be computed using graph algorithms [5]. Several works have been proposed to extend the stochastic watershed. In [6] a variant based on regional regularization of the image was proposed. In [7] a multi-scale approach including bagging was proposed. In [8], the probabilities of contours are computed from a gaussian model of image regions.

In this paper we propose an alternative approach for the computation of the probability density function for the boundaries in the image. In the original formulation of [4], each simulation is performed on a gradient image [9] that usually only considers the color differences between pixels. Therefore, the classical stochastic watershed cannot use more advanced pixel features such as patches to construct the pdf for the boundaries in the image. However, texture features such as patches have shown to be very efficient in many computer vision tasks [10]. We introduce an approach to build a pdf for the boundaries in the image that can consider any feature describing a pixel. Our approach relies on the construction of several permutation orderings of the pixels of an image, starting from a random pixel and using a dedicated Hamiltonian path construction on a graph. In section II we present how to construct such a stochastic Hamiltonian path and construct a pdf. In Section III, we present some results and comparisons. Last section concludes.

II. Stochastic permutation ordering watershed

A. Notations

We consider that the domain Ω of the image is a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where vertices $\mathcal{V} = \{v_1, \ldots, v_m\}$ correspond to pixels and edges $e_{ij} = (v_i, v_j)$ connect vertices. With this definition, images are represented as graphs signals [11] that associate vectors to vertices and are defined by the mapping $f : \mathcal{G} \rightarrow \mathcal{F} \subset \mathbb{R}^n$ where $\mathcal{F}$ is a non-empty set of vectors (we will consider only RGB color vectors, i.e., $n = 3$). To each vertex $v_i \in \mathcal{G}$ is associated a vector $\mathbf{v}_i = f(v_i)$. The set $\mathcal{T} = \{\mathbf{v}_1, \cdots, \mathbf{v}_m\}$ denotes the vectors associated to all the vertices of the graph. Consequently, one has $|\mathcal{V}| = m$. We will use the notation $\mathcal{T}[i] = \mathbf{v}_i$ to denote the i-th element of the set $\mathcal{T}$.

The graph $\mathcal{G}$ used to represent an image can be a classical grid-graph where pixels are connected with 8-adjacency, but it can be interesting to go beyond this classical grid-graph construction and to consider graphs of the higher connectivities. We will consider several possible graphs to represent an image. The graph can use only spatial or spectral information, or both.

- B-adjacency graph (denoted $\mathcal{G}_B$): this graph construction connects each vertex v_i to all the vertices contained in a square box of size $(2B + 1) \times (2B + 1)$ around v_i. 8-adjacency grid graphs do correspond to graphs $\mathcal{G}_1$.
- Global K-Nearest Neighbor graph (denoted $\mathcal{G}_*^K$): this graph construction connects each vertex v_i to its K

nearest neighbors (in terms of spectral distance) within the set of all vertices.

The first graphs $\mathcal{G}_B$ consider only the spatial proximity information to connect vertices whereas graphs $\mathcal{G}_*^K$ consider only the spectral proximity information and are fully nonlocal graphs [12].

B. Complete lattices

The classical construction of algebraic morphological operators relies on complete lattices [13] that impose the need of an ordering relationship between the elements to be processed. Within this theoretical background, morphological operations can be described as mappings between complete lattices. A complete lattice $(\mathcal{T}, \leq)$ is a non-empty set equipped with a total ordering relation, such that every non-empty subset $\mathcal{P}$ of $\mathcal{T}$ has a lower bound $\wedge\mathcal{P}$ and an upper bound $\vee\mathcal{P}$. If MM is well defined for gray scale functions, there exists no general admitted extension that permits to perform morphological operations on vectors since there is no natural ordering of vectors [14]. If many approaches have been proposed for color images [15], very few deal with patches [16], [17]. In this paper we will define an ordering relation between the vectors of a set $\mathcal{T}$ with the use of h-orderings [18]. This corresponds to defining a surjective transform h from $\mathcal{T}$ to $\mathcal{L}$ where $\mathcal{L}$ is a complete lattice equipped with the conditional total ordering [18]. We refer to $\leq_h$ as the h-ordering given by:

$$h : \mathcal{T} \to \mathcal{L} \text{ and } \mathbf{v} \to h(\mathbf{v}), \forall(\mathbf{v}_i, \mathbf{v}_j) \in \mathcal{T} \times \mathcal{T}$$
$$\mathbf{v}_i \leq_h \mathbf{v}_j \Leftrightarrow h(\mathbf{v}_i) \leq h(\mathbf{v}_j) \ . \tag{1}$$

Then, $\mathcal{T}$ is no longer required to be a complete lattice, since the ordering of $\mathcal{T}$ can be induced upon $\mathcal{L}$ by means of h [15]. When h is bijective, this corresponds to defining a space filling curve [19] and we will use this property in the sequel.

C. Stochastic Hamiltonian path

To construct a complete lattice with an image-adaptive h-ordering, we propose to construct a space-filling curve [19] on a graph $\mathcal{G}$ associated to the image. This corresponds to the construction of an Hamiltonian path on the image: a path that goes through all the vertices of the graph and traverses each vertex only once. We consider a general approach and build an order for all the set $\mathcal{T}$ in the form of a global Hamiltonian path.

Given the set $\mathcal{T}$, the construction of an Hamiltonian path amounts to define a sorted permutation $\mathcal{P} = \mathbf{P}\mathcal{T}$ of the vectors of $\mathcal{T}$ with $\mathbf{P}$ a permutation matrix of size $m \times m$. Let σ be a permutation of the index set $\mathcal{J} = \{1, \cdots, m\}$. If $\sigma(i) = j$, then $\mathbf{P}_{ij} = 1$ and 0 otherwise. The induced permutation is $\mathcal{P} = \mathbf{P}\mathcal{T} = \{\mathbf{v}_{\sigma^{-1}(1)}, \cdots, \mathbf{v}_{\sigma^{-1}(m)}\}$ where $(\sigma^{-1} \circ \sigma)$ is the identity. Any permutation is not of interest, and spatial and spectral constraints have to be taken into account. To this end we consider the construction of a smooth permutation. The smoothness of an ordered set is expressed by the Total Variation of its elements:

$$\|\mathcal{T}\|_{TV} = \sum_{i=1}^{m-1} \|\mathbf{v}_i - \mathbf{v}_{i+1}\| \tag{2}$$

The optimal permutation operator $\mathbf{P}$ can be obtained by minimizing the total variation of $\mathbf{P}\mathcal{T}$:

$$\mathbf{P}^* = \arg\min_{\mathbf{P}} \|\mathbf{P}\mathcal{T}\|_{TV} \tag{3}$$

This optimization problem provides a permutation such that the corresponding Hamiltonian path is the shortest one and is equivalent to solve the traveling salesman problem. This problem being too computationally demanding for large sets, it can be solved using a randomized version of nearest neighbors heuristics as presented in [20]. We recall the principle of this algorithm in Algorithm 1. This algorithm starts from an arbitrary vertex and continues by finding the two nearest unexplored neighbor vertices and choosing one of them at random. Both these random choices make the obtained permutation stochastic.

Algorithm 1: Permutation construction as a global Hamiltonian path on a graph.

Input: set of vectors $\mathcal{T} = \{\mathbf{v}_1, \cdots, \mathbf{v}_m\}$ and graph $\mathcal{G}$;
Generate a random probability vector $\mathbf{p} = [p_1, \cdots, p_m]$;
Randomly choose an index $j \in \mathcal{J} = \{1, \cdots, m\}$;
Set $\sigma^{-1}(1) = j$ (or equivalently $\sigma(j) = 1$);
Set $\mathcal{P} = \{\mathbf{v}_{\sigma^{-1}(1)}\}$ and $\mathcal{J} = \{\sigma^{-1}(1)\}$;
for $i = 1$ *to* $m - 1$ **do**
  Let $N(v_{\sigma^{-1}(i)}) = \{k | (v_{\sigma^{-1}(i)}, v_k) \in \mathcal{E}\} \setminus \mathcal{I}$;
  if $|N(v_{\sigma^{-1}(i)})| = 1$ **then**
    $\sigma^{-1}(i+1) = N(v_{\sigma^{-1}(i)})$
  else
    if $|N(v_{\sigma^{-1}(i)})| \geq 2$ **then**
      Find the first v_{j_1} and second v_{j_2} nearest neighbors of $v_{\sigma^{-1}(i)}$ in $N(v_{\sigma^{-1}(i)})$
    else
      if $|N(v_{\sigma^{-1}(i)})| = 0$ **then**
        Find the first v_{j_1} and second v_{j_2} nearest neighbors of $v_{\sigma^{-1}(i)}$ in $\mathcal{J} \setminus \mathcal{I}$
      else
    $q_i = \frac{1}{1+exp\left(\frac{\|\mathbf{v}_{\sigma^{-1}(i)} - \mathbf{v}_{j_1}\| - \|\mathbf{v}_{\sigma^{-1}(i)} - \mathbf{v}_{j_2}\|}{\alpha}\right)}$;
    If $(q_i < p_{\sigma^{-1}(i)})$ then $\sigma^{-1}(i+1) = j_2$ and $\sigma^{-1}(i+1) = j_1$ otherwise;
  $\mathcal{P} = \mathcal{P} \cup \{\mathbf{v}_{\sigma^{-1}(i+1)}\}$;
  $\mathcal{I} = \mathcal{I} \cup \{\sigma^{-1}(i+1)\}$;
Output: ordered set $\mathcal{P}$ and associated index set $\mathcal{I}$

After the construction of the permutation, we define the h-ordering from the constructed permutation as $h(\mathbf{v}_i) = \sigma(i)$ and this defines the complete lattice $(\mathcal{T}, \leq_h)$. Given a graph signal $f : \mathcal{G} \to \mathcal{T}$, a new representation is obtained in the form of the pair $(\mathcal{I}, \mathcal{P})$ with $\mathcal{I}(v_i) = \sigma(i)$. When a graph signal is encoded in this way, the spectral information is not directly carried by the index $\mathcal{I}$, but is stored in a separate piece of data called a palette: the set $\mathcal{P}$ of sorted vectors. Figure 1 presents the obtained permutation and index from an original image, as well as the constructed Hamiltonian path, on a 8-adjacency grid graph $\mathcal{G}_1$. The original graph signal f can be directly recovered since $f(v_i) = \mathcal{P}[\mathcal{I}(v_i)] = \mathcal{T}[i] = \mathbf{v}_i$. It can be interesting to construct graphs that combine the use of both spatial and spectral proximity to infer the set of edges. For graphs $\mathcal{G}_*^K$, it is even recommended since their construction cannot ensure that the graph is connected. In this special case,

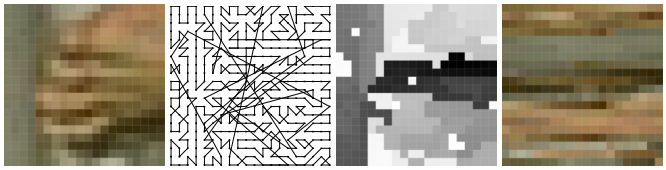

Fig. 1. From left to right: original image, an Hamiltonian path constructed on graph $\mathcal{G}_1$, the associated index and palette images.

Algorithm 1 will not be able to continue the construction of the path in the vertices of the neighborhood if they have been all explored. This case was foreseen in Algorithm 1, and in the case where $|N(v_{\sigma^{-1}(i)})| = 0$, meaning that no vertex is available to continue the path, the search of the nearest neighbor is performed on the set of all the remaining vertices $\mathcal{J} \backslash \mathcal{I}$. These cases can be easily seen as the "jumping" edges in the constructed path shown in Figure 1(b). In the sequel we will consider graphs $\mathcal{G}_{20} \cup \mathcal{G}_*^{21}$. Finally, the spectral proximity used to determine the nearest neighbors within the set of adjacent vertices can consider any distance measure. In particular, instead of using a classical L_2 distance between the vectors $\mathbf{v}_i$ associated to vertices, one can consider patches $\mathbf{p}_i^w = \big(f(x_i + t), \forall t \in [-w/2, w/2]^2\big)^T$ of size $w \times w$.

D. Constructing the probability density function

As Algorithm 1 starts from a random vertex, a different stochastic permutation ordering is obtained for each run of the algorithm. If this is problematic for the construction of morphological operators, we can make the most of it and propose an alternative formulation of the stochastic watershed [4]. In this latter approach M segmentations are obtained from N random markers to build a probability density function using a Parzen estimator to combine them altogether. This pdf is then combined with the initial gradient to produce a probabilistic gradient that can be used for watershed segmentation. The stochastic nature of this approach comes from the use of random markers. We propose a novel approach that we will call stochastic permutation ordering watershed. In our case the stochastic aspect will come from the generation of several permutation orderings from different starting vertices. The principle is the following. We built M stochastic permutation orderings using Algorithm 1. This provides M representations $(\mathcal{I}_i, \mathcal{P}_i)$ with $i \in [1, M]$. Figure 2 presents for an image, several different obtained indexes and permutations. As it can be seen, the orderings of the colors can be very different in the palette (or equivalently the lattice) because of the stochastic nature of the constructed Hamiltonian paths. As a consequence the index palette is also very different from one stochastic permutation ordering to another. Therefore the minima extracted from index images $\mathcal{I}_i$ will not be located at the same position in all index images and we can obtain different segmentations from these minima, as in the original stochastic watershed. However in our case the germs are not randomly generated but are the minima of the index images generated from stochastic permutation orderings. Figure 2 presents the obtained minima and watershed segmentations

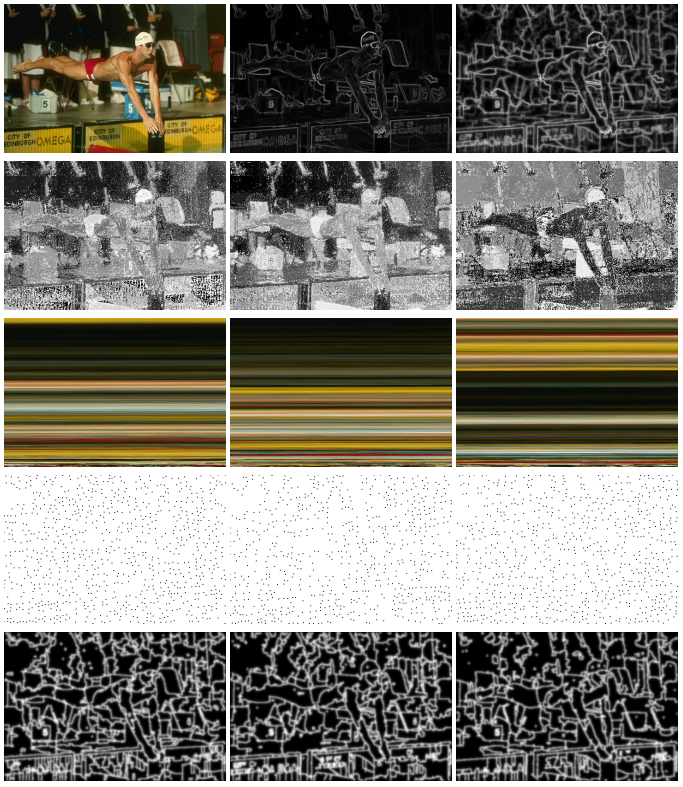

Fig. 2. First row: initial image f, Di Zenzo gradient ∇f, probabilistic gradient $\overline{\nabla f}$. Second row: stochastic orderings $\mathcal{I}_i$. Third row: stochastic palettes $\mathcal{P}_i$. Fourth row: minima $m(\mathcal{I}_i)$ of the stochastic orderings $\mathcal{I}_i$. Fifth row: watershed lines $WS_i(f)$ from the minima $m(\mathcal{I}_i)$ on the gradient ∇f.

for each of the obtained indexes with the Di Zenzo gradient [9]. For an image f, we denote as $WS_i(f) = WS(I_i, \nabla f)$ the watershed obtained from the minima $m(\mathcal{I}_i)$ of the ith stochastic ordering $\mathcal{I}_i$ on the gradient ∇f. Then, as in [4], the pdf is computed with the Parzen method:

$$\text{pdf}(f) = \frac{1}{M} \sum_{i=0}^{M} G(WS_i(f)) \tag{4}$$

with G a Gaussian kernel of variance σ^2 ($\sigma = 2$ in our case). Finally this pdf is combined with the initial Di Zenzo Gradient to produce a probabilistic gradient :

$$\overline{\nabla f} = \frac{\text{pdf}(f) + \nabla f}{2} \tag{5}$$

This enables to reinforce the gradients contours of high probability. First row of Figure 2 shows the obtained probabilistic gradient. This was obtained using 11 combined different stochastic permutations orderings even if only 3 are shown in the Figure. As it can be seen, the probabilistic gradient contains much more information than the classical Di Zenzo gradient. So far, the distance used to generate the different stochastic ordering in Algorithm 1 was based only on the color vectors of the vertices $\mathbf{v}_i = f(v_i)$. As previously explained, we can used instead a distance on other features such as the patches $\mathbf{p}_i^w$ of size $w \times w$ or structure tensors $\mathbf{t}_i$ of size 2×2. This directly enables to have a feature-based stochastic

watershed which is not possible with the original formulation of [4]. We investigate this in the next Section.

III. Examples

In this section we investigate the interest of the proposed stochastic permutation ordering watershed with respect to the original stochastic watershed [4]. We also investigate the interest of using patches or structure tensors instead of colors to compare vertices in Algorithm 1. We perform our experiments on images of the Bayeux Tapestry[1]. On such images, historian experts need to delineate interactively some characters. Therefore, a precise segmentation is required but by using simple object/background seed labeling by point click to ease the end-users use. The characters are visually easy to identify but the reduced number of colors, the fine embroidery as well as the texture differences in the linen fabric can make the segmentation hard. Figure 3 presents some results on four distinct parts of the tapestry. To obtain the segmentation, a watershed is performed on the probabilistic gradient from the user seeds. We compare our approach with the classical stochastic watershed. To have a fair comparison, for the computation of pdf(f) we consider the same number of random realizations than ours (11) and generate 200 random seeds. In contrast to [4], we do not add any pre-processing of the image. For each image to segment, Figure 3 presents on two lines, from left to right: the original image f, the user seeds, the probabilistic gradient $\overline{\nabla f}$ and the segmentation result for the classical stochastic watershed, our stochastic permutation ordering watershed with colors $\mathbf{v}_i$, 7×7 patches $\mathbf{p}_i^7$, 15×15 patches $\mathbf{p}_i^{15}$, and structure tensors $\mathbf{t}_i$. As it can be seen, the probabilistic gradient obtained from the classical stochastic watershed (CSW) has few regions in homogeneous areas but some visually important edges do not have a high probability. This is not the case of the probabilistic gradient obtained from our proposed stochastic permutation ordering watershed (SPOW). In the segmentation results, the blue squares do identify the strong segmentation errors. The results obtained with colors vectors for SPOW are always better than for CSW. One can see that for the observed errors this is due to not having edges of high probabilities. The use of 7×7 patches can further enhance the segmentation. However the use of very large 15×15 patches provides a result close to the CSW proving that if patches can be of interest, the size of the patterns of interest is very important. Finally with structure tensors, the probabilistic gradient is very fine, but this tends to over detect structures in textured areas. As a conclusion, for this kind of images the use of our proposed SPOW with 7×7 patches provides the best results and this shows the interest of our proposal.

IV. Conclusion

We have proposed an alternative formulation to the stochastic watershed. Starting from a random pixel, an Hamiltonian path is constructed to build a complete lattice. From the latter a new representation of an image in the form of a palette and an index image is obtained. Minima are extracted from the latter and used to perform several watershed that are combined into a probability density function. A probabilistic gradient is computed from the obtained pdf and the Di Zenzo gradient. The advantage of the proposed approach relies in its ability to incorporate any features to represent pixels for the computation of the pdf. The experimental results have shown that this has some benefits with respect to the classical stochastic watershed.

[1] Panorama of the Bayeux Tapestry, GREYC Image (UMR UNICAEN/ENSICAEN/CNRS), CERTIC (UNICAEN) from the 2017 photographic campaign of La Fabrique de patrimoines en Normandie, 2017-2018.

References

[1] F. Meyer and S. Beucher, "Morphological segmentation," *J VIS COMMUN IMAGE R*, vol. 1, no. 1, pp. 21–46, 1990.

[2] P. Soille, *Morphological Image Analysis: Principles and Applications*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2003.

[3] L. Vincent and P. Soille, "Watersheds in digital spaces: an efficient algorithm based on immersion simulations," *IEEE T PATTERN ANAL*, vol. 13, no. 6, pp. 583–598, 1991.

[4] J. Angulo and D. Jeulin, "Stochastic watershed segmentation," in *ISMM*, 2007, pp. 265–276.

[5] F. Malmberg and C. L. L. Hendriks, "An efficient algorithm for exact evaluation of stochastic watersheds," *PATTERN RECOGN LETT*, vol. 47, pp. 80–84, 2014.

[6] F. López-Mir, V. Naranjo, S. Morales, and J. Angulo, "Probability density function of object contours using regional regularized stochastic watershed," in *IEEE ICIP*, 2014, pp. 4762–4766.

[7] G. Franchi and J. Angulo, "Bagging stochastic watershed on natural color image segmentation," in *ISMM*, vol. LNCS 9082, 2015, pp. 422–433.

[8] J. Angulo, "Statistical gaussian model of image regions in stochastic watershed segmentation," in *GSI*, vol. LNCS 9389, 2015, pp. 396–405.

[9] S. D. Zenzo, "A note on the gradient of a multi-image," *COMPUT VIS IMAGE UND*, vol. 33, no. 1, pp. 116–125, 1986.

[10] A. Buades, B. Coll, and J.-M. Morel, "Image denoising methods. a new nonlocal principle," *SIAM Review*, vol. 52, no. 1, pp. 113–147, 2010.

[11] O. Lézoray, "Manifold-based mathematical morphology for graph signal editing of colored images and meshes," in *IEEE SMC*, 2016, pp. 109–114.

[12] A. Elmoataz, O. Lezoray, and S. Bougleux, "Nonlocal discrete regularization on weighted graphs: a framework for image and manifold processing," *IEEE T IMAGE PROCESS*, vol. 17, no. 7, pp. 1047–1060, 2008.

[13] C. Ronse, "Why mathematical morphology needs complete lattices," *SIGNAL PROCESS*, vol. 21, no. 2, pp. 129–154, 1990.

[14] V. Barnett, "The ordering of multivariate daa," *J R STAT SOC SER A-G*, vol. 139, no. 3, pp. 318–355, 1976.

[15] E. Aptoula and S. Lefèvre, "Multivariate mathematical morphology applied to colour image analysis," in *Multivariate image processing: methods and applications*, C. Collet, J. Chanussot, and K. Chehdi, Eds. ISTE - John Wiley, 2009, pp. 303–337.

[16] S. Velasco-Forero and J. Angulo, "On nonlocal mathematical morphology," in *ISMM*, 2013, pp. 219–230.

[17] O. Lézoray, "Complete lattice learning for multivariate mathematical morphology," *J VIS COMMUN IMAGE R*, vol. 35, pp. 220–235, 2016.

[18] J. Goutsias, H. Heijmans, and K. Sivakumar, "Morphological operators for image sequences," *COMPUT VIS IMAGE UND*, vol. 62, no. 3, pp. 326–346, 1995.

[19] J. Chanussot and P. Lambert, "Bit mixing paradigm for multivalued morphological filters," in *International Conference on Image Processing and Its Applications*, vol. 2, 1997, pp. 804 – 808.

[20] I. Ram, M. Elad, and I. Cohen, "Image processing using smooth ordering of its patches," *IEEE T IMAGE PROCESS*, vol. 22, no. 7, pp. 2764–2774, 2013.

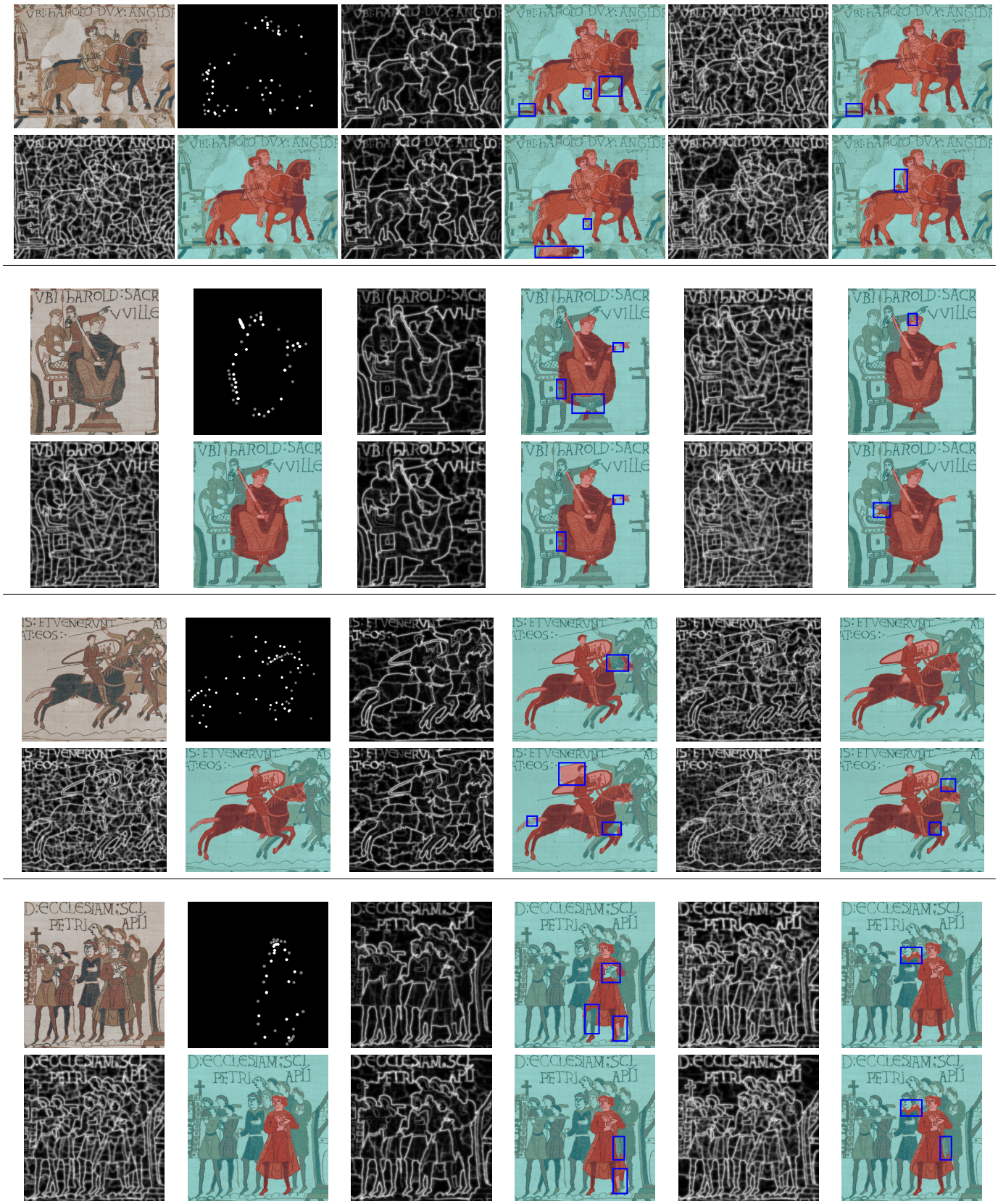

Fig. 3. Segmentation results, with for each image, on two lines, from left to right: the original image f, the user seeds, the probabilistic gradient $\overline{\nabla f}$ and the segmentation result for the classical stochastic watershed, our stochastic permutation ordering watershed with colors $\mathbf{v}_i$, 7×7 patches $\mathbf{p}_i^7$, 15×15 patches $\mathbf{p}_i^{15}$, and structure tensors $\mathbf{t}_i$.