TEXTURE SEGMENTATION WITH DIFFUSIONS ON ARBITRARY GRAPHS.

X. Dong and I. Pollak

Purdue University
School of Electrical and Computer Engineering
West Lafayette, IN 47907

ABSTRACT

We propose a novel multiscale segmentation algorithm based on an extension of stabilized inverse diffusion equations [10]. The new algorithm is capable of processing vector-valued images defined on arbitrary graphs. We illustrate it through segmenting texture mosaics and natural images.

1. INTRODUCTION.

Stabilized inverse diffusion equations (SIDEs) were developed in [10] for segmentation and restoration of grayscale images and signals. The solution of such an equation is a family of piecewise-constant coarsenings of the input data. The forward diffusion for such an equation can be viewed as a recursive region merging process.

Section 2 shows how to adapt SIDEs to the processing of vector-valued images defined on arbitrary graphs, resulting in a region merging segmentation algorithm for such images. Possible applications, illustrated in Section 3, include segmentation of vector-valued images consisting of features vectors extracted from a textured image.

2. SIDES FOR VECTOR-VALUED IMAGES DEFINED ON AN ARBITRARY DOMAIN.

We define a vector-valued image on an arbitrary finite set $\mathcal{N}$ of points as any function which assigns a vector of K real numbers to every point in $\mathcal{N}$. In our segmentation tasks, it is important to define adjacency relationships on the points in $\mathcal{N}$, and therefore we assume that $\mathcal{N}$ is the set of nodes of an undirected graph $\mathcal{G} = (\mathcal{N}, \mathcal{L})$ where the set of *links* $\mathcal{L}$ consists of unordered pairs of distinct nodes. If $\{m, n\}$ is a link, we say that the nodes m and n are *neighbors*. For example, $\mathcal{G}$ could be a finite 2D rectangular grid where each node has four neighbors: east, west, north, and south.

We let N be the total number of nodes and, without loss of generality, denote the nodes by the integers $1, 2, \ldots, N$, i.e., $\mathcal{N} = \{1, 2, \ldots, N\}$. An image $\mathbf{u}$ can be thought of as an $N \times K$-dimensional vector: $\mathbf{u} = (\vec{u}_1, \ldots, \vec{u}_N) \in \mathbb{R}^{N \times K}$, where each *intensity vector* $\vec{u}_n = (u_{n,1}, \ldots, u_{n,K})$ is in $\mathbb{R}^K$. An intensity vector can, for example, be the vector of red, green, and blue intensities for a color image, or correspond to K features extracted from a texture image. Each entry of an intensity vector is called a *channel*.

To define a metric on the space $\mathbb{R}^K$ of all intensity vectors, we use the following inner product between two intensity vectors $\vec{u}_m$ and $\vec{u}_n$:

$$\langle \vec{u}_m, \vec{v}_n \rangle_w \triangleq \sum_{k=1}^{K} u_{m,k} w_k u_{n,k}, \tag{1}$$

where the weights $w = (w_1, \ldots, w_K)$ allow us to assign different relative importance to different channels of an image. We denote the corresponding norm by $\| \cdot \|_w$.

To specify the scale-space that we use in this paper, we need the following further definitions. We say that a set of nodes $R \subset \mathcal{N}$ is a *region* if, for any two nodes $m, n \in R$, there exists a sequence of links connecting m and n. Two regions R_1, R_2 are called *neighbors* if there exist two nodes $m \in R_1$, $n \in R_2$ which are neighbors, i.e., such that $\{m, n\} \in \mathcal{L}$. We let NBR-PAIRS be the set of all pairs of neighbor regions, and we let NBRS(R_i) be the set of all regions that are neighbors of a region R_i.

For any partition $\mathcal{S} = \{R_1, \ldots, R_I\}$ of the set $\mathcal{N}$ of nodes, we let $U_{\mathcal{S}}$ be the set of all piecewise constant images which are constant over each region $R_i \in \mathcal{S}$. We use $\vec{\mu}_i(\mathbf{u})$ to denote the intensity vector of any such image $\mathbf{u}$ within region R_i. Note that $U_{\mathcal{S}}$ is a vector space. To impose a metric on this space, we define the following inner product.

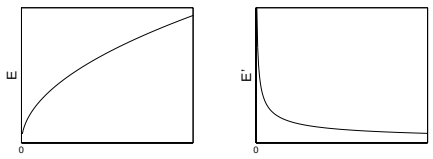
$$\langle \mathbf{u}, \mathbf{v} \rangle \triangleq \sum_{i=1}^{I} a(R_i)\ \langle \vec{\mu}_i(\mathbf{u}), \vec{\mu}_i(\mathbf{v}) \rangle_w, \tag{2}$$

where $\langle \cdot \rangle_w$ is the inner product for the intensity vectors defined in Eq. (1), and where $a(R_i)$ is a nonnegative weight function which enables us to weight the contributions of various regions differently. For example, $a(R_i)$ may encode some information about the shape of R_i.

We consider functionals $\mathcal{E}$ defined for images in $U_{\mathcal{S}}$ which have the following form:

$$\mathcal{E}(\mathbf{u}) = \sum_{(R_i, R_j) \in \text{NBR-PAIRS}} b(R_i, R_j)\ E(\|\vec{\mu}_i(\mathbf{u}) - \vec{\mu}_j(\mathbf{u})\|_w),$$

This work was supported in part by a National Science Foundation CAREER award CCR-0093105 and an NSF grant IIS-0329156.



(a) (b)

Fig. 1: (a) The SIDE energy function E, also encountered in the models of [4, 13]. (b) Its derivative E'.

where $b(R_i, R_j)$ is a nonnegative weight function, and $E(\cdot)$ is monotonically increasing on $[0, +\infty)$, with $E(0) = 0$.

Given an image $\mathbf{u}(0) \in U_{\mathcal{S}}$, we generate a scale-space $\mathbf{u}(t)$ by solving the following gradient descent procedure for $t > 0$:

$$\dot{\mathbf{u}}(t) = -\nabla \mathcal{E}(\mathbf{u}(t)), \tag{3}$$

where $\dot{\mathbf{u}}(t)$ is the derivative of $\mathbf{u}(t)$ with respect to the scale parameter t, and ∇ stands for the gradient in the space $U_{\mathcal{S}}$ equipped with the inner product (2). The following proposition shows how to implement this descent equation.

Proposition 1. *The gradient descent procedure (3) can be equivalently written*[1] *as follows:*

$$\dot{\vec{\mu}}_i = \frac{1}{a(R_i)} \sum_{R_j \in \text{NBRS}(R_i)} b(R_i, R_j) \frac{\vec{\mu}_j - \vec{\mu}_i}{\|\vec{\mu}_j - \vec{\mu}_i\|_w} \cdot E'(\|\vec{\mu}_j - \vec{\mu}_i\|_w), \text{ for } i = 1, \ldots, I.$$

3. SEGMENTATION EXAMPLES.

We adopt Gabor energy features (GEF) from [5] to generate the vector-valued feature image which serves as the initial data for Eq. (3). We start the evolution of (3) with every pixel being a separate region. When the intensities of two neighbor regions become equal during the evolution, these two regions get merged into one, and the gradient descent equation (3) proceeds using the new set of regions. To encourage eventual merging of every pair of neighbor regions, we use a SIDE energy function E [10]—i.e., such function that its derivative E' is monotonically nonincreasing on $(0, +\infty)$, see Fig. 1. In all the experiments below, we assume that the correct number of regions is known, and we stop the evolution when this number of regions is reached.

The function $a(R_i)$ that we use encourages relatively rapid evolution of small regions. Our function $b(R_i, R_j)$ is the length of the boundary between R_i and R_j. The weights w_k are used to emphasize coarser-scale features during late stages of the evolution.

Experiment 1: Two Textures, Straight Boundary. We form 90 two-texture test images using all pairs of ten Brodatz textures [2]. Each test image is obtained by concatenating two 256×256 texture images. Our nonlinear diffusion is evolved until two regions remain. Our segmentations are very accurate, with only 0.6% of the pixels misclassified, on average, with 0.4% standard deviation. The worst result among the 90 experiments is 3.6% misclassifications.

Experiment 2: Segmentation of Natural Images. Segmentation of natural images is important since it is often used as the first stage of image analysis algorithms for various tasks such as database organization and retrieval, classification, detection and recognition of objects in images, compression, etc. Fig. 2 illustrates the performance of our algorithm on several images from the Berkeley segmentation dataset [1, 6]. While precise quantitative evaluation of the performance of our algorithm on such images is beyond the scope of this paper (indeed, analyzing the performance of a segmentation algorithm on a natural image is a challenging open problem [6, 12]), note that these segmentations are comparable to the ones produced by recent algorithms such as [3,7–9,11]. Comparing to the results reported in [11], note that our algorithm captures the outline of the small birds in the center of the top left image while the algorithm in [11] does not; in the church image, our algorithm accurately captures the outlines of the two crosses while the algorithm in [11] does not. Our segmentations of the leopard and bear images are very similar to [11]; on the other hand, in the deer image the algorithm in [11] is able to segment the legs of the small deer while our algorithm is not.

[1] Here, we abbreviate $\vec{\mu}_i(\mathbf{u}(t))$ as $\vec{\mu}_i$.

4. REFERENCES

[1] The Berkeley segmentation dataset and benchmark. www.cs.berkeley.edu/projects/vision/grouping/segbench

[2] P. Brodatz. *A photographic album for artists and designers.* Dover, New York, 1966.

[3] M. Galun, E. Sharon, R. Basri, and A. Brandt. Texture segmentation by multiscale aggregation of filter responses and shape elements. In *Proc. of IEEE International Conference on Computer Vision*, pages 716-723, Nice, France, 2003.

[4] D. Geman and G. Reynolds. Constrained restoration and the recovery of discontinuities. *IEEE Trans. on PAMI*, 14(3), 1992.

[5] P. Kruizinga and N. Petkov. Nonlinear operator for oriented texture. *IEEE Trans. on Image Processing*, 8(10), 1999.

[6] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proc. IEEE International Conference on Computer Vision*, pages 416-423, July 2001.

[7] R. Nock and F. Nielsen. Statistical region merging. *IEEE Trans. on PAMI*, 26(11):1452-1458, November 2004.

[8] R.J. O'Callaghan and D.R. Bull. Combined morphological-spectral unsupervised image segmentation. *IEEE Trans. on Image Processing*, 14(1):49-62, January 2005.

[9] A. Petrovic, O.D. Escoda, and P. Vandergheynst. Multiresolution segmentation of natural images: From linear to nonlinear scale-space representations. *IEEE Trans. on Image Processing*, 13(8):1104-1114, August 2004.

[10] I. Pollak, A. S. Willsky, and H. Krim. Image segmentation and edge enhancement with stabilized inverse diffusion equations. *IEEE Trans. on Image Processing*, 9(2), February 2000.

[11] X. Ren and J. Malik. Learning a classification model for segmentation. In *Proc. IEEE International Conf. on Computer Vision*, pages 10-17, Nice, France, 2003.

[12] Y. Rubner, J. Puzicha, C. Tomasi, J.M. Buhmann. Empirical evaluation of dissimilarity measures for color and texture. *Computer Vision and Image Understanding*, 84(1):25-43, October 2001.

[13] S.C. Zhu and D. Mumford. Prior learning and Gibbs reaction-diffusion. *IEEE Trans. on PAMI*, 19(11), 1997.

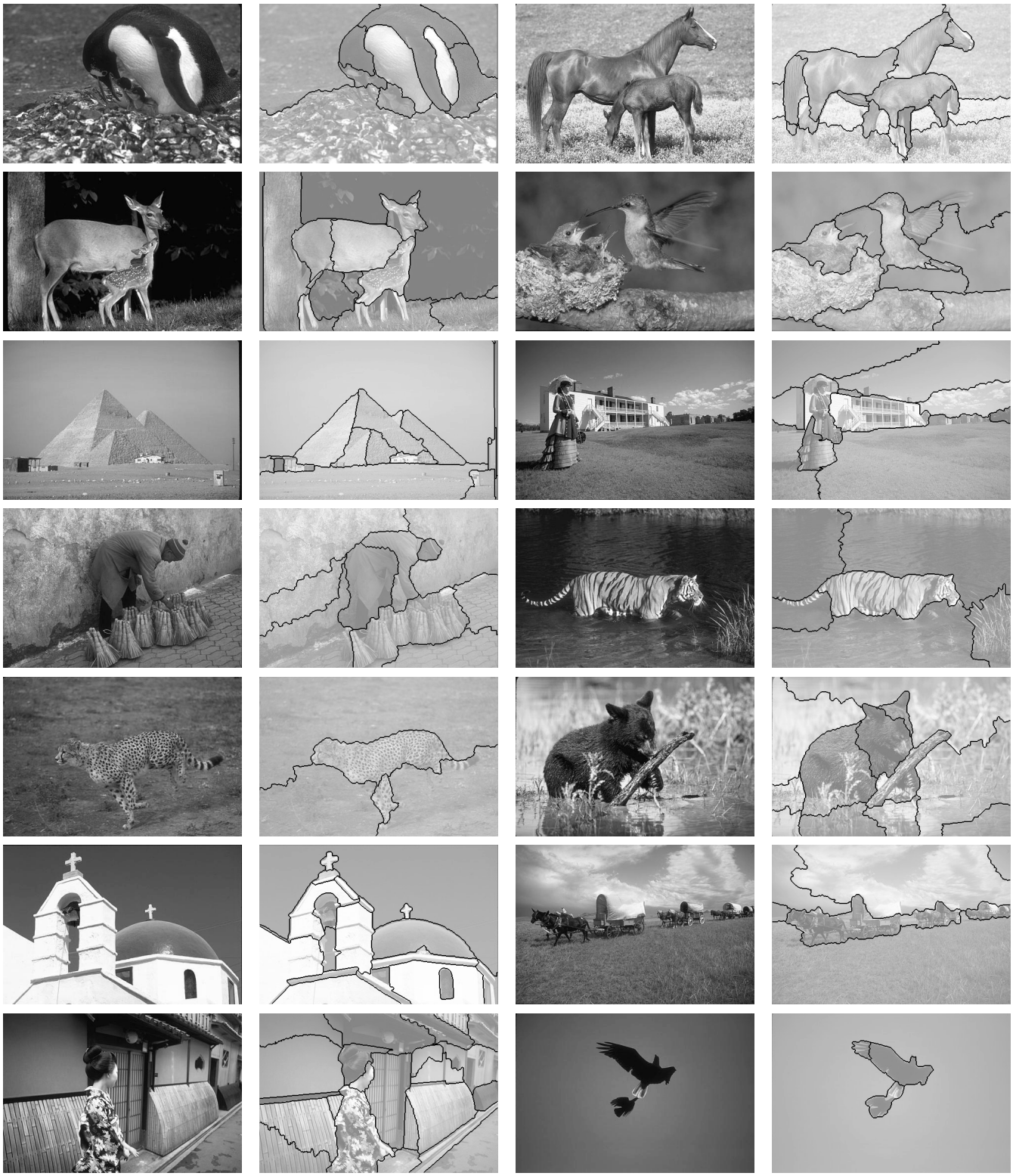


Fig. 2: Segmentation of natural images from the Berkeley segmentation dataset [1, 6].