SIAM J. IMAGING SCIENCES
Vol. 5, No. 3, pp. 1022–1054

Nonlocal Active Contours[*]

Miyoun Jung[†], Gabriel Peyré[†], and Laurent D. Cohen[†]

Abstract. This article introduces a novel class of active contour models for image segmentation. It makes use of nonlocal comparisons between pairs of patches within each region to be segmented. The corresponding variational segmentation problem is implemented using a level set formulation that can handle an arbitrary number of regions. The pairwise interaction of features constrains only the local homogeneity of image features, which is crucial in capturing regions with smoothly spatially varying features. This segmentation method is generic and can be adapted to various segmentation problems by designing an appropriate metric between patches. We instantiate this framework using several classes of features and metrics. Piecewise smooth grayscale and color images are handled using L^2 distance between image patches. We show examples of efficient segmentation of natural color images. Locally oriented textures are segmented using the L^2 distance between patches of Gabor coefficients. We use a Wasserstein distance between local empirical distributions for locally homogeneous random textures. A correlation metric between local motion signatures is able to segment piecewise smooth optical flows.

Key words. image segmentation, active contours, level sets, nonlocal method, patch, Wasserstein distance, texture, motion, multiphase

AMS subject classification. 68U10

DOI. 10.1137/11085863X

1. Introduction.

Image segmentation is a fundamental imaging problem that requires an efficient modeling of image and texture features. This article proposes a novel class of active contour models that unifies patch processing and piecewise regular image models. It makes use of nonlocal comparisons between pairs of patches within the segmented regions. We thus refer to it as a "nonlocal" approach, following the terminology initiated in [7]. The model provides a general framework since it can be adapted to various segmentation problems by designing a metric between patches.

1.1. Edge-based versus region-based active contours.

Many existing active contour methods segment an image according to either edge or region information. The evolution of the segmenting curve is driven by the minimization of some variational energy that takes into account this information. A popular way to numerically implement such an active contour evolution is though the level set framework of Osher and Sethian [39].

Edge-based active contours. Edge-based active contour models use some edge detector and evolve the segmenting curve towards sharp gradients of pixel intensity. Starting from the snakes model of Kass, Witkin, and Terzopoulos [28], many edge-based active contour models

---

[*] Received by the editors December 12, 2011; accepted for publication (in revised form) May 17, 2012; published electronically September 4, 2012. This work was partially supported by Fondation Sciences Mathématiques de Paris, ANR grant NatImages ANR-08-EMER-009, and ANR grant MESANGE ANR-08-BLAN-0198.
http://www.siam.org/journals/siims/5-3/85863.html

[†] CEREMADE, Université Paris-Dauphine, 75775 Paris Cedex 16, France (jung@ceremade.dauphine.fr, peyre@ceremade.dauphine.fr, cohen@ceremade.dauphine.fr).

have been proposed, in particular geometric snakes based on geodesic energies [10, 14, 36, 11, 29].

Region-based active contours. These local edge-based models tend to be sensitive to the noise that affects edge detectors. To avoid the difficult problem of edge localization, region-based active contour models incorporate more global information to obtain segmented regions with homogeneous features. One of the first region-based methods is the Mumford–Shah model [37], where the image is approximated using a smooth function inside each region. Many variants of this initial model have been proposed that allow simpler and more efficient implementations [15]. Chan and Vese [13] proposed an active contour method which approximates an image with a constant function inside each region. This Chan–Vese (CV) model has been extended to deal with vector-valued images [12] and textures [47].

Hybrid active contours. Several hybrid models have been proposed to make use of both local (edge-based) and global (region-based) information. Kimmel [31] designed an energy mixing both the geodesic and the CV energies. Sagiv, Sochen, and Zeevi [45] used a similar energy (integrated active contours, or IAC) that is also extended to vector-valued images and texture segmentation.

1.2. Piecewise smooth active contours. Natural images exhibit smooth variations of feature values (e.g., color gradients or texture orientation variations) over both the objects composing the scene and the background. It is thus important to design segmentation methods that can handle piecewise smooth images.

The initial Mumford–Shah approach [37] takes into account piecewise smooth images and has been implemented, for instance, in [3, 51, 52]. This model, however, remains difficult to implement numerically. A simpler class of piecewise smooth models has been introduced more recently [6, 43, 55]. These models estimate in parallel to the segmentation process a piecewise smooth parameter field such as the local mean of the features. Of particular interest is the locally binary fitting (LBF) model [34, 54, 53] that shares some similarity with our method.

Our nonlocal model also tackles the problem of designing piecewise smooth active contours, but it uses a different approach that does not require the estimation of a spatially varying parameter field. A chief advantage of our method is that it only requires the design of a metric to compare patches extracted around each pixel. It is thus easy to apply the method to a wide variety of features, and we show in the numerical results several instances of our method for pixels, textures, local statistics, and video features.

1.3. Multiphase active contours. Several methods have been proposed to extend general binary segmentation energies to the case of an arbitrary number of regions; see, for instance, [57, 56, 41, 46, 52]. In this paper we focus on two particular level set methods. The approach of Samson et al. [46] introduces a single level set function per region and makes use of an additional repulsive energy to avoid the overlap of the regions. The approach of Chan and Vese [52] requires only $\log_2(N)$ level set functions to partition the image into N nonoverlapping regions depending on all possible combinations of the level set signs.

We extend our nonlocal energy in a multiphase framework using two different approaches borrowed from [52] and [46]. An important benefit of using these nonlocal multiphase energies is that a smaller number of level set functions is required to perform complicated segmentations into multiple regions. Each of these approaches has advantages and inconveniences, and we show in the numerical examples how our method fits perfectly into these two frameworks.

1.4. Nonlocal image processing. Nonlocal processing refers to the general methodology of designing energies using nonlocal comparisons of patches extracted in the image. Starting from the initial paper of Buades, Coll, and Morel [7], nonlocal energies have proved to be efficient for many imaging problems, including denoising [7, 32], inverse problems [22, 42], semisupervised classification [21, 19, 24], and unsupervised segmentation [4, 9].

In our work, we use a nonlocal energy that enforces the nonlocal similarity of patches inside each region to be segmented. This patch comparison principle drives the active contour to optimize the homogeneity of each region. Note that this approach differs significantly from previous patch-based segmentation methods such as [21, 19, 24, 4, 9] where nonlocal energies are used to regularize the contour.

1.5. Examples of image features. All of the previous methods have been implemented using a wide variety of feature spaces, depending on the specificity of the images to be segmented. Besides classical gray-values and color features (which are usually compared using an L^2 norm), we detail below some popular features that we consider in the applications and numerical illustrations of our method.

Gabor features. For locally oriented texture segmentation, a popular class of features is computed as the output of filter banks. Gabor filters [20] have been used to discriminate the local orientation and frequency of the texture [33, 40, 47, 45]. The CV energy [12] has been used by Sandberg, Chan, and Vese [47] to impose a global homogeneity of the texture orientation in each region. The IAC of Sagiv, Sochen, and Zeevi [45] extends the hybrid region/edge-based active contour of [31] to deal with vector-valued features.

Statistical features. To deal with more random textured images, it is possible to make use of the empirical statistics within each region to be segmented. The resulting statistical region-based active contours make use of pointwise similarity measures between distributions (such as the Kullback–Leibler divergence) to compare the distributions in a parametric or nonparametric (using Parzen windows) fashion; see, for instance, [30, 58, 41, 23]. Traditional pointwise statistical distances are simple to compute but are unstable when using localized distributions. To address these issues, Ni et al. [38] propose using the L^1 Wasserstein distance in order to extend the segmentation model of Chan and Vese [13]. This work is extended to color image segmentation in [2] using a Wasserstein metric only on the brightness channel, thus resulting in a one-dimensional (1-D) optimal transport metric. The Wasserstein metric fits nicely into our nonlocal framework, and we propose using an approximate distance, initially introduced in [44], to handle features in arbitrary dimensions (e.g., colors in three dimensions).

Motion features. Motion segmentation aims at grouping together pixels undergoing the same motion. Most existing works jointly compute an optical flow and a segmentation [17, 5]. Shi and Malik [48] propose avoiding the difficult estimation of the optical flow. They perform the segmentation using features representing a local probability distribution of motions. We make use of the same local features that are integrated into our nonlocal energy. The resulting method is able to detect moving objects with a smoothly varying motion.

1.6. Contributions. The main contribution of this paper is a novel class of segmentation energies that impose a local homogeneity of patch features. It is thus a nonlocal method (because it compares patches that are not necessarily neighbors) but one that enables a fine tuning of the scale at which these patches are compared (hence the local homogeneity).

Let us note that we use a mathematically rigorous derivation of the evolution PDEs using shape gradient flows. To the best of our knowledge, it is the first time multiphase evolutions are performed within such a shape gradient framework.

A second contribution of the paper is a systematic exploration of this approach on a wide range of segmentation problems, which includes grayscale and color images, textures, random vector fields, and videos. We show comparisons with several other popular edge-based, region-based, and hybrid methods (extended to patches) to highlight the benefit of patch comparisons.

1.7. Organization. Section 2 recalls the generic variational level set framework that we use to implement the active contour evolutions and presents our computation of the gradient of new energy terms we use in our models. Section 3 introduces our new model, first in a simplified (unnormalized) case, and then in a more advanced (normalized) setting. Section 4 describes the competing methods that we have tested in the numerical sections and how we extend them to use patches with arbitrary metrics. Sections 5–8 show numerical examples of applications of our method to various settings (image, textures, videos). Section 9 details two different extensions of our method to handle an arbitrary number of regions.

Note that some early results making use of this nonlocal segmentation method were presented in the conference papers [25, 26, 27].

2. Variational image segmentation.

This section recalls some basic notions about image segmentation as well as more advanced concepts using shape gradients. These will be used in the remaining part of the paper.

2.1. Variational minimization. The goal is to segment an input image f^0 using a feature map $f : [0,1]^2 \to \mathbb{R}^d$ computed from f^0, where d is dimensionality of the feature space. Since we aim at proposing a generic segmentation framework, we do not specify the exact nature of the features in this section or how f is computed from f^0. Several examples of features are detailed in sections 5–8.

Let $\Omega \subset [0,1]^2$ be some region that captures the objects of interest. We aim at finding a contour that represents the boundary $\partial\Omega$ of the region. We consider a variational minimization problem

$$\min_{\Omega} \left\{ \mathcal{E}(\Omega) = E(\Omega) + \gamma L(\Omega) \right\}, \tag{2.1}$$

where E is a region energy that we will define in the following sections, and L is a smoothing term that regularizes the boundary $\partial\Omega$ of the region. In the numerical experiments, $L(\Omega) = |\partial\Omega|$ is simply the length of the boundary. The parameter $\gamma > 0$ should be adapted to the expected regularity of the boundary of the region.

2.2. Parametric active contours with shape gradients. A mathematically sound way to derive a minimizing PDE of the energy (2.1) makes use of the so-called shape gradients; see, for instance, [1, 23, 18].

The shape gradient $\mathcal{E}'(\Omega)$ of an energy $\mathcal{E}$ evaluated at Ω is a scalar field $\mathcal{E}'(\Omega)(x)$ defined on each point of the boundary $x \in \partial\Omega$. It is characterized by introducing a family of smooth deformations $\Omega_\tau = T(\tau, \Omega)$ for $\tau \geqslant 0$, where $T : \mathbb{R}_+ \times [0,1]^2 \to [0,1]^2$ and $T(0,x) = x$, so that $\Omega_0 = \Omega$. We denote by $v(x) = \frac{\partial T}{\partial \tau}(0,x)$ the associated velocity field. The energy $\mathcal{E}$ has a shape gradient $\mathcal{E}'(\Omega)$ at some Ω if it satisfies

$$\mathcal{E}(\Omega_\tau) = \mathcal{E}(\Omega) + \tau\langle v,\, \mathcal{E}'(\Omega)n_\Omega\rangle_{\partial\Omega} + o(\tau) \tag{2.2}$$

(see [1]), where n_Ω is the unit normal of $\partial\Omega$ pointing outward, and the inner product along the boundary between two vector fields $v(x), w(x) \in \mathbb{R}^2$ is

$$\langle v,\, w\rangle_{\partial\Omega} = \int_{\partial\Omega} \langle v(x),\, w(x)\rangle \mathrm{d}x. \tag{2.3}$$

A parametric active contour is represented for each time $t \geqslant 0$ of the evolution by an explicit parameterization $s \mapsto \gamma(s,t)$ of the boundary $\partial\Omega$ of Ω. The shape gradient defines an evolution equation that minimizes the energy $\mathcal{E}$:

$$\frac{\partial \gamma}{\partial t}(s,t) = -\mathcal{E}'(\Omega)(\gamma(s,t))n_\Omega(\gamma(s,t)), \tag{2.4}$$

where Ω is the shape associated to $\gamma(\cdot,t)$ at time t. The curve $\gamma(\cdot,t)$ converges as $t \to +\infty$ to the boundary of some Ω which is a stationary point of $\mathcal{E}$ (it satisfies $\mathcal{E}'(\Omega) = 0$).

2.3. Level set active contours. In order to compute numerically a contour evolution to minimize (2.1), following [39, 13], the segmented region Ω is represented using a level set function $\varphi : [0,1]^2 \to \mathbb{R}$ so that $\Omega = \{x \setminus \varphi(x) > 0\}$.

The descent direction $-\mathcal{E}'(\Omega)n_\Omega$ is extended to level sets by selecting a valid level set gradient $\nabla\mathcal{E}(\varphi)$ so that

$$\forall\, x \in \partial\Omega, \quad \nabla\mathcal{E}(\varphi)(x) = \|\nabla\varphi(x)\|\mathcal{E}'(\Omega)(x). \tag{2.5}$$

The parametric evolution (2.4) is turned into a level set evolution (2.6) of $\varphi(x,t)$ with an artificial time $t \geqslant 0$:

$$\frac{\partial \varphi}{\partial t} = -\nabla\mathcal{E}(\varphi) = -\left(\nabla E(\varphi) + \gamma\nabla L(\varphi)\right). \tag{2.6}$$

We detail in the following section the expressions of level set shape gradients in some particular cases that will be used in the remainder of this paper.

2.4. Shape gradient examples. For a boundary energy written as

$$L_g(\Omega) = \int_{\partial\Omega} g(x)\mathrm{d}x, \tag{2.7}$$

where $\mathrm{d}x$ refers to 1-D integration along the boundary of Ω, a valid shape gradient extended to a level set function reads [1]

$$\nabla L_g(\varphi) = -\|\nabla\varphi\|\mathrm{div}\left(g\frac{\nabla\varphi}{\|\nabla\varphi\|}\right). \tag{2.8}$$

For instance, in the original minimization (2.1), using the length of the boundary, one has $L = L_g$ with $g = 1$.

The following proposition gives a level set gradient for the simplest region-based energy written as

$$\mathcal{E}_g(\Omega) = \int_\Omega g(x)\mathrm{d}x \tag{2.9}$$

(note that g does not depend on Ω).

Proposition 1. *A level set gradient of $\mathcal{E}_g$ reads as*

$$\nabla\mathcal{E}_g(\varphi) = \|\nabla\varphi\| g. \tag{2.10}$$

Proof. The usual balloon force corresponds to $g = 1$ [16]; see, for instance, [1] for the general case. ■

In the following propositions, we derive the level set gradients of two types of region-based energies. We first consider a region-based energy defined using pairwise interactions terms:

$$\mathcal{F}_g(\Omega) = \int_{\Omega\times\Omega} g(x, y)\mathrm{d}x\mathrm{d}y. \tag{2.11}$$

Proposition 2. *A level set shape gradient of $\mathcal{F}_g$ is*

$$\nabla\mathcal{F}_g(\varphi)(x) = \|\nabla\varphi(x)\| \int_\Omega \big(g(x, y) + g(y, x)\big)\mathrm{d}y. \tag{2.12}$$

Proof. One has $\mathcal{F}_g(\Omega) = \int_\Omega G_x(\Omega)\mathrm{d}x$, where $G_x(\Omega) = \int_\Omega g(x, y)\mathrm{d}y$. Note that for a fixed x, $G_x(\Omega)$ is of the form (2.9), and following (2.10), its shape gradient is $G'_x(\Omega) = g(x, \cdot)$. Thus, if τ is a smooth deformation of Ω and v is its velocity field, then

$$G_x(\Omega_\tau) = G_x(\Omega) + \tau\langle v,\, g(x, \cdot)n_\Omega\rangle_{\partial\Omega} + o(\tau), \tag{2.13}$$

following from (2.2). Integrating with respect to x, the previous expression on Ω_τ gives

$$\begin{aligned}\mathcal{F}_g(\Omega_\tau) &= \int_{\Omega_\tau} G_x(\Omega)\mathrm{d}x + \tau\Big\langle \int_{\Omega_\tau} g(x, \cdot)\mathrm{d}x\, n_\Omega, v\Big\rangle_{\partial\Omega} + o(\tau)\\ &= \int_\Omega\int_{\Omega_\tau} g(y, x)\mathrm{d}y\mathrm{d}x + \tau\Big\langle \int_{\Omega_\tau} g(x, \cdot)\mathrm{d}x\, n_\Omega, v\Big\rangle_{\partial\Omega} + o(\tau).\end{aligned} \tag{2.14}$$

Once again the inner term $\int_{\Omega_\tau} g(y, x)\mathrm{d}y$ in (2.14) can be expressed, similarly to (2.13), as

$$\int_{\Omega_\tau} g(y, x)\mathrm{d}y = \int_\Omega g(y, x)\mathrm{d}y + \tau\langle v,\, g(\cdot, x)n_\Omega\rangle_{\partial\Omega} + o(\tau). \tag{2.15}$$

This derives the equation for the shape gradient of the energy $\mathcal{F}_g(\Omega_\tau)$:

$$\mathcal{F}_g(\Omega_\tau) = \mathcal{F}_g(\Omega) + \tau\Big\langle \int_\Omega \big(g(x, \cdot) + g(\cdot, x)\big)\mathrm{d}x\, n_\Omega, v\Big\rangle_{\partial\Omega} + o(\tau). \tag{2.16}$$ ■

Last, we will consider more complicated pairwise interactions of the form

$$\mathcal{F}_{f,g,h}(\Omega) = \int_\Omega f(x)\frac{G_x(\Omega)}{H_x(\Omega)}\mathrm{d}x, \quad \text{where} \quad \left\{ \begin{array}{l} G_x(\Omega) = \int_\Omega g(x,y)\mathrm{d}y, \\ H_x(\Omega) = \int_\Omega h(x,y)\mathrm{d}y. \end{array}\right. \tag{2.17}$$

The following proposition derives the expression of a level set gradient.

Proposition 3. *A level set gradient of* $\mathcal{F}_{f,g,h}$ *is*

$$\nabla\mathcal{F}_{f,g,h}(\varphi)(x) = \|\nabla\varphi(x)\| \left(f(x)\frac{G_x(\Omega)}{H_x(\Omega)} + \int_\Omega f(y)\frac{g(y,x)H_y(\Omega) - h(y,x)G_y(\Omega)}{H_y(\Omega)^2}\mathrm{d}y \right). \tag{2.18}$$

Proof. For a fixed x, let $A_x(\Omega) = G_x(\Omega)/H_x(\Omega)$. One has

$$\begin{aligned} A_x(\Omega_\tau) &= \frac{G_x(\Omega) + \tau\langle g(x,\cdot)n_\Omega,\, v\rangle_{\partial\Omega} + o(\tau)}{H_x(\Omega) + \tau\langle h(x,\cdot)n_\Omega,\, v\rangle_{\partial\Omega} + o(\tau)} \\ &= A_x(\Omega) + \tau\langle A_x'(\Omega)n_\Omega,\, v\rangle_{\partial\Omega} + o(\tau), \end{aligned} \tag{2.19}$$

where the shape gradient of A_x reads as

$$A_x'(\Omega) = \frac{g(x,\cdot)H_x(\Omega) - h(x,\cdot)G_x(\Omega)}{H_x(\Omega)^2}. \tag{2.20}$$

Integrating (2.19) multiplied by $f(x)$ on Ω_τ gives

$$\mathcal{F}_{f,g,h}(\Omega_\tau) = \int_{\Omega_\tau} \alpha(x)\mathrm{d}x + o(\tau), \tag{2.21}$$

where

$$\alpha(x) = f(x)\left(A_x(\Omega) + \tau\langle A_x'(\Omega)n_\Omega,\, v\rangle_{\partial\Omega}\right), \tag{2.22}$$

and hence the result (2.18) is obtained by expanding

$$\int_{\Omega_\tau} \alpha(x)\mathrm{d}x = \int_\Omega \alpha(x)\mathrm{d}x + \tau\langle \alpha n_\Omega,\, v\rangle_{\partial\Omega} + o(\tau). \tag{2.23}$$ ■

Let us mention that the shape gradient of an energy $\tilde{E}(\Omega) = E(\Omega^c)$, associated to the complementary domain $\Omega^c = [0,1]^2\backslash\Omega$, is easily derived as

$$\tilde{E}'(\Omega) = -E'(\Omega^c) \quad \Longrightarrow \quad \nabla\tilde{E}(\varphi) = -\nabla E(-\varphi) \tag{2.24}$$

since $-\varphi$ is a level set function associated to Ω^c.

In the following, we will make use of the identities (2.8), (2.10), (2.12), and (2.18) to derive the expression of $\nabla E(\varphi)$ for the different energies E considered.

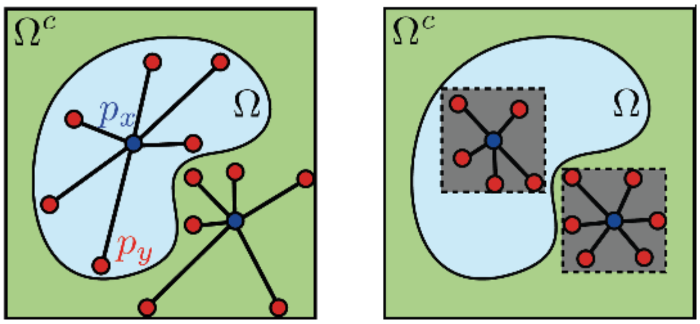


Figure 1. *Schematic display of the patch comparison principle using a large value of* σ *(left) and a smaller value of* σ *(right). Only patches* p_x *and* p_y *located jointly inside or outside* Ω *are compared. The dotted box in the right figure shows the set of locations* y *around* x *where* $G_\sigma(x,y)$ *is nonzero. Only pairs of patches located inside this area are compared.*

2.5. Numerical implementation. The segmentation is applied to a discretized image f of $n \times n$ pixels. The gradient and divergence operators appearing in (2.8) are computed using first order finite difference approximations. Integrals such as $\int_\Omega f(x)\mathrm{d}x$ are approximated by $\frac{1}{n^2}\sum_i f(i/n)$, where $i \in \{0,\dots,n-1\}^2$.

The gradient flow (2.6) is then discretized using a gradient descent

$$\varphi^{(\ell+1)} = \varphi^{(\ell)} - \xi_\ell \left(\nabla E(\varphi^{(\ell)}) + \gamma \nabla L(\varphi^{(\ell)})\right), \tag{2.25}$$

where $\xi_\ell > 0$ is a suitable time step size.

To ensure the stability of the level set evolution (2.6), one needs to reinitialize it from time to time. This corresponds to replacing φ by the signed distance function to the level set $\{x \setminus \varphi(x) = 0\}$.

3. Nonlocal active contours.

This section introduces two region-based energies that integrate pairwise interactions between pairs of patches inside and outside the region Ω. These energies constrain the local homogeneity of image features.

3.1. Unnormalized nonlocal active contours.

3.1.1. Pairwise patch interaction. A patch in some image f around a pixel $x \in [0,1]^2$ is defined as

$$p_x(t) = f(x+t) \quad \forall t \in [-\tau/2, \tau/2]^2. \tag{3.1}$$

The nonlocal interaction between two patches is measured using a metric $d(\cdot,\cdot) \geqslant 0$ that accounts for the similarity between patches. Typically $d(p_x, p_y)$ should be small for patches that are "similar," where the similarity is typically problem-dependent.

3.1.2. Pairwise interaction energy. The local homogeneity of the region (and of its complementary region) is measured by considering all possible pairwise patch interactions at a given scale $\sigma > 0$, as seen in Figure 1. This gives rise to the following nonlocal unnormalized (NL-U) energy of a region:

$$E^U_{\mathrm{NL}}(\Omega) = \bar{E}^U_{\mathrm{NL}}(\Omega) + \bar{E}^U_{\mathrm{NL}}(\Omega^c) \quad \text{with} \quad \bar{E}^U_{\mathrm{NL}}(\Omega) = \int_{\Omega\times\Omega} K(x,y)\mathrm{d}x\mathrm{d}y, \tag{3.2}$$

where $\Omega^c = [0,1]^2 \backslash \Omega$ is the complementary one of the region Ω, and K is the pairwise interaction kernel as

$$K(x,y) = G_\sigma(x,y) d(p_x, p_y). \tag{3.3}$$

The kernel $G_\sigma(x,y)$ is a decaying function of $\|x-y\|$. For the numerical examples, we use a truncated Gaussian function so that $G_\sigma(\cdot, y)$ has a bounded support inside a square of width ξ:

$$G_\sigma(x,y) = \begin{cases} e^{-\frac{\|x-y\|^2}{2\sigma^2}} & \text{if} \quad \|x-y\|_\infty \leqslant \xi, \\ 0 & \text{otherwise}. \end{cases} \tag{3.4}$$

Thus, the kernel G_σ depends on the values of σ and ξ. The parameter $\sigma > 0$ is important since it controls the scale of the local homogeneity one requires for the segmented object. If the region is made of a nearly constant pattern, one should use a large σ. In contrast, if the region exhibits fast feature variations, σ should be chosen smaller. For simplicity, we use the same scale for both inside and outside the region, but one could, of course, use two distinct parameters. This scale should also be adapted to the initial curve at time $t = 0$. If this initial curve is far away from the object boundary, a large windowing function might be required.

The level set gradient of this NL-U energy is computed as

$$\nabla E^U_{\mathrm{NL}}(\varphi) = \nabla \bar{E}^U_{\mathrm{NL}}(\varphi) - \nabla \bar{E}^U_{\mathrm{NL}}(-\varphi) \quad \text{with} \quad \nabla \bar{E}^U_{\mathrm{NL}}(\varphi) = \nabla \mathcal{F}_K(\varphi), \tag{3.5}$$

where $\nabla \mathcal{F}_K$ is defined in Proposition 2. Note that we have used the rule (2.24) to derive the gradient of the complementary region Ω^c.

3.1.3. Limitations. The nonlocal active contours model works well when the size of patches is small. Section 5.2 reports several examples of segmentation using pixel features (intensity or color values) and a weighted L^2 distance between patches. The local homogeneity property of the energy (3.2) enables this model to correctly detect objects which are only locally homogeneous, and can deal with separated objects with different features.

Section 5.3, however, shows that this unnormalized model suffers from a segmentation bias when the patch width τ is large. The segmented region is shifted away from the object boundary with an amount proportional to the patch width τ. This becomes problematic when used with large patches because of the lack of precision of the resulting segmentation. Large patches are, however, desirable as the noise level increases, since robustness requires more pixels to evaluate the local homogeneity.

3.2. Normalized nonlocal active contours. To reduce the segmentation bias introduced by the nonlocal active contour energy (3.2), we define a novel normalized nonlocal (NL-N) energy

$$E^N_{\mathrm{NL}}(\Omega) = \bar{E}^N_{\mathrm{NL}}(\Omega) + \bar{E}^N_{\mathrm{NL}}(\Omega^c), \tag{3.6}$$

where

$$\bar{E}^N_{\mathrm{NL}}(\Omega) = \int_\Omega \frac{\int_\Omega K(x,y)\mathrm{d}y}{\int_\Omega G_\sigma(x,y)\mathrm{d}y} \mathrm{d}x \tag{3.7}$$

with $K(x,y)$ defined in (3.3). Note that the unnormalized energy NL-U defined in (3.2) is recovered by setting the denominator $\int_\Omega G_\sigma(x,y)\mathrm{d}y$ to 1.

In practice, the correction factor $\int_\Omega G_\sigma(x,y)\mathrm{d}y$ is far from being constant, in particular when the size of patches is large. This normalization is thus crucial in reducing the disparities that increase as a pixel approaches the boundary of the segmented region.

The level set gradient of this energy is computed as

$$\nabla E^N_{\mathrm{NL}}(\varphi) = \nabla \bar{E}^N_{\mathrm{NL}}(\varphi) - \nabla \bar{E}^N_{\mathrm{NL}}(-\varphi), \tag{3.8}$$

where we use the shape gradient derivative (2.18) as follows:

$$\nabla \bar{E}^N_{\mathrm{NL}}(\varphi) = \nabla \mathcal{F}_{1,K,G_\sigma}(\varphi). \tag{3.9}$$

3.3. Summary of the algorithm. Figure 2 summarizes the main steps of our algorithm. The main computational load is the precomputation of the kernel $K(x,y)$. This complexity is somehow alleviated by restricting the computation to $\|x-y\|_\infty \leqslant \xi$.

Initialization:

1. Input data $f^0 : [0,1]^2 \to [0,1]^d$ ($d = 1$ or 3) and initial level set function $\varphi^{(0)}$.
2. Metric $d(\cdot,\cdot)$, for instance, L^2 distance (5.1), Wasserstein distances (7.3), (7.5), motion distance (8.1).
3. Parameters $\gamma, \tau, \sigma, \xi$ (see section 4.5).

Preprocessing:

1. Compute the feature map $f : [0,1]^2 \mapsto \mathbb{R}^d$ from the input f^0, for instance, $f = f^0$ for pixels (section 5), Gabor coefficients (6.1), or local distributions (7.1).
2. Compute the kernels $G_\sigma(x,y)$ (3.4) and $K(x,y)$ (3.3) for all $\|x-y\|_\infty \leqslant \xi$.

Minimization: Set $\ell = 0$.

1. Compute the gradient $\nabla E(\varphi^{(\ell)})$ using (3.5) for $E = E^U_{\mathrm{NL}}$ or (3.8) for $E = E^N_{\mathrm{NL}}$.
2. Compute $\nabla L(\varphi^{(\ell)})$ using (2.8) with $g = 1$.
3. Compute $\varphi^{(\ell+1)}$ using the gradient descent (2.25).
4. Set $\ell \leftarrow \ell + 1$ and go back to 1 until convergence.

Figure 2. *Nonlocal active contour algorithm.*

4. Other methods. In the following sections, we compare our nonlocal active contour model with some previously proposed segmentation methods. To perform a fair comparison with our method, we extend these techniques (when this is possible) to handle patches with a comparison function $d(\cdot,\cdot)$.

4.1. Chan–Vese active contours. The Chan–Vese (CV) model [13] assumes that the inside Ω and outside Ω^c of the region to be segmented are globally homogeneous by measuring the average variation with respect to template patches p_1, p_2:

$$E_{\mathrm{CV}}(\Omega, p_1, p_2) = \bar{E}_{\mathrm{CV}}(\Omega, p_1) + \bar{E}_{\mathrm{CV}}(\Omega^c, p_2), \quad \text{where} \quad \bar{E}_{\mathrm{CV}}(\Omega, p) = \int_\Omega d(p_x, p)\mathrm{d}x. \tag{4.1}$$

Here we incorporated the CV model with a general distance for the fair comparison with our models.

The level set gradient of this energy reads as

$$\nabla E_{\mathrm{CV}}(\varphi, p_1, p_2) = \nabla \mathcal{E}_{g_1}(\varphi) - \nabla \mathcal{E}_{g_2}(-\varphi), \quad \text{where} \quad g_i(x) = d(p_x, p_i),$$

and $\nabla \mathcal{E}_g$ is defined in (2.10) (here p_1 and p_2 are assumed to be fixed). Following [13], the active contour evolution is obtained by performing a gradient descent step (2.25) with $\nabla E(\varphi^{(\ell)}) = \nabla E_{\mathrm{CV}}(\varphi^{(\ell)}, p_1, p_2)$. After each gradient step, the value of the templates p_1, p_2 is updated as

$$\begin{cases} p_1 = \Gamma_1(\Omega), \\ p_2 = \Gamma_1(\Omega^c), \end{cases} \quad \text{where} \quad \Gamma_U(\Omega) = \underset{p}{\operatorname{argmin}} \int_\Omega U(x) d(p_x, p) \mathrm{d}x, \tag{4.2}$$

with $U(x) = 1$ and $\Omega = \left\{x \setminus \varphi^{(\ell)}(x) > 0\right\}$. We detail in the following section (see (5.2) and (7.6)) how to compute Γ_1 for the special case of the L^2 and 1-D Wasserstein metrics $d(\cdot, \cdot)$.

4.2. Geodesic active contours (GAC). The geodesic active contour [11] looks for a curve that has a minimum length according to a metric $g(x) \geqslant 0$:

$$E_{\mathrm{GAC}}(\Omega) = \int_{\partial\Omega} g(x) \mathrm{d}x. \tag{4.3}$$

The metric g is usually defined using an edge detector. In the numerical experiments, it is computed by rescaling a gradient-based detector $\tilde{g}$ to $[0, 1]$ using an affine transform. We used a modified version of the original edge function of the form (6.2) as in [11], which provides better (clearer) edges than using (6.2),

$$\tilde{g}(x) = \frac{1}{\delta + G_b \star \|\nabla f(x)\|^q}, \tag{4.4}$$

where $\star$ is the spatial convolution with Gaussian filters $G_b(x) = \frac{1}{2\pi b^2} e^{-\|x\|^2/2b^2}$. We fix in the numerical examples $\delta = 0.1$, $q = 1$, and adapt $b > 0$ for each tested image.

The active contour evolution is derived by performing a gradient descent of E_{GAC}, with an additional balloon force [14]

$$\frac{\partial \varphi}{\partial t} = -\nabla E_{\mathrm{GAC}}(\varphi) + \eta g \|\nabla \varphi\|, \tag{4.5}$$

where $\nabla E_{\mathrm{GAC}} = \nabla L_g$ as defined in (2.8) and $\eta g \|\nabla \varphi\|$ helps to avoid poor local minima by forcing the curve to move forward/outward (depending on the sign of η).

4.3. Integrated active contours (IAC). The IAC, introduced by [45], is a weighted average of the GAC (4.3) and CV (4.1) energies incorporated with patches:

$$E_{\mathrm{IAC}}(\varphi, p_1, p_2) = E_{\mathrm{CV}}(\varphi, p_1, p_2) + \mu E_{\mathrm{GAC}}(\varphi). \tag{4.6}$$

The active contour evolution is computed as described in section 4.1 for the CV model.

4.4. Locally binary fitting (LBF). The CV region-based model (4.1) imposes a global homogeneity of the regions to be segmented. To cope with spatially varying regions, the LBF model [34] makes the templates $p_1(x)$ and $p_2(x)$ of the energy vary spatially:

$$\begin{aligned} E_{\text{LBF}}(\Omega, p_1, p_2) = &\int_\Omega \int_{[0,1]^2} G_\sigma(x,y) d(p_x, p_1(y)) \mathrm{d}y\mathrm{d}x \\ &+ \int_{\Omega^c} \int_{[0,1]^2} G_\sigma(x,y) d(p_x, p_2(y)) \mathrm{d}y\mathrm{d}x. \end{aligned} \tag{4.7}$$

Here we also incorporated the LBF model with a general distance, for the same reason as we did for the CV model.

The active contour evolution is computed as described in section 4.1 for the CV model, where the level set of the LBF energy (assuming p_1 and p_2 fixed) reads as

$$\nabla E_{\text{LBF}}(\varphi, p_1, p_2) = \nabla \mathcal{E}_{g_1}(\varphi) - \nabla \mathcal{E}_{g_2}(-\varphi), \tag{4.8}$$

where

$$g_i(x) = \int_{[0,1]^2} G_\sigma(x,y) d(p_x, p_i(y)) \mathrm{d}y \tag{4.9}$$

and $\nabla \mathcal{E}_g$ is defined in (2.10). The values of p_1, p_2 are updated at iterations ℓ defined in (2.25),

$$\left\{ \begin{array}{l} p_1(y) = \Gamma_{G_\sigma}(\Omega), \\ p_2(y) = \Gamma_{G_\sigma}(\Omega^c), \end{array} \right. \quad \text{where} \quad \Gamma_{G_\sigma}(\Omega) = \underset{p}{\operatorname{argmin}} \int_\Omega G_\sigma(x,y) d(p_x, p) \mathrm{d}x, \tag{4.10}$$

as defined in (4.2), and $\Omega = \left\{ x \setminus \varphi^{(\ell)}(x) > 0 \right\}$.

4.5. Parameters of the methods. Our nonlocal method as well as the competing methods require setting up several parameters that can have a significant impact on the segmentation results. To ensure the reproducibility of the comparisons reported in the following numerical section, we provide the list of parameters for each experiment in each figure's captions. Furthermore, the implementation of the methods can be retrieved online.[1]

Here are the parameters of the different methods:

- n: width of the image.
- γ: weight of the curve regularization (see (2.1)).
- τ: patch width (see (3.1)).
- (σ, ξ): scale of the patch comparison function (see (3.3) and (4.7)), only for NL-U, NL-N, and LBF.
- b: smoothing bandwidth of the edge detector (see (4.4), (6.2)), only for GAC and IAC.
- η: amplitude of the balloon force (see (4.5)), only for GAC.
- μ: weight between edge and region terms (see (4.6)), only for IAC.
- λ: weight for the energy preventing the domains from overlapping (see (9.5)), only for the multiphase repulsive (MR) method.

In the next four sections, we apply all of the formalism introduced so far to different segmentation problems using different kinds of features and metrics.

---

[1] See https://sites.google.com/site/miyounjungr/publications.

5. Pixel-based segmentation.

5.1. Weighted L^2 patch distance. The simplest features are simply the pixel values $f(x) \in \mathbb{R}^d$ of the input image, where $d = 1$ for grayscale images, and $d = 3$ for color images. In this section we use a weighted L^2 distance (5.1), similar to the patch distance used for nonlocal denoising,

$$d(p_x, p_y) = \int_t G_a(t) \|p_x(t) - p_y(t)\|^2 \mathrm{d}t. \tag{5.1}$$

The Gaussian weight $G_a(t) = e^{-\frac{\|t\|^2}{2a^2}}$ is used to give more influence to the central pixel.

To implement the CV and LBF models described in sections 4.1 and 4.4, the update operator defined in (4.2) and (4.10) reduces to a linear averaging

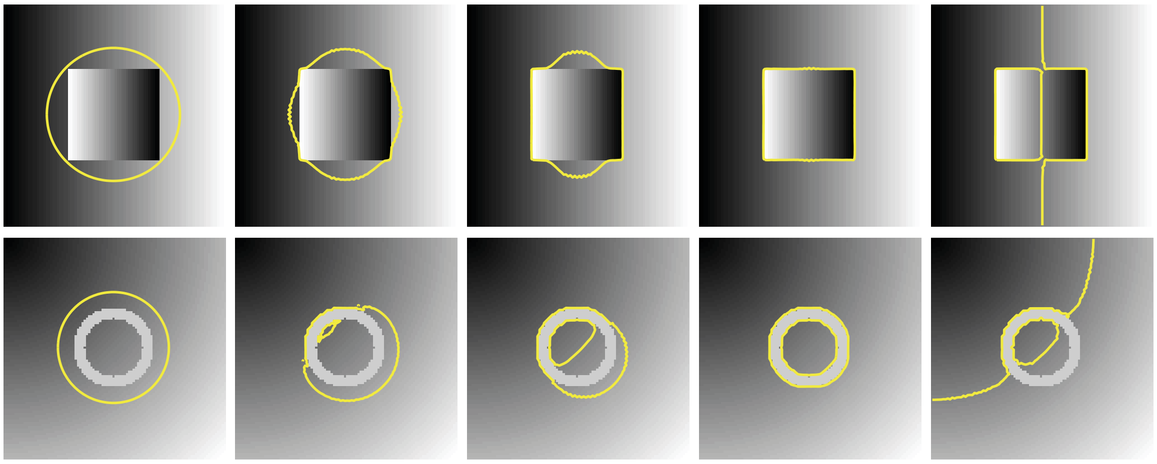
$$\Gamma_U(y) = \frac{\int_\Omega U(x) p_x \mathrm{d}x}{\int_\Omega U(x) \mathrm{d}x}. \tag{5.2}$$

5.2. Examples with unnormalized model (NL-U). This section presents results of the L^2 distance (5.1) with intensity and color features. In all the examples, we use the small size of patches of width $\tau = 3/n$ for a discretized image of size $n \times n$ and $a = 0.5/n$.

In Figures 3 and 4, we test our method on several synthetic images with spatially varying background and/or object, or with several separated objects with different intensities. In all the examples, our model correctly detects the objects due to the local homogeneity property, in contrast to the two-phase CV model [13] requiring a global homogeneity in each region. The first example in Figure 3 demonstrates well the effect of this property, and the second example shows, in addition, the detection of an interior contour. The bottom object in the first example in Figure 4 has spatially varying intensities, and, moreover, the intensities of its left side are close to those of the background. Thus, the CV model fails to segment this piecewise smooth object, regarding its left side as background, as shown in Figure 5, while our model captures the boundary with small gradients. Furthermore, Figure 4 shows detection of multiple separated objects with different intensities.

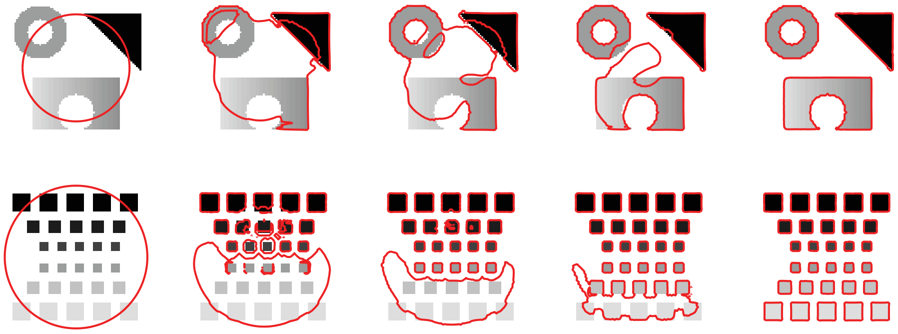
Figure 5 shows segmentation results on the same image with the methods described in section 4: CV, GAC, and IAC. For the IAC model, two final curves are shown with two different but close parameters μ ($\mu_1 > \mu_2$). Because μ is a balancing term between the region-based and edge-based energies, when $\mu > \mu_1$ (or $\mu < \mu_2$), the model tends to act like the geodesic snake model (or CV model). Thus, with the given initial curves, all the models fail to detect the correct object boundaries. Note that, with good initial curves surrounding all the boundaries, the IAC model would be able to detect the boundaries with large values of μ, but in our model one circle around the objects as an initial curve is enough to achieve a correct segmentation. Also, with the same initial curves used in Figure 5, our NL-U model gives the same resulting curves as in Figure 4. Thus, our model is less sensitive to the choice of initial curves than edge-based active contour models.

Figure 6 presents segmentation results with our NL-U method on grayscale images. We compare our model with the CV and IAC models. Our model detects objects in the image with spatially varying background and multiple objects with different intensities, while both the CV and IAC models fail to segment objects correctly.



$\ell = 0$ $\ell = 10$ $\ell = 20$ $\ell = 150$ CV

Figure 3. *Curve evolution for the segmentation of two images with spatially varying background and object using NL-U and comparison with CV. ℓ is the iteration number. Parameters: $n = 100$, $(\tau, a) = (3/n, 0.5/n)$, $(\sigma, \xi) = (10/n, 31/n)$, $\gamma = 15/(n^5\xi^2\tau^2)$ (NL-U), $\gamma = 0.02$ (CV).*



$\ell = 0$ $\ell = 10$ $\ell = 20$ $\ell = 50$ final curve

Figure 4. *Curve evolution for the detection of objects with spatially varying object, or with several separated objects with different intensities, using NL-U model. ℓ is the iteration number. Parameters: $n = 100$, $(\tau, a) = (3/n, 0.5/n)$, $(\sigma, \xi) = (10/n, 31/n)$, $\gamma = 15/(n^5\xi^2\tau^2)$.*

Figure 7 shows segmentation results with our NL-U method on natural color images. We compare our model with the vector-valued CV model [12] and the IAC model. By using an initial curve near the boundary of the object(s) and a small windowing function, our model detects the boundary of nonhomogeneous object(s). The segmentation result is fairly good compared with CV and IAC models that capture only part of the object(s). On the other hand, these examples also show a limitation of our model: in order to detect the boundary of nonhomogeneous objects, the initial curve needs to be located near the object boundary so that a small windowing function can be used.

5.3. Comparison of unnormalized and normalized models. The unnormalized energy (3.2) works well with a small size of patches or pixel values, as seen in Figures 3–7. However,

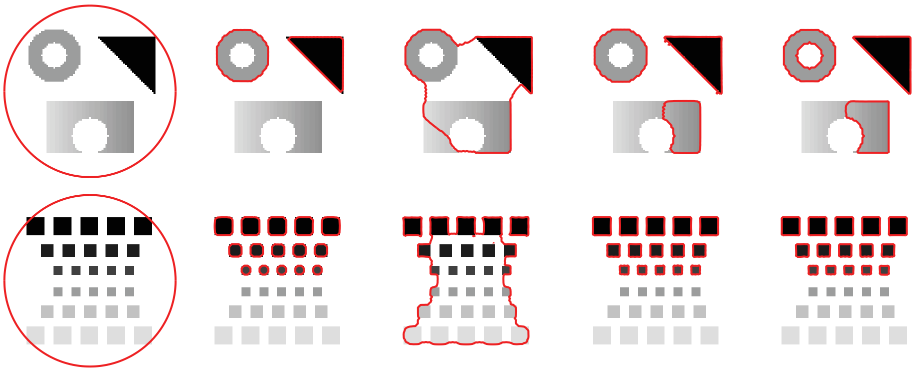

Initial curve GAC IAC with $\mu = \mu_1$ IAC with $\mu = \mu_2$ CV

Figure 5. *Comparison of CV, GAC, and IAC (different* μ *values). Parameters:* $b = 0.5/n$, $\eta = -0.3/n$ *(GAC),* $(\mu_1, \mu_2) = (3.6/n, 3.5/n)$ *(top),* $(1/n, 0.9/n)$ *(bottom),* $\gamma = 0$ *(IAC),* $\gamma = 0.02/n$ *(CV).*

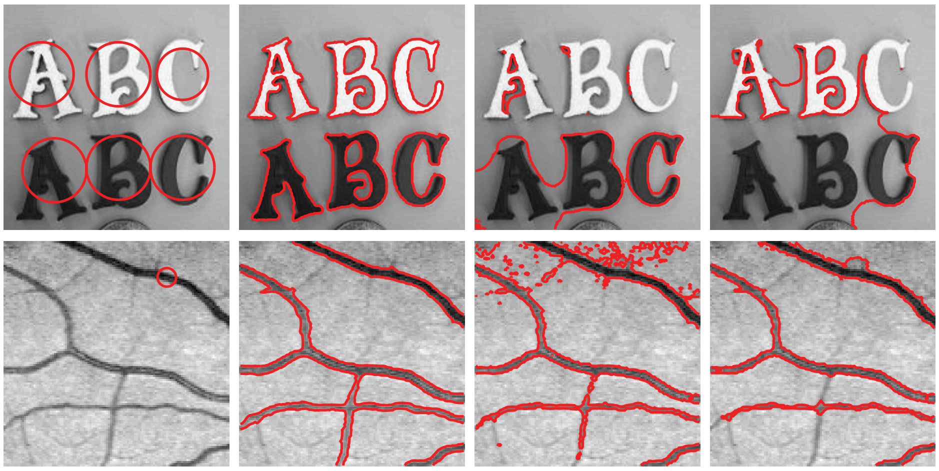

Initial curve NL-U CV IAC

Figure 6. *Grayscale image segmentation. Final curves of our unnormalized model NL-U, CV, and IAC. Parameters:* $n = 176, 121$, $(\tau, a) = (3/n, 0.5/n)$*; top:* $\gamma = 10/(n^5\xi^2\tau^2)$, $(\sigma, \xi) = (10/n, 31/n)$ *(NL-U),* $\gamma = 0.2/n$ *(CV),* $b = 0.5/n$, $\mu = 0.3/n$, $\gamma = 0$ *(IAC); bottom:* $\gamma = 5/(n^5\xi^2\tau^2)$, $(\sigma, \xi) = (10/n, 21/n)$ *(NL-U),* $\gamma = 0.005/n$ *(CV),* $b = 0.5/n$, $\mu = 0.5/n$, $\gamma = 0$ *(IAC).*

this energy has a segmentation bias near the object boundary when using a large size of patches, even though contour is attracted to the boundary. Despite this limitation of the energy (3.2), we keep it as our model due to its simplicity as well as the fact that there are many cases where the use of a small size of patches or pixel values is enough to segment regions.

Figure 8 presents examples that require a large size of patches and compares the results of our unnormalized (3.2) and normalized (3.6) energies. Figure 8 makes use of the L^2 distance (5.1) with intensity feature. In this example, the unnormalized model does not provide satisfactory results with any kind of patches: patches of width $\tau = 1/n$ (1 pixel), $\tau = 3/n$ (3×3 pixels) with $a = \infty$. By using patches of width $\tau = 1/n$, the unnormalized model

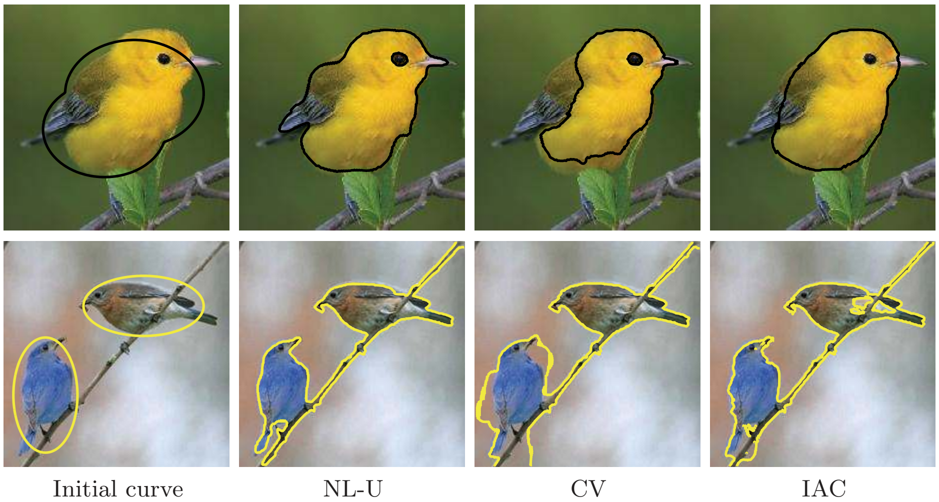


Figure 7. *Natural color image segmentation. Final curves of our unnormalized model NL-U, vector-valued CV* [12]*, and IAC. Parameters:* $n = 189, 225$*,* $(\tau, a) = (3/n, 0.5/n)$*,* $(\sigma, \xi) = (10/n, 41/n)$*; top:* $\gamma = 100/(n^5\xi^2\tau^2)$ *(NL-U),* $\gamma = 0.5/n$ *(CV),* $b = 0.5/n$*,* $\mu = 5/n$*,* $\gamma = 0$ *(IAC); bottom:* $\gamma = 60/(n^5\xi^2\tau^2)$*, (NL-U),* $\gamma = 0.3/n$ *(CV),* $b = 0.5/n$*,* $\mu = 1/n$*,* $\gamma = 0$ *(IAC).*

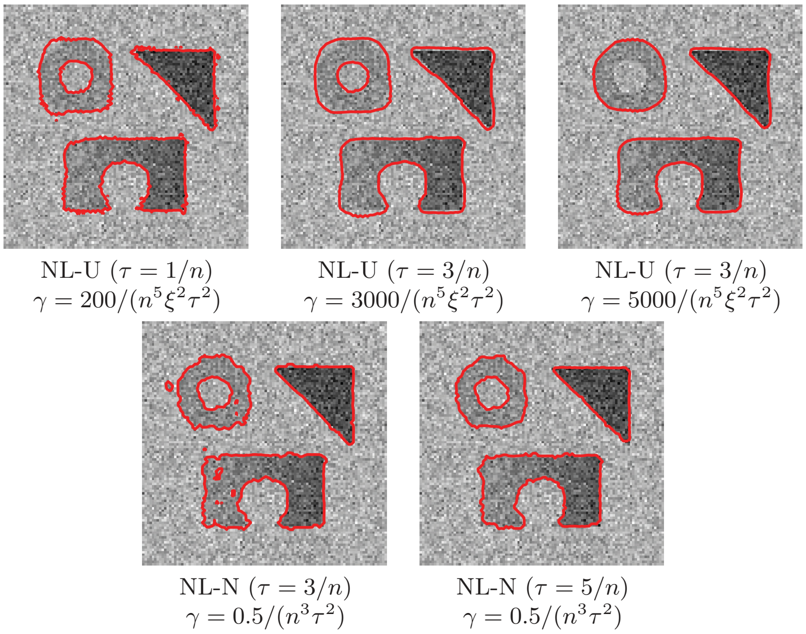

NL-U ($\tau = 1/n$) $\gamma = 200/(n^5\xi^2\tau^2)$
NL-U ($\tau = 3/n$) $\gamma = 3000/(n^5\xi^2\tau^2)$
NL-U ($\tau = 3/n$) $\gamma = 5000/(n^5\xi^2\tau^2)$
NL-N ($\tau = 3/n$) $\gamma = 0.5/(n^3\tau^2)$
NL-N ($\tau = 5/n$) $\gamma = 0.5/(n^3\tau^2)$

Figure 8. *Comparison of our unnormalized* (3.2) *and normalized* (3.6) *models using the* L^2 *distance* (5.1)*. Parameters:* $n = 100$*,* τ *variable,* $a = \infty$*,* $(\sigma, \xi) = (\infty, 31/n)$*,* γ *variable.*

produces noisy final curves, and by using patches of width $\tau = 3/n$, it results in smoother final curves that are, however, not located on the object boundaries in spite of adjusting the smoothness parameter γ. On the other hand, the normalized model with patches of width

$\tau = 5/n$ provides a smooth final curve, located exactly on the boundary.

Section 7.4 shows other comparisons of NL-U and NL-N in the case of local statistical features. The conclusion remains the same, namely that the normalized model helps to reduce the segmentation bias.

6. Locally oriented texture segmentation.

A popular set of features for discriminating the local orientation of textures are computed from the responses to a set of Gabor filters; see, for instance, [47, 45].

6.1. Gabor features.

Given an input image $f^0 : [0,1]^2 \to \mathbb{R}$, $f(x) \in \mathbb{R}^d$ is defined as the absolute value of the filtering of f^0 with d complex filters:

$$f_\nu(x) = |f^0 \star h_\nu| \quad \forall\, \nu \in \{0, \ldots, d-1\} \tag{6.1}$$

with $x = (x_1, x_2) \in [0,1]^2$ and

$$h_\nu(x) = e^{\frac{2i\pi}{n}\eta_\nu(\cos(\theta_\nu)x_1 + \sin(\theta_\nu)x_2)} G_{s_\nu}(x).$$

The parameter $\eta_\nu > 0$ is the frequency of the filtering, $\theta_\nu \in [0, \pi)$ is the orientation, and $s_\nu > 0$ is the spatial width of the filter. In the numerical examples, the parameters η_ν, θ_ν, s_ν are adapted to obtain the best texture segmentation, and f_ν are rescaled to $[0,1]$ using an affine transform.

The resulting feature map $f(x)$ is a multichannel image, and the patches p_x are compared using the weighted L^2 norm (5.1).

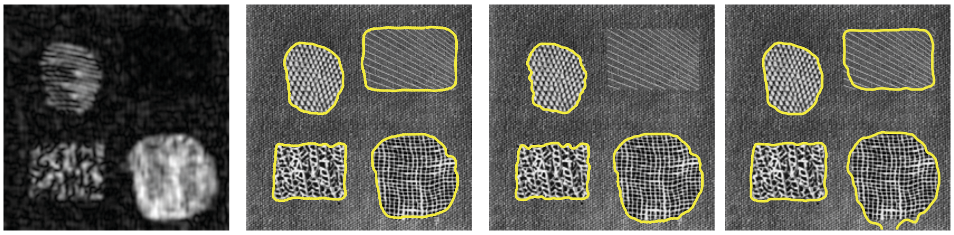
6.2. Examples.

Figure 9 shows a comparison between the NL-U and the CV models applied to Gabor coefficients [47]. In this case, we use $d = 8$ filters with $\eta_\nu \in \{2, 2.5, 3, 3.5\}$, $\theta_\nu \in \{0, \pi/2\}$, $s_\nu = 2$. We have also tested the CV method with a larger number $d = 64$ of features with $\eta_\nu \in \{2, 3, 4, 5\}$, $\theta_\nu \in \{0, \pi/4, \pi/2, 3\pi/4\}$, $s_\nu \in \{2, 2\sqrt{2}, 4, 4\sqrt{2}\}$. The Gabor-based CV method fails to detect the object on the top right side because the intensity values of that object in the Gabor domain are small and comparable with the coefficients of the background. Using a larger number of features helps but is not enough. In contrast, our model correctly detects all the objects thanks to the local homogeneity criterion.

Figure 10 shows an example where both the object and the background are composed of textures with smoothly varying orientations, simulating perspective. We use $d = 4$ filters with $\eta_\nu \in \{0.7, 1.6\}$, $\theta_\nu = 0$, $s_\nu \in \{4, 4\sqrt{2}\}$ in the first example, and $d = 8$ filters with $\eta_\nu \in \{2, 3, 4, 5\}$, $\theta_\nu = 0$, $s_\nu \in \{4, 4\sqrt{2}\}$ in the second one. Similarly to Figure 3, the Gabor based CV model [47] fails to segment the object, while our model does a correct segmentation. For the IAC model, we use another rescaled edge function $\tilde{g}(x)$ as in [11],

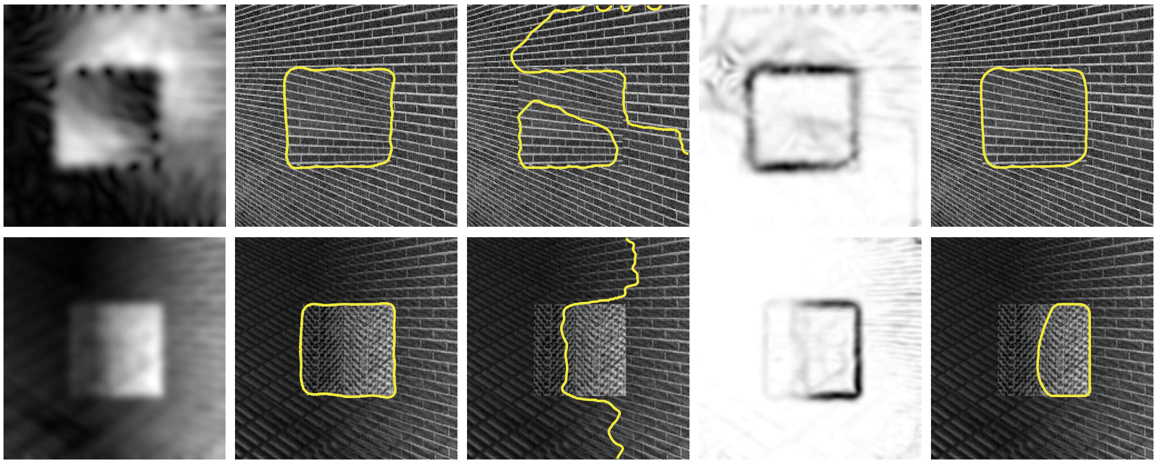
$$\tilde{g}(x) = \frac{1}{\delta + \|\nabla(G_b \star f)(x)\|^q}, \tag{6.2}$$

with fixed $\delta = 0.1$ and $q = 2$. The IAC model detects the object in the first example but not in the second, which shows the difficulty of designing a good edge detector $g(x)$ for texture features. In contrast, our model can segment the object in both cases without the need for any edge detector.



A feature f_ν NL-U CV with $d = 8$ (left) and 64 (right)

Figure 9. *Texture segmentation with Gabor features. Parameters:* $n = 202$, $(\tau, a) = (3/n, 0.5/n)$, $(\sigma, \xi) = (10/n, 31/n)$, $\gamma = 40/(n^5\xi^2\tau^2)$ *(NL-U),* $\gamma = 0.5/n$ *(CV).*



A feature f_ν NL-U CV Edge $\tilde{g}$ (6.2) IAC

Figure 10. *Texture segmentation of smoothly varying textures with Gabor features. Parameters:* $n = 200, 196$, $(\tau, a) = (3/n, 0.5/n)$, $(\sigma, \xi) = (10/n, 41/n)$, $\gamma = 100/(n^5\xi^2\tau^2)$ *(NL-U),* $\gamma = 0.2/n$ *(CV),* $b = 3.75/n$ *(top),* $b = 0.5/n$ *(bottom),* $\mu = 20/n$, $\gamma = 0$ *(IAC).*

7. Statistical segmentation.

While Gabor features described in section 6 are useful in dealing with locally oriented geometrical textures, textures with more randomness require the use of statistical features. We propose in this section a simple framework that estimates the local first order statistics of the pixel distribution. A key idea is to use an optimal transport distance to compare the local statistics, which is important in dealing with arbitrary pixel statistics. Note that more advanced statistical features could be used (e.g., distribution of pairs of pixels), but this is outside the scope of this paper.

7.1. Local statistical features.

Our statistical features are discrete densities in $\mathbb{R}^d$ made of P Dirac distributions:

$$\mu_X = \frac{1}{P}\sum_{i=0}^{P-1} \delta_{X_i}, \quad \text{where} \quad X_i \in \mathbb{R}^d. \tag{7.1}$$

Here $d = 1$ for a grayscale image, and $d = 3$ for a color image. We simply estimate the pixel value density around a pixel x as the empirical distribution μ_{p_x} where p_x is a patch of $P = \tau^2$

pixels extracted around x.

Note that it is possible to use a more advanced estimation procedure such as a Parzen window method. The estimation μ_{p_x} using a sum of Dirac has the advantage of simplicity and leads to statistical distances that can be computed very quickly.

The L^2 Wasserstein distance between two distributions μ_X, μ_Y in $\mathbb{R}^d$ is defined as

$$W(\mu_X, \mu_Y)^2 = \min_{\sigma \in \Sigma_P} \sum_{i=0}^{P-1} \|X_i - Y_{\sigma(i)}\|^2, \tag{7.2}$$

where Σ_P is the set of all the permutations of P elements. For simplicity we have restricted our attention to distributions having the same number P of points, which is the case for our application to segmentation. Note also that our method can handle arbitrary L^q Wasserstein metrics for $q \geqslant 1$, but we use only the L^2 in the numerical examples.

The permutation σ minimizing (7.2) is the optimal assignment between the two point clouds. This optimal assignment problem can be solved using combinatorial optimization schemes in $O(P^{5/2}\log(P))$ operations when $d > 1$; see [8].

In the 1-D case, the optimal assignment σ that solves (7.2) can be computed in $O(P\log(P))$ operations by ordering the point clouds X and Y as

$$X_{\sigma_X(i)} \leqslant X_{\sigma_X(i+1)} \quad \text{and} \quad Y_{\sigma_Y(i)} \leqslant Y_{\sigma_Y(i+1)}$$

with two permutations $\sigma_X, \sigma_Y \in \Sigma_N$. The Wasserstein distance is then the L^2 norm of the sorted vectors:

$$W(\mu_X, \mu_Y)^2 = \sum_{i=0}^{N-1} |X_{\sigma_X(i)} - Y_{\sigma_Y(i)}|^2. \tag{7.3}$$

Note the major computational difference between the assignment problem (7.2) in dimension $d = 1$ and in higher dimensions $d > 1$, where no $O(P\log(P))$ algorithm is available.

7.2. Sliced Wasserstein distance. The numerical complexity of solving (7.2) in dimension $d > 1$ is prohibitive for imaging applications such as our segmentation problem. To obtain a fast numerical scheme, we follow the work of Rabin et al. [44] that introduces a sliced Wasserstein distance. It is defined as a sum of 1-D Wasserstein distances of projected distributions:

$$SW(\mu_X, \mu_Y)^2 = \sum_{\theta \in \Theta} W(\mu_{X_\theta}, \mu_{Y_\theta})^2, \quad \text{where} \quad X_\theta = \{\langle X_i,\, \theta\rangle\}_{i=1}^{P-1}. \tag{7.4}$$

Here $X_\theta, Y_\theta \subset \mathbb{R}$ are projected 1-D distributions, and $\Theta \subset \mathbb{R}^d$ is a discrete set of directions, sampled on the unit sphere (i.e., $\theta \in \mathbb{R}^d$ with $\|\theta\| = 1$).

Evaluating this sliced distance (7.4) has a complexity of $O(|\Theta|P\log(P))$ operations which is advantageous over the original Wasserstein distance (7.2) if Θ is not too large. Although there is no mathematical proof of the quality of the approximation of W using SW, numerical observations suggest that SW is a good approximation for solving minimization problems involving the Wasserstein metric; see [44]. Note that other approximations of the Wasserstein distance have been proposed, in particular [50], but we found that the sliced approximation (7.4) was precise enough for color image segmentation.

The sliced approximation (7.4) is used to measure the similarity between patches to perform statistical region-based segmentation. We thus define a new similarity measure between patches as

$$d(p_x, p_y) = SW(\mu_{p_x}, \mu_{p_y})^2. \tag{7.5}$$

7.3. Wasserstein barycenter. It is possible to compute in close form the CV and the LBF update operators (4.2) and (4.10) in the case of 1-D features ($d = 1$). In this case, it corresponds to an averaging of the sorted values:

$$\Gamma_U(\Omega) = p, p(i) = \frac{\int_\Omega U(x) p_x(\sigma_x(i)) \mathrm{d}x}{\int_\Omega U(x) \mathrm{d}x}, \tag{7.6}$$

where $\sigma_x \in \Sigma_P$ is a permutation such that the values $(p_x(\sigma_x(i)))_i$ are sorted in increasing order.

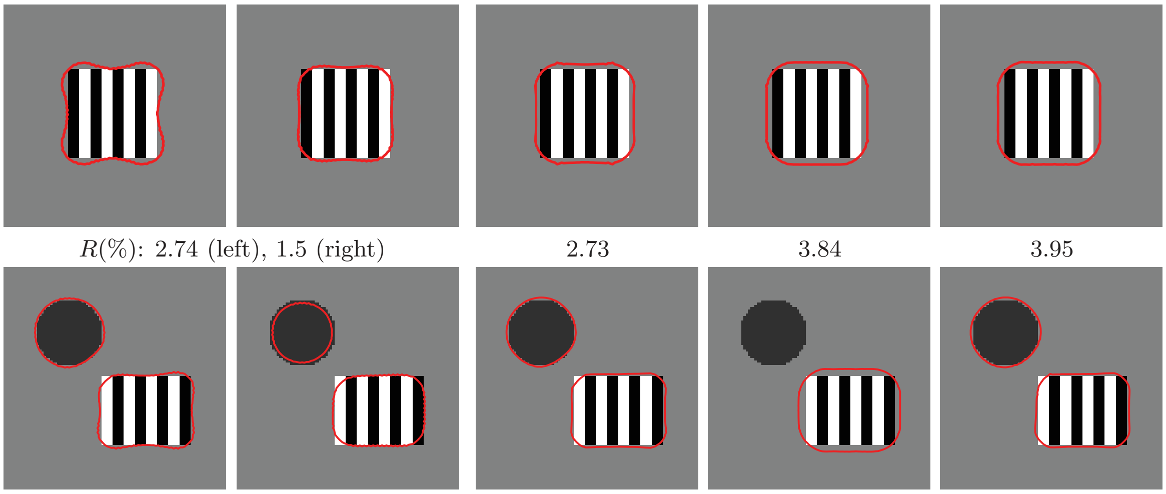
There is no closed form expression for $\Gamma_U(\Omega)$ in the general case $d > 1$, although an approximation algorithm has been introduced in [44]. It is, however, beyond the scope of this paper to extend LBF and CV to the case of multidimensional distributions.

7.4. Examples. In the numerical examples, we use a L^2 sliced Wasserstein distance (which corresponds to $q = 2$).

Figure 11 presents simple examples where the L^2 patch distance (5.1) cannot be applied because the black and white stripe pattern is a texture that is not homogeneous in the pixel domain. Furthermore, these examples require a large patch size to capture the texture statistics. This figure presents comparisons of the unnormalized model (3.2), the normalized model (3.6), and the CV (4.1) and LBF models (4.7), using the 1-D Wasserstein distance (7.3), and also presents the mismatching rate ($R\%$) between a resulting curve and the ideal curve exactly located on the object boundary.

As we already mentioned in section 5.3, the unnormalized model introduces a segmentation bias near the object boundary when using a large size of patches. This bias cannot be reduced completely by adjusting the parameter γ. The normalized model (3.6), in column 3, reduces the bias to some extent, independently of the value of γ. The LBF model seems to provide results similar to those of our normalized model, but it results in a more biased curve than our model in the first example using a large size of patches ($\tau = 15/n$). Column 4 shows that the CV model also suffers from a segmentation bias: the curves are located a few pixels away from the boundaries. Although our models and the CV/LBF models have similar behaviors on globally homogeneous textures, this example highlights the importance of our normalization.

Figure 12 shows more complicated examples of grayscale ($d = 1$) synthetic texture segmentation. This shows that our normalized model (3.6) with the Wasserstein distance (7.3) detects objects with smoothly varying distributions of intensities and separated multiple objects with different distributions of intensities. It also shows the curve evolution of our model, starting from given initial curves, and displays the convergence of the energy $E(\varphi^{(\ell)})$ as a function of the iteration index ℓ.



$R(\%)$: 2.74 (left), 1.5 (right) | 2.73 | 3.84 | 3.95

$R(\%)$: 1.89 (left), 2.44 (right) | 1.91 | 10.1 | 1.97

NL-U (top: $\gamma' = 10^4$, $\cdot 10^5$) (bottom: $\gamma' = 2 \cdot 10^4$, 10^5) | NL-N ($\gamma = 5/(n^3\tau^2)$) | CV ($\gamma = 5/(n^3\tau^2)$) | LBF ($\gamma' = 2000$)

Figure 11. *Comparison of models with the Wasserstein distance function* (7.3). *$R(\%)$ indicates the mismatching rate between a resulting curve and the ideal curve exactly located on the object boundary. Parameters: $n = 100$, $\tau = 15/n$ (top), $\tau = 11/n$ (bottom), $(\sigma, \xi) = (\infty, 31/n)$, γ variable (NL-U, LBF: $\gamma = \gamma'/(n^5\xi^2\tau^2)$).*

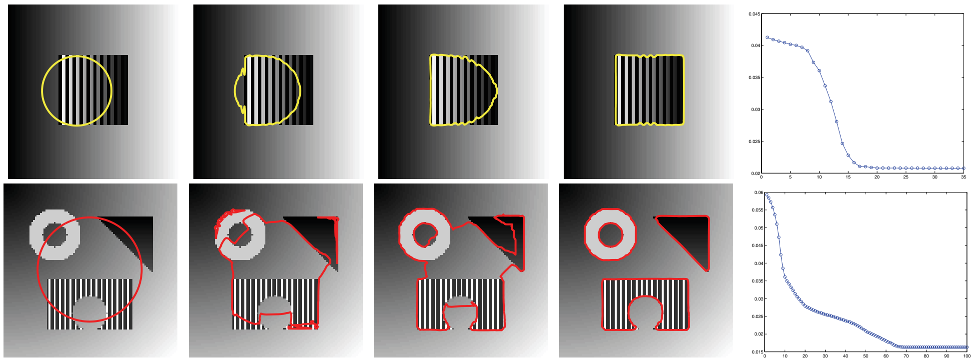
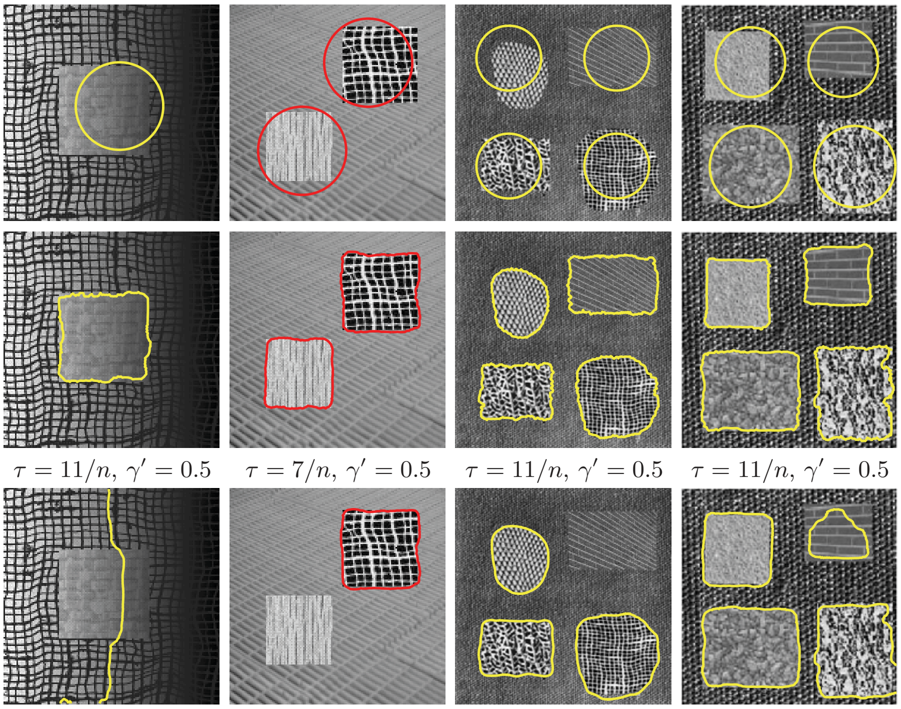


Figure 12. *Segmentation results with Wasserstein distance* (7.3). *Left: curve evolution using our NL-N model* (3.6). *Right: plot of energy $\mathcal{E}(\varphi^{(\ell)})$ vs iteration ℓ. Parameters: $n = 100$, $\tau = 3/n$, $(\sigma, \xi) = (\infty, 31/n)$, $\gamma = 0.5/(n^3\tau^2)$.*

Figure 13 presents texture segmentation results of our model (3.6) and a comparison with the CV model (4.1). Again due to the local homogeneity, our model discriminates different textures having different distributions of intensities, while CV fails for the correct discrimination.

Figure 14 shows examples of color texture segmentation using the sliced Wasserstein distance (7.4). We considered only $|\Theta| = 3$ projection directions, i.e., $\Theta = \{(1, 0, 0), (0, 1, 0), (0, 0, 1)\}$, which was enough to obtain satisfactory segmentations in all the given examples. We also compare our model with the vector-valued CV model [12] and with the color exten-



$\tau = 11/n,\ \gamma' = 0.5$ $\tau = 7/n,\ \gamma' = 0.5$ $\tau = 11/n,\ \gamma' = 0.5$ $\tau = 11/n,\ \gamma' = 0.5$

$\tau = 11/n,\ \gamma' = 5$ $\tau = 7/n,\ \gamma' = 1$ $\tau = 11/n,\ \gamma' = 5$ $\tau = 11/n,\ \gamma' = 5$

Figure 13. *Segmentation results with Wasserstein distance* (7.3). *Top row: initial curves. Middle: results of our NL-N model* (3.6). *Bottom: results of the CV model extended with Wasserstein distance. Parameters:* $n = 148, 147, 202, 186$ *(left to right),* τ *variable,* $(\sigma, \xi) = (\infty, 31/n)$, $\gamma = \gamma'/(n^3\tau^2)$ *variable.*

sion [2] of the original method proposed in [38]. In all the examples, our model correctly detects the boundary of objects and segments separated multiple objects with different distributions of color values, in contrast to the other models, which do not locate the curve on the exact boundaries, detect only part of objects, or fail to detect objects. Note that the second example was degraded by the random-valued impulsive noise (noise with any intensity value is randomly distributed over the entire image) of density 0.3 (probability of the corrupted image pixels).

8. Motion segmentation.

To perform motion segmentation, we use the motion profile features introduced by Shi and Malik in [48, 49] within our nonlocal active contour framework.

8.1. Motion features.

The goal is to segment moving objects between two consecutive frames f^0, f^1 of a video sequence. The local movement at a pixel x is represented by a motion profile $f(x) \in \mathbb{R}^d$. In the following we use a patch of size $\tau \times \tau$ so that $f(x) = p_x$ is a motion profile.

The profile $p_x \in \mathbb{R}^d$ is a local signature, where $(p_x)_i$ is an estimation of the probability that pixel x in frame f^0 moves toward pixel $x + \delta_i$, where $(\delta_i)_{i=0}^{d-1} \subset \mathbb{R}^2$ is a discrete grid of two-dimensional relative movements, as seen in Figure 15. For the numerical examples, we use δ_i that ranges on a uniform square grid of step $1/n$ and of width $11/n$ (which means that the expected displacement amplitude is 5 pixels).

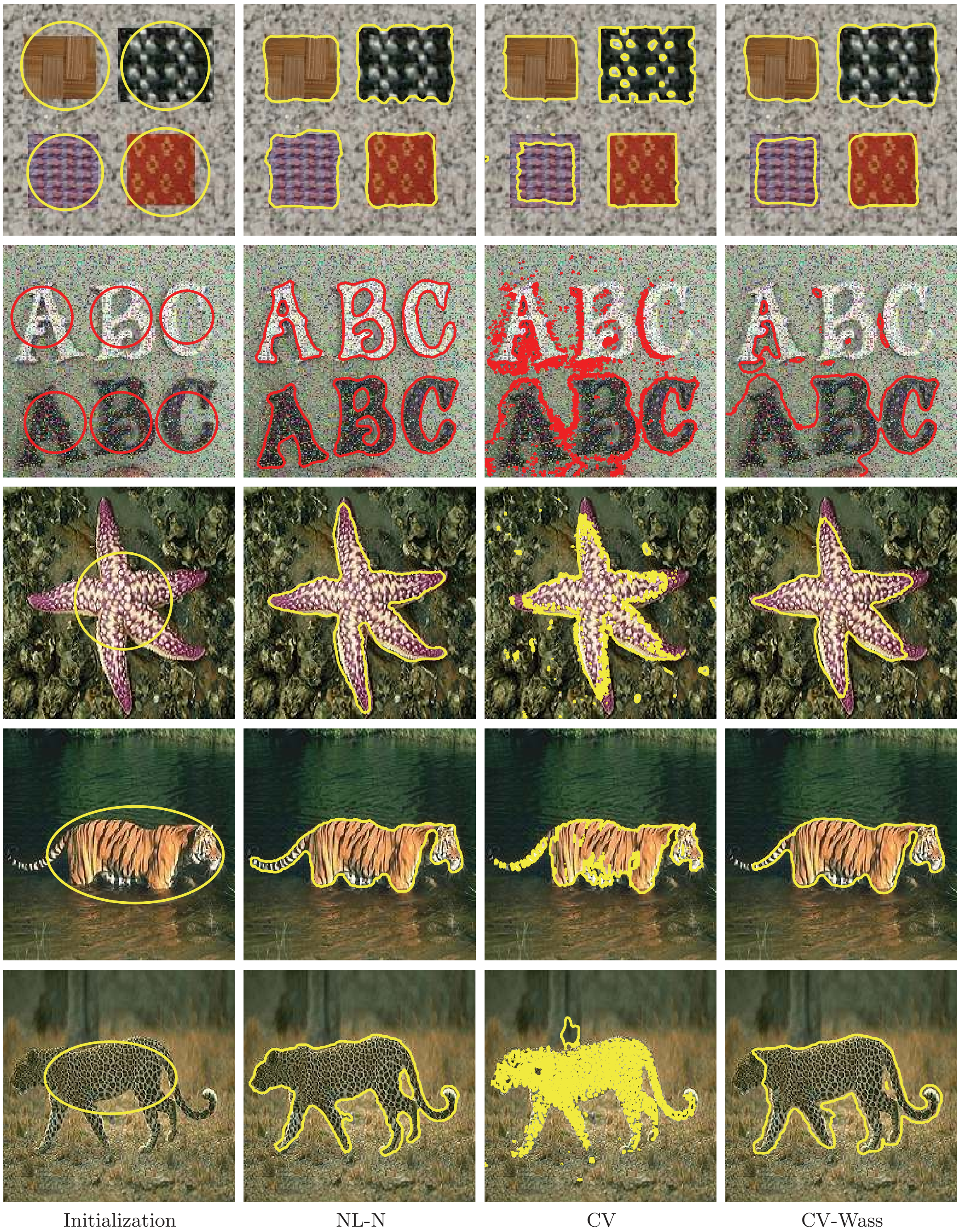
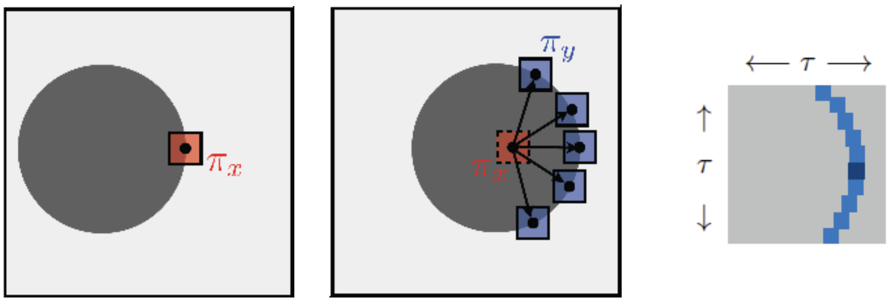


Figure 14. *Texture segmentation of our NL-N model* (3.6) *with the sliced Wasserstein distance* (7.4) *with* $|\Theta| = 3$ *fixed directions, and comparison with the vector-valued CV model* [12] *and the extended work (CV-Wass)* [2] *of* [38]. *Row* 1 *parameters:* $n = 185$, $\tau = 11/n$, $(\sigma, \xi) = (\infty, 31/n)$, $\gamma = 1.5/(n^3\tau^2)$. *Row* 2 *parameters:* $n = 176$, $\tau = 5/n$, $(\sigma, \xi) = (\infty, 31/n)$, $\gamma = 0.1/(n^3\tau^2)$. *Row* 3 *parameters:* $n = 192$, $\tau = 9/n$, $(\sigma, \xi) = (\infty, 31/n)$, $\gamma = 0.5/(n^3\tau^2)$. *Row* 4 *parameters:* $n = 192$, $\tau = 7/n$, $(\sigma, \xi) = (\infty, 31/n)$, $\gamma = 1/(n^3\tau^2)$. *Row* 5 *parameters:* $n = 193$, $\tau = 7/n$, $(\sigma, \xi) = (\infty, 31/n)$, $\gamma = 0.5/(n^3\tau^2)$.



Frame f^0 Frame f^1 Motion profile p_x

Figure 15. *Schematic display of the computation of a motion profile* p_x *for a translating disk. The blue patches* π_y*, where* $y = x + \delta_i$ *for some* i *in frame* f^1*, show the most probable patches that match* π_x *in frame* f^0*, thus resulting in a large value of* $(p_x)_i$ *(blue colors on the image in the last figure).*

To evaluate the movement, the patch π^0_x of size $\tau_m \times \tau_m$ extracted from f^0 around pixel x is compared to the patches $\pi^1_{x+\delta_i}$ extracted from frame f^1 around pixels $x+\delta_i$. Following [48, 49] the value of the profile is a decaying function of the L^2 distance between the patches:

$$(p_x)_i = \frac{1}{Z_x} e^{\frac{-\|\pi^0_x - \pi^1_{x+\delta_i}\|^2}{2\sigma_m^2}},$$

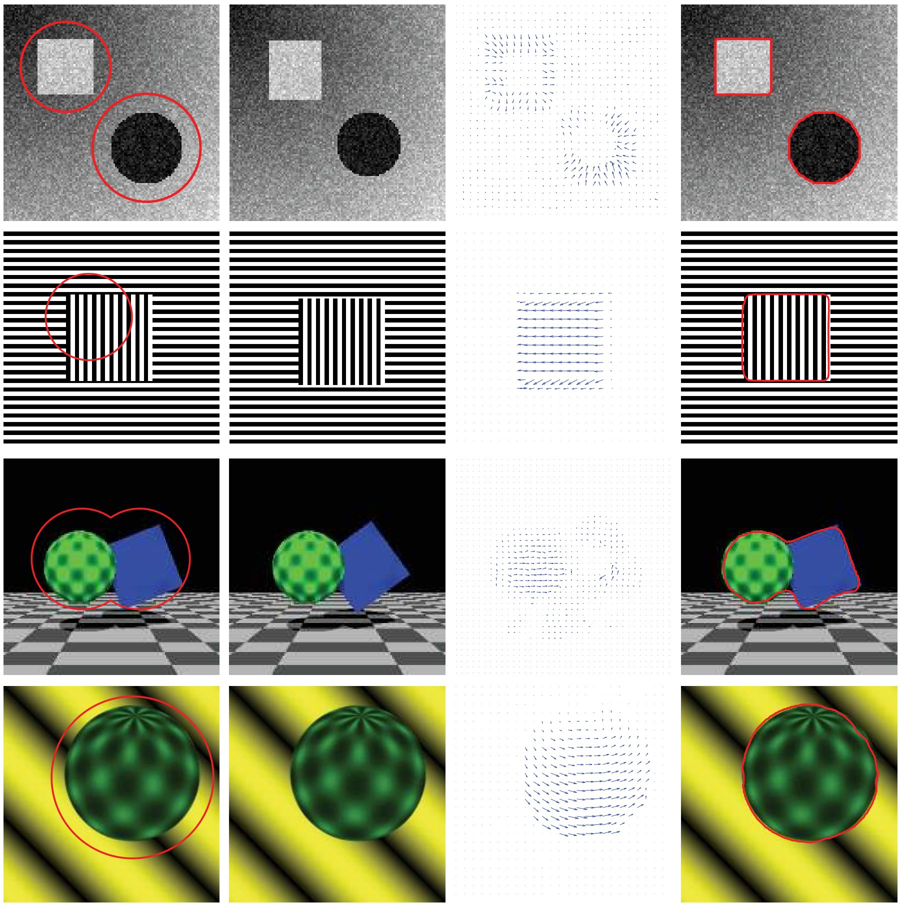
where the constant Z_x ensures that $\sum_i (p_x)_i = 1$ so that p_x resembles a density distribution of movements.

The nonlocal segmentation method is then applied to the segmentation of the profiles p_x using a correlation distance between the square root of the profiles:

$$d(p_x, p_y) = 1 - \sum_i \sqrt{(p_x)_i (p_y)_i}. \tag{8.1}$$

Note that we compute the square root of the motion profile, instead of using the motion profile itself as in [48, 49], in order to ensure that $d(p_x, p_x) = 0$.

8.2. Examples. Figures 16 and 17 present some motion segmentation results with the motion distance (8.1). The first two columns (a) and (b) show the two input images f^0 and f^1. In the first example, both objects translate with opposite directions and also exhibit a change of size (scaling). The second example shows a simpler movement (uniform translation), but the object has a uniform texture. The third example shows two three-dimensional objects which are both translating and rotating in space. The fourth example shows a single object which is rotating. These examples show different types of movements that are not uniform translation and result in a smoothly varying motion profile, thus being well suited for our nonlocal segmentation method. To better show the movement in these sequences, column (c) shows the optical flow computed using the Lucas–Kanade method [35]. Note that this flow is given for illustration purposes only and is not used to compute the segmentation. It is important because optical flow computation is a difficult task that would often lead to poor segmentation results. The last column (d) shows the segmentation results using our normalized energy with the motion distance.



(a) f^0 (b) f^1 (c) Optical flow (d) NL-N

Figure 16. *Segmentation of moving objects with our NL-N model* (3.6) *and the motion distance* (8.1)*. Parameters:* $n = 120, 100, 150, 200$ *(top to bottom),* $\tau = 11/n$*,* $(\sigma, \xi) = (\infty, 31/n)$*,* $\gamma = 0.005/n, 0.005/n, 0.03/n, 0.008/n$ *(top to bottom).*

Figure 17. *Segmentation of a moving car with our NL-N model* (3.6) *and the motion distance* (8.1)*. Initial curve (first column) on the first frame and final curves (second–fourth columns) on the first,* 13*th, and* 31*st frames of the sequence are shown. Parameters:* $n = 191$*,* $\tau = 11/n$*,* $(\sigma, \xi) = (\infty, 41/n)$*,* $\gamma = 0.005/n$*.*

Figure 17 shows that our model is able to segment a moving car of a video sequence. The initial curve is given only in the first frame (it needs to be quite close to the car), and the resulting curve is used as the initial curve for the next frame. The segmentation results are shown when f^0 is the first, 13th, and 31st frame of a movie sequence.

For these numerical examples, we use motion patches of width $\tau_m = 3/n$, except for the taxi sequence where $\tau_m = 7/n$. The motion profiles are computed using a variance that is adapted manually for each case. We use $\sigma_m = 0.2/n\tau_m$ for the first and second examples, $\sigma_m = 0.06/n\tau_m$ for the third, $\sigma_m = 0.08/n\tau_m$ for the fourth, and $\sigma_m = 0.25/n\tau_m$ for the taxi sequence.

We note that we try to initially locate curves outside moving objects, especially when objects have little texture where the motion profiles are almost flat; see, for instance, the first and third examples. In this way, our model can detect the object boundary where the variation of motion profile is largest.

9. Multiphase segmentation.

This section generalizes the models proposed in the previous sections to handle an arbitrary number $N \geqslant 2$ of regions. This is performed by extending the energy $E(\Omega)$ associated to a single domain Ω to an energy $E_{\mathrm{M}}(\Omega_1, \ldots, \Omega_m)$ on m domains and then replacing the original problem (2.1) by a joint minimization on the m domains

$$\min_{\Omega_1,\ldots,\Omega_m} \mathcal{E}_{\mathrm{M}}(\Omega_1, \ldots, \Omega_m) = E_{\mathrm{M}}(\Omega_1, \ldots, \Omega_m) + \gamma \sum_{j=1}^{m} L(\Omega_j). \tag{9.1}$$

Section 9.1 recalls the multiphase extension proposed in [57, 46], where the number of segmented regions N is equal to the number of domains $N = m$. Section 9.2 details an alternative extension where $N = 2^m$, which follows closely [52].

The problem (9.1) can be solved using a level set function φ_j to represent each domain $\Omega_j = \{x \setminus \varphi_j(x) > 0\}$. The active contour evolution is obtained by performing a gradient descent of (9.1), which leads to a system of m coupled PDEs:

$$\forall\, i = 1, \ldots, m, \quad \frac{\partial \varphi_i}{\partial t} = -\frac{\partial E_{\mathrm{M}}}{\partial \varphi_i}(\varphi_1, \ldots, \varphi_m) - \gamma \nabla L(\varphi_i), \tag{9.2}$$

where $\frac{\partial E_{\mathrm{M}}}{\partial \varphi_i}$ is a level set gradient direction associated to the mapping $\Omega_i \mapsto E_{\mathrm{M}}(\Omega_1, \ldots, \Omega_m)$.

9.1. Multiphase repulsive (MR) method.

Following the multiphase representation [57, 46], we introduce an energy to enforce that $\{\Omega_1, \ldots, \Omega_m\}$ are disjoint:

$$F(\Omega_1, \ldots, \Omega_m) = \int_{[0,1]^2} \left(\sum_{j=1}^{m} \chi_{\Omega_j}(x) - 1 \right)^2 \mathrm{d}x, \tag{9.3}$$

where we have used the indicator function

$$\chi_\Omega(x) = \begin{cases} 1 & \text{if} \quad x \in \Omega, \\ 0 & \text{otherwise.} \end{cases} \tag{9.4}$$

Any single region energy $E(\Omega)$ is extended into a multiphase energy as

$$E_{\mathrm{M}}(\Omega_1,\dots,\Omega_m)=\lambda F(\Omega_1,\dots,\Omega_m)+\sum_{j=1}^{m}E(\Omega_j), \tag{9.5}$$

where $\lambda>0$ is a parameter that should be large enough to prevent the domains from overlapping. The active contour evolution (9.2) is implemented using

$$\frac{\partial E_{\mathrm{M}}}{\partial\varphi_i}(\varphi_1,\dots,\varphi_m)(x)=\lambda\frac{\partial F}{\partial\varphi_i}(\varphi_1,\dots,\varphi_m)(x)+\nabla E(\varphi_i), \tag{9.6}$$

where ∇E is the level set gradient of the energy considered, and $\frac{\partial F}{\partial\varphi_i}$ is detailed in the following proposition.

Proposition 4. *One has, for $i=1,\dots,m$,*

$$\frac{\partial F}{\partial\varphi_i}(\varphi_1,\dots,\varphi_m)(x)=\|\nabla\varphi_i(x)\|\left(-1+2\sum_{j\neq i}\chi_{\Omega_j}(x)\right), \tag{9.7}$$

where $\Omega_j=\{x\setminus\varphi_j(x)>0\}$.

Proof. One can write

$$F(\Omega_1,\dots,\Omega_m)=\int_{\Omega_i}\left(\sum_{j\neq i}\chi_{\Omega_j}(x)\right)^2\mathrm{d}x+\int_{\Omega_i^c}\left(\sum_{j\neq i}\chi_{\Omega_j}(x)-1\right)^2\mathrm{d}x. \tag{9.8}$$

Using Proposition 1, one has

$$\frac{\partial F}{\partial\varphi_i}(\varphi_1,\dots,\varphi_m)(x)=\|\nabla\varphi_i(x)\|\left(\sum_{j\neq i}\chi_{\Omega_j}(x)\right)^2-\left(\sum_{j\neq i}\chi_{\Omega_j}(x)-1\right)^2 \tag{9.9}$$

and hence the result. ■

9.2. Multiphase intersection (MI) method. Following the method introduced in [52], we consider all possible intersections obtained using the sets $\{\Omega_j,\Omega_j^c\}_{j=1}^m$. This gives a segmentation of the image using $N=2^m$ nonoverlapping regions.

Any single-region energy $E(\Omega)$ is extended to a multiphase setting as

$$E_{\mathrm{M}}(\Omega_1,\dots,\Omega_m)=\sum_{s\in\{+1,-1\}^m}E(\Omega_1^{s_1}\cap\dots\Omega_m^{s_m}), \tag{9.10}$$

where we have used the notation $\Omega_j^{+1}=\Omega_j$ and $\Omega_j^{-1}=\Omega_j^c$. Note that the minimization (9.1) is performed on m domains, but it produces a segmentation in 2^m regions.

Computing the partial level set gradient $\frac{\partial E}{\partial\varphi_j}$ can be quite involved since it is made of 2^m terms. To obtain simpler formula, the following proposition makes use of the special structure of our nonlocal energies and introduces the indicator functions of a set $(\Omega\times\Omega)\cup(\Omega^c\times\Omega^c)$:

$$\hat{\chi}_\Omega(x,y)=\begin{cases}1 & \text{if}\quad (x,y)\in(\Omega\times\Omega)\cup(\Omega^c\times\Omega^c),\\ 0 & \text{otherwise.}\end{cases} \tag{9.11}$$

Proposition 5. *Let $\varphi_1, \dots, \varphi_m$ be fixed level set functions. For $i = 1, \dots, m$, we denote, for any function $f(x,y)$,*

$$f^{[i]}(x,y) = f(x,y) \prod_{j \neq i} \hat{\chi}_{\Omega_j}(x,y). \tag{9.12}$$

For $E = \bar{E}^{U}_{NL}$ of (3.2), a level set gradient is given, for $j = 1, \dots, m$, by

$$\frac{\partial E_{\mathrm{M}}}{\partial \varphi_i}(\varphi_1, \dots, \varphi_m) = \|\nabla \varphi_i(x)\| \left(\int_{\Omega_i} K^{[i]}(x,y)\mathrm{d}y - \int_{\Omega_i^c} K^{[i]}(x,y)\mathrm{d}y \right), \tag{9.13}$$

where K is defined in (3.3). For $E = \bar{E}^{N}_{NL}$ of (3.6), a level set gradient is given, for $j = 1, \dots, m$, by

$$\frac{\partial E_{\mathrm{M}}}{\partial \varphi_i}(\varphi_1, \dots, \varphi_m) = \nabla \mathcal{F}_{1,K^{[i]},G^{[i]}_\sigma}(\varphi_i) - \nabla \mathcal{F}_{1,K^{[i]},G^{[i]}_\sigma}(-\varphi_i), \tag{9.14}$$

where $\nabla\mathcal{F}$ is defined in (2.18).

Proof. The multiphase extension of the unnormalized energy (3.2) can be conveniently written as follows:

$$E^{\text{NL-U}}_{\mathrm{M}}(\Omega_1, \dots, \Omega_m) = \sum_{i=1}^{m} \iint K^{[i]}(x,y)\hat{\chi}_{\Omega_i}(x,y)\mathrm{d}y\mathrm{d}x = \sum_{i=1}^{m} \left\{ \mathcal{F}_{K^{[i]}}(\Omega_i) + \mathcal{F}_{K^{[i]}}(\Omega_i^c) \right\}, \tag{9.15}$$

where $\mathcal{F}_g$ is defined in (2.11). The result follows from Proposition 2, using the fact that $K^{[i]}$ is a symmetric kernel.

The multiphase extension of the normalized energy (3.6) (NL-N) is written as follows:

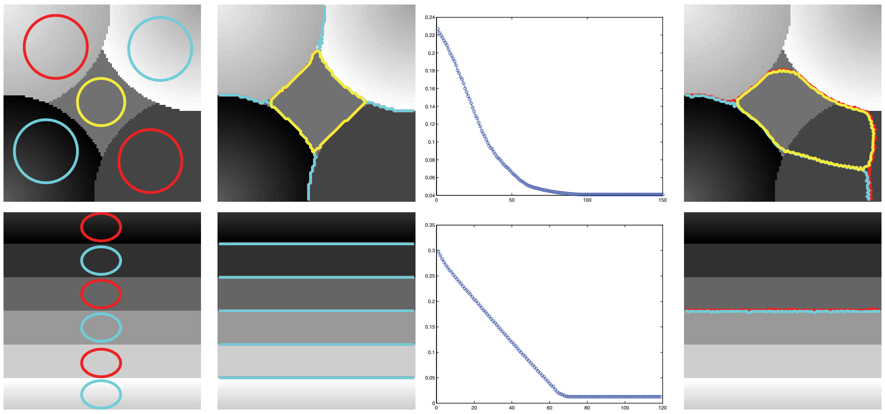
$$\begin{aligned} E^{\text{NL-N}}_{\mathrm{M}}(\Omega_1, \dots, \Omega_m) &= \sum_{i=1}^{m} \int \frac{\int K^{[i]}(x,y)\hat{\chi}_{\Omega_i}(x,y)\mathrm{d}y}{\int G^{[i]}_\sigma(x,y)\hat{\chi}_{\Omega_i}(x,y)\mathrm{d}y}\mathrm{d}x \\ &= \sum_{i=1}^{m} \left\{ \mathcal{F}_{1,K^{[i]},G^{[i]}_\sigma}(\Omega_i) + \mathcal{F}_{1,K^{[i]},G^{[i]}_\sigma}(\Omega_i^c) \right\}, \end{aligned} \tag{9.16}$$

where $\mathcal{F}_{f,g,h}$ is defined in (2.17). The result follows from Proposition 3. ■

9.3. Examples. In the following numerical experiments, we consider the multiphase extensions of the CV energy E_{CV} defined in (4.1), the NL-U energy defined in (3.2), and the NL-N energy defined in (3.6).

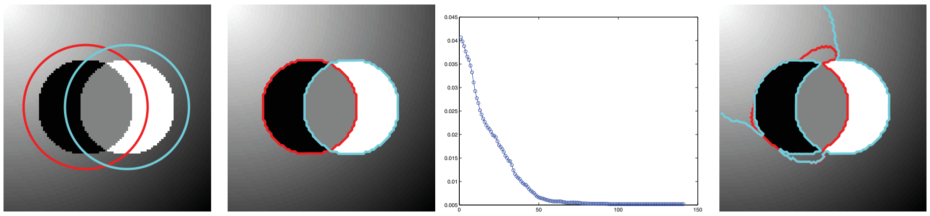
Note that for the CV energy, there is an additional parameter for each of the N regions to be segmented, which generalizes the pair (p_1, p_2) of parameters in the original energy (4.1). Each parameter is updated during the active contour evolution in a way similar to (4.2).

Figure 18 presents segmentation results of the multiphase model MR with our unnormalized energy NL-U using the L^2 patch distance (5.1) and the CV energy. Since two foreground objects are touching, the nonlocal energy requires multiphase extensions to perform a correct segmentation. Due to the fact that our nonlocal energy can segment the region with spatially smoothly varying features and separated regions with different features with one level



Initial curves NL-U Plot of energy $\mathcal{E}_{\mathrm{M}}^{\text{NL-U}}$ CV

Figure 18. *Segmentation results of multiphase method MR with the NL-U energy and the CV energy. Parameters:* $n = 100$, $\tau = 1/n$, $(\sigma, \xi) = (\infty, 31/n)$; *NL-U:* $\gamma = 100/(n^5\xi^2\tau^2)$, $\lambda = 200/(n^5\xi^2\tau^2)$ *(top),* $\lambda = 300/(n^5\xi^2\tau^2)$ *(bottom); CV:* $(\gamma, \lambda) = (0.1/(n^3\tau^2), 0.1/(n^2\tau^2))$.



Initial curves NL-U Plot of energy $\mathcal{E}_{\mathrm{M}}^{\text{NL-U}}$ CV

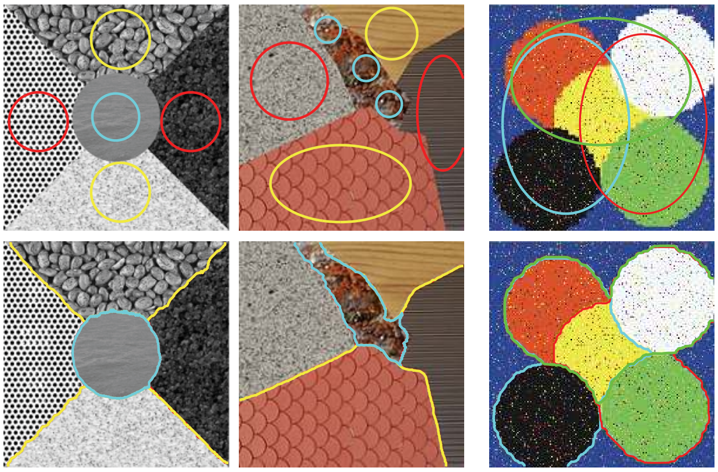
Figure 19. *Segmentation results of multiphase method MI with the NL-U energy and the CV energy. Parameters:* $n = 100$, $\tau = 1/n$, $(\sigma, \xi) = (\infty, 31/n)$, $\gamma = 100/(n^5\xi^2\tau^2)$ *(NL-U),* $\gamma = 0.1/(n^3\tau^2)$ *(CV).*

set function, the multiphase model MR with our NL-U energy also could segment the regions with smoothly varying intensity values, and it could segment the regions with a small number of level set functions ($m = 3$ for the first example, $m = 2$ for the second example). On the other hand, the CV energy fails to segment the first example due to the spatially varying intensity values, and it needs more level set functions for the second example. This figure also displays the convergence of the energy $\mathcal{E}_{\mathrm{M}}$ using the MR method (9.5) with the NL-U energy as a function of the iteration index ℓ.

Figure 19 presents segmentation results of the multiphase model MI with our unnormalized energy NL-U using the L^2 patch distance (5.1) and the CV energy. The multiphase model MI with our NL-U energy could segment the regions with smoothly varying background with two level set functions, while the CV energy fails in the segmentation due to the spatially varying background. This figure also displays the convergence of the energy $\mathcal{E}_{\mathrm{M}}$ using the MI method (9.15) with the NL-U energy as a function of the iteration index ℓ.

Figure 20 shows the result with our normalized energy (NL-N) using the sliced Wasserstein distance between patches (7.5). The sliced distance for color distributions in $\mathbb{R}^3$ is implemented using $\Theta = \{(1,0,0), (0,1,0), (0,0,1)\}$.

The first two examples are grayscale and color textures that are handled efficiently using



MR method MR method MI method

Figure 20. *Segmentation results of MR and MI methods with NL-N energy and Wasserstein patch distance. Parameters:* $n = 192$, $\tau = 7/n$, $(\sigma, \xi) = (\infty, 41/n)$, $(\gamma, \lambda) = (0.7/(n^3\tau^2), 3/(n^2\tau^2))$. *Parameters:* $n = 176$, $\tau = 7/n$, $(\sigma, \xi) = (\infty, 41/n)$, $(\gamma, \lambda) = (0.8/(n^3\tau^2), 4/(n^2\tau^2))$. *Parameters:* $n = 160$, $\tau = 3/n$, $(\sigma, \xi) = (\infty, 41/n)$, $\gamma = 0.1/(n^3\tau^2)$.

our statistical patch distance. Although there are five regions, the MR method gives a correct segmentation with $m = 3$ level set functions. In these examples, the repulsive force plays an important role in making the curves evolve to the boundaries, especially when a small value of σ (locality parameter) is used.

The third example is contaminated by salt-and-pepper noise (noisy pixel takes either salt value 1 or pepper value 0) with noise density 0.05 (salt noise and pepper noise have a noise density of 0.05/2).This shows a complicated segmentation situation because four regions are meeting at some locations. The MI method gives a correct segmentation with $m = 3$ functions that can segment up to eight phases. Note that the MR method would have required at least $m = 4$ functions. This is, however, much smaller than what would have been required (at least $m = 6$) if the CV energy had been used instead of NL-N.

10. Conclusion. In this article, we have proposed a novel class of nonlocal energies for image segmentation, which makes use of patches and allows us to constrain local homogeneity of features. We have illustrated the superiority of our models over existing active contour models. Due to the local homogeneity property, our segmentation model is able to detect regions with smoothly spatially varying features and segment separated objects with different features with a level set function. We have instantiated our models based on intensity, color, texture, or motion information, by designing appropriate metrics between patches such as the L^2 norm for piecewise smooth features (intensity, color values, or Gabor features for locally oriented textures), Wasserstein distance for locally homogeneous random fields, or motion signature correlation. The Wasserstein distance and its sliced approximation allow us to segment complicated textural features in arbitrary dimension, and the motion distance enables us to detect moving objects with locally varying motion. We also extend our models to the multiphase level set frameworks that enable us to segment an image with multiple junctions.

Our multiphase models are also able to partition regions with smoothly varying features and with a smaller number of level set functions, due to the local homogeneity property. All these properties are significant extensions of existing region-based models crucial in solving difficult image segmentation problems.

REFERENCES

[1] G. Aubert, M. Barlaud, O. Faugeras, and S. Jehan-Besson, *Image segmentation using active contours: Calculus of variations or shape gradients?*, SIAM J. Appl. Math., 63 (2003), pp. 2128–2154.

[2] Z. Bao, Y. Liu, Y. Peng, and G. Zhang, *Variational color image segmentation via chromaticity-brightness decomposition*, in Proceedings of MMM'10, the 16th International Conference on Advances in Multimedia Modeling, Springer-Verlag, Berlin, Heidelberg, 2010, pp. 295–302.

[3] A. Blake and A. Zisserman, *Visual Reconstruction*, MIT Press Ser. Artificial Intelligence, MIT Press, Cambridge, MA, 1987.

[4] X. Bresson and T. Chan, *Non-local Unsupervised Variational Image Segmentation Models*, Technical report, UCLA CAM Report 08-67, UCLA, Los Angeles, CA, 2008.

[5] T. Brox, A. Bruhn, and J. Weickert, *Variational motion segmentation with level sets*, in Proceedings of ECCV, the Ninth European Conference on Computer Vision, Springer-Verlag, Berlin, Heidelberg, 2006, pp. 471–483.

[6] T. Brox and D. Cremers, *On the statistical interpretation of the piecewise smooth Mumford–Shah functional*, in Scale Space and Variational Methods in Computer Vision, Lecture Notes in Comput. Sci. 4485, Springer-Verlag, Berlin, Heidelberg, 2007, pp. 203–213.

[7] A. Buades, B. Coll, and J. M. Morel, *A review of image denoising algorithms, with a new one*, Multiscale Model. Simul., 4 (2005), pp. 490–530.

[8] R. Burkard, M. Dell'Amico, and S. Martello, *Assignment Problems*, SIAM, Philadelphia, 2009.

[9] B. Caldairou, F. Rousseau, N. Passat, P. Habas, C. Studholme, and C. Heinrich, *A non-local fuzzy segmentation method: Application to brain MRI*, in Computer Analysis of Images and Patterns, Lecture Notes in Comput. Sci. 5702, Springer-Verlag, Berlin, Heidelberg, 2009, pp. 606–613.

[10] V. Caselles, F. Catté, T. Coll, and F. Dibos, *A geometric model for active contours in image processing*, Numer. Math., 66 (1993), pp. 1–31.

[11] V. Caselles, R. Kimmel, and G. Sapiro, *Geodesic active contours*, Int. J. Comput. Vision, 22 (1997), pp. 61–79.

[12] T. Chan, B. Sandberg, and L. Vese, *Active contours without edges for vector-valued images*, J. Vis. Commun. Image Rep., 11 (2000), pp. 130–141.

[13] T. Chan and L. Vese, *Active contours without edges*, IEEE Trans. Image Process., 10 (2001), pp. 266–277.

[14] L. D. Cohen, *On active contour models and balloons*, CVGIP: Image Underst., 53 (1991), pp. 211–218.

[15] L. D. Cohen, *Avoiding local minima for deformable curves in image analysis*, in Curves and Surfaces with Applications in CAGD, A. Le Méhauté, C. Rabut, and L. L. Schumaker (eds.), Vanderbilt University Press., Nashville, TN, 1997.

[16] L. D. Cohen and I. Cohen, *Finite element methods for active contour models and balloons for* 2-*D and* 3-*D images.*, IEEE Trans. Pattern. Anal. Mach. Intell., 15 (1993), pp. 1131–1147.

[17] D. Cremers and S. Soatto, *Motion competition: A variational framework for piecewise parametric motion segmentation*, Int. J. Comput. Vision, 62 (2005), pp. 249–165.

[18] M. C. Delfour and J.-P. Zolésio, *Shapes and Geometries: Metrics, Analysis, Differential Calculus, and Optimization*, 2nd ed., SIAM, Philadelphia, 2011.

[19] A. Elmoataz, O. Lezoray, and S. Bougleux, *Nonlocal discrete regularization on weighted graphs: A framework for image and manifold processing*, IEEE Trans. Image Process., 17 (2008), pp. 1047–1060.

[20] D. Gabor, *Theory of communication*, J. Inst. Elect. Eng., 93 (1946), pp. 429–441.

[21] G. Gilboa and S. Osher, *Nonlocal linear image regularization and supervised segmentation*, Multiscale Model. Simul., 6 (2007), pp. 595–630.

[22] G. Gilboa and S. Osher, *Nonlocal operators with applications to image processing*, Multiscale Model. Simul., 7 (2008), pp. 1005–1028.

[23] A. Herbulot, S. Jehan-Besson, S. Duffner, M. Barlaud, and G. Aubert, *Segmentation of vectorial image features using shape gradients and information measures*, J. Math. Imaging Vision, 25 (2006), pp. 365–386.

[24] N. Houhou, X. Bresson, A. Szlam, T. F. Chan, and J. P. Thiran, *Semi-supervised segmentation based on non-local continuous min-cut*, in SSVM '09, Scale Space and Variational Methods in Computer Vision, Lecture Notes in Comput. Sci. 5567, Springer-Verlag, Berlin, Heidelberg, 2009, pp. 112–123.

[25] M. Jung, G. Peyré, and L. D. Cohen, *Nonlocal active contours*, in Scale Space and Variational Methods in Computer Vision, Lecture Notes in Comput. Sci. 6667, Springer-Verlag, Berlin, Heidelberg, 2012.

[26] M. Jung, G. Peyré, and L. D. Cohen, *Nonlocal segmentation and inpainting*, in Proceedings of ICIP'11, the 18th IEEE International Conference on Image Processing, Brussels, Belgium, 2011, pp. 3373–3376.

[27] M. Jung, G. Peyré, and L. D. Cohen, *Texture segmentation via non-local non-parametric active contours*, in Energy Minimization Methods in Computer Vision and Pattern Recognition, Lecture Notes in Comput. Sci. 6819, Springer-Verlag, Berlin, Heidelberg, 2011, pp. 74-88.

[28] M. Kass, A. Witkin, and D. Terzopoulos, *Snakes: Active contour models*, Int. J. Comput. Vision, 1 (1988), pp. 321–331.

[29] S. Kichenassamy, A. Kumar, P. Olver, A. Tannenbaum, and A. Yezzy, *Gradient flows and geometric active contour models*, in Proceedings of ICCV '95, the Fifth International Conference on Computer Vision, Cambridge, MA, IEEE Computer Society, Washington, DC, 1995, pp. 810–815.

[30] J. Kim, J. W. Fisher, A. Yezzi, M. Cetin, and A. S. Willsky, *A nonparametric statistical method for image segmentation using information theory and curve evolution*, IEEE Trans. Image Process., 14 (2005), pp. 1486–502.

[31] R. Kimmel, *Fast edge integration*, in Geometric Level Set Methods in Imaging, Vision, and Graphics, S. Osher and N. Paragios, eds., Springer-Verlag, New York, 2003.

[32] S. Kindermann, S. Osher, and P. W. Jones, *Deblurring and denoising of images by nonlocal functionals*, Multiscale Model. Simul., 4 (2005), pp. 1091–1115.

[33] T. S. Lee, D. Mumford, and A. Yuille, *Texture segmentation by minimizing vector-valued energy functionals: The coupled-membrane model*, in Proceedings of ECCV '92, the Second European Conference on Computer Vision, G. Sini, ed., Springer-Verlag, London, 1992, pp. 165–173.

[34] C. M. Li, C. Y. Kao, J. C. Gore, and Z. H. Ding, *Implicit active contours driven by local binary fitting energy*, in Proceedings of CVPR'07, IEEE Conference on Computer Vision and Pattern Recognition, 2007, pp. 1–7.

[35] B. Lucas and T. Kanade, *An iterative image registration technique with an application to stereo vision*, in Proceedings of IJCAI, the 7th International Conference on Artificial Intelligence, Vol. 2, Morgan Kaufmann, San Francisco, 1981, pp. 674–679.

[36] R. Malladi, J. A. Sethian, and B. C. Vemuri, *Shape modeling with front propagation: A level set approach*, IEEE Trans. Pattern Anal. Mach. Intell., 17 (1995), pp. 158–175.

[37] D. Mumford and J. Shah, *Optimal approximations by piecewise smooth functions and associated variational problems*, Comm. Pure Appl. Math., 42 (1989), pp. 577–685.

[38] K. Ni, X. Bresson, T. F. Chan, and S. Esedoglu, *Local histogram based segmentation using the Wasserstein distance*, Int. J. Comput. Vision, 84 (2009), pp. 97–111.

[39] S. Osher and J. Sethian, *Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations*, J. Comput. Phys., 79 (1988), pp. 12–49.

[40] N. Paragios and R. Deriche, *Geodesic active contours for supervised texture segmentation*, in Proceedings of ICCV, IEEE Conference on Computer Vision and Pattern Recognition, 1999, pp. 422–427.

[41] N. Paragios and R. Deriche, *Geodesic active regions: A new framework to deal with frame partition problems in computer vision*, J. Vis. Commun. Image Rep., 13 (2002), pp. 249–268.

[42] G. Peyré, S. Bougleux, and L. Cohen, *Non-local regularization of inverse problems*, Inverse Probl. Imaging, 5 (2011), pp. 511–530.

[43] J. Piovano, M. Rousson, and T. Papadopoulo, *Efficient segmentation of piecewise smooth images*, in Scale Space and Variational Methods in Computer Vision, Lecture Notes in Comput. Sci. 4485, Springer-Verlag, Berlin, Heidelberg, 2007, pp. 709–720.

[44] J. Rabin, G. Peyré, J. Delon, and M. Bernot, *Wasserstein barycenter and its application to texture mixing*, in Scale Space and Variational Methods in Computer Vision, Lecture Notes in Comput. Sci. 6667, Springer-Verlag, Berlin, Heidelberg, 2012, pp. 435-446.

[45] C. Sagiv, N. A. Sochen, and Y. Y. Zeevi, *Integrated active contours for texture segmentation*, IEEE Trans. Image Process., 15 (2006), pp. 1633–1646.

[46] C. Samson, L. Blanc-Féraud, G. Aubert, and J. Zerubia, *A level set model for image classification*, Int. J. Comput. Vision, 40 (2000), pp. 187–197.

[47] B. Sandberg, T. Chan, and L. Vese, *A Level-set and Gabor Based Active Contour Algorithm for Segmenting Textured Images*, Technical report, UCLA CAM Report 02-39, UCLA, Los Angeles, CA, 2002.

[48] J. Shi and J. Malik, *Motion segmentation and tracking using normalized cuts*, in Proceedings of ICCV, the Sixth International Conference on Computer Vision, IEEE Computer Society, Washington, DC, 1998, pp. 1154–1160.

[49] J. Shi and J. Malik, *Normalized cuts and image segmentation*, IEEE Trans. Pattern Anal. and Mach. Intell., 22 (2002), pp. 888–905.

[50] S. Shirdhonkar and D. Jacobs, *Approximate earth mover's distance in linear time*, in Proceedings of the 21st IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, 2008.

[51] A. Tsai, A. Yezzi, and A. S. Willsky, *Curve evolution implementation of the Mumford-Shah functional for image segmentation, denoising, interpolation, and magnification*, IEEE Trans. Image Process., 10 (2001), pp. 1169–1186.

[52] L. Vese and T. Chan, *A multiphase level set framework for image segmentation using the Mumford and Shah model*, Int. J. Comput. Vision, 50 (2002), pp. 271–293.

[53] L. Wang, L. He, A. Mishra, and C. M. Li, *Active contours driven by local Gaussian distribution fitting energy*, Signal Process., 89 (2009), pp. 2435–2447.

[54] L. Wang, C. M. Li, Q. Suna, D. Xia, and C.-Y. Kao, *Active contours driven by local and global intensity fitting energy with application to brain MR image segmentation*, Comput. Med. Imaging Graph., 33 (2009), pp. 520–531.

[55] X. F. Wang, D. S. Huang, and H. Xu, *An efficient local Chan–Vese model for image segmentation*, Pattern Recogn., 43 (2010), pp. 603–618.

[56] A. Yezzi, A. Tsai, and A. Willsky, *A statistical approach to snakes for bimodal and trimodal imagery*, in Proceedings of ICCV'99, International Conference on Computer Vision, Vol. II, IEEE Computer Society, Washington, DC, 1999, pp. 898–903.

[57] H. K. Zhao, T. Chan, B. Merriman, and S. Osher, *A variational level set approach to multiphase motion*, J. Comput. Phys., 127 (1996), pp. 179–195.

[58] S. Zhu and A. Yuille, *Region competition: Unifying snakes, region growing, and Bayes/MDL for multiband image segmentation*, IEEE Trans. Pattern Anal. Mach. Intell., 18 (1996), pp. 884–900.