Shape-Tailored Local Descriptors and their Application to Segmentation and Tracking

Naeemullah Khan[1], Marei Algarni[1], Anthony Yezzi[2], and Ganesh Sundaramoorthi[1]
[1]King Abdullah University of Science & Technology (KAUST), Saudi Arabia
[2] School of Electrical & Computer Engineering, Georgia Institute of Technology, USA
{naeemullah.khan,marei.algarni,ganesh.sundaramoorthi}@kaust.edu.sa, ayezzi@ece.gatech.edu

Abstract

We propose new dense descriptors for texture segmentation. Given a region of arbitrary shape in an image, these descriptors are formed from shape-dependent scale spaces of oriented gradients. These scale spaces are defined by Poisson-like partial differential equations. A key property of our new descriptors is that they do not aggregate image data across the boundary of the region, in contrast to existing descriptors based on aggregation of oriented gradients. As an example, we show how the descriptor can be incorporated in a Mumford-Shah energy for texture segmentation. We test our method on several challenging datasets for texture segmentation and textured object tracking. Experiments indicate that our descriptors lead to more accurate segmentation than non-shape dependent descriptors and the state-of-the-art in texture segmentation.

1. Introduction

Local invariant descriptors (e.g., [27, 26, 10, 39, 37]) are image statistics at each pixel that describe neighborhoods in a way that is invariant to geometric and photometric nuisances. They are typically computed by aggregating smoothed oriented gradients within a neighborhood of the pixel. These descriptors play an important role in characterizing local textural properties. This is because a texture consists of small tokens, called textons [20], which may vary by small geometric and photometric nuisances but are otherwise stationary. Careful construction of these descriptors is crucial since they play a key role in low-level segmentation, which in turn plays a role in higher level tasks such as object detection and segmentation.

Existing local invariant descriptors aggregate oriented gradients in predefined pixel neighborhoods that could contain image data from different textured regions, especially near the boundary of the texture. This leads to ambiguity in grouping descriptors, especially for descriptors near the boundary. This could lead to segmentation errors if descriptors are grouped to form a segmentation. The problem is exacerbated when the textons in the textures are large. In this case, the neighborhood of the descriptor needs to be chosen large to fully capture texton data. See Fig. 1. Ideally, one would need to construct local descriptors that aggregate oriented gradients only from within textured regions. However, the segmentation is not known a-priori. Thus, it is necessary to solve for the local descriptors and the region of the segmentation in a joint problem.

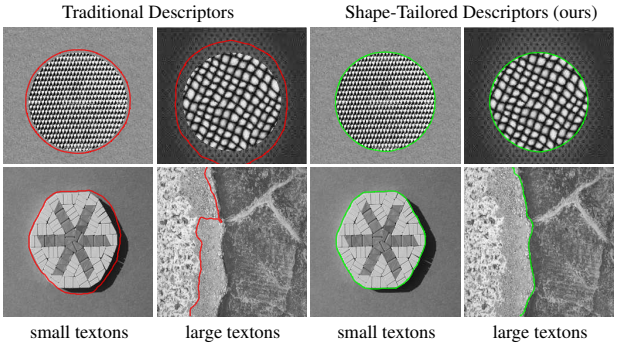

Figure 1. [Left]: Descriptors that aggregate local image data across boundaries of textured regions lead to segmentation errors. The problem is exacerbated as the texton size increases. [Right]: Segmentation by Shape-Tailored Descriptors (our method).

In this paper, we address this joint problem. This is accomplished in two steps. First, we construct novel dense local invariant descriptors, called *Shape-Tailored Local Descriptors (STLD)*. These descriptors are formed from shape-dependent scale spaces of oriented gradients. The shape-dependent scale spaces are the solution of Poisson-like partial differential equations (PDE). Of particular importance is the fact that these scale-spaces are defined within a region of arbitrary shape and do not aggregate data outside the region of interest. Second, we incorporate Shape-Tailored Descriptors into the Mumford-Shah energy [29] as an example energy based on these descriptors. Optimization jointly

estimates Shape-Tailored Descriptors and their support region, which forms the segmentation.

Contributions: 1. Our main contribution is to define new dense local descriptors by using shape-dependent scale spaces of oriented gradients. 2. We show that our new descriptors give more accurate segmentation than their non shape-dependent counterparts for texture segmentation. 3. We apply our descriptors to disocclusion detection [43] in object tracking improving state-of-the-art.

1.1. Related Work

Many approaches [45, 32, 22, 9, 28, 18, 33] to texture segmentation partition the image into regions that have global intensity distributions that are maximally separated by a distance on distributions. A drawback of global intensity distributions is that spatial relations are lost. This is important in characterizing textures. Spatial correlations between neighboring pixels are considered in [3] by creating a vector of the four neighboring pixel values for each pixel. Grouping these vectors improves segmentation. A recent approach [19] uses frequencies of neighboring pixel pairs within the image to determine texture boundaries. In [38], small neighborhoods are obtained from a super-pixelization and used in segmentation. Super-pixels may cross texture boundaries, aggregating data across boundaries.

Larger neighborhoods are considered in [30]. Gabor filters at various scales and orientations have been used widely in texture analysis (e.g., [27]), and the response of these filters (or others [35, 42]) have been used as a descriptor in texture segmentation (e.g., [24, 36]), and as an edge-detector [1]. These approaches depend on the size of the neighborhood chosen. The optimal size is determined by peaks in the entropy profile of intensity distributions of increasingly sized neighborhoods in [17, 4, 16]. An aspect that remains an issue in all these methods is that neighborhoods may cross texture boundaries, which our method addresses. In [14], these boundary effects are mitigated by a top-down correction step, however, the method only deals with neighborhoods that are a few pixels in length.

We use variational methods to optimize the Mumford-Shah energy incorporating our descriptors. Many active contours [21] are driven to group pixel intensities based on intensity statistics. For example, global intensity means in the regions are used in [8, 44], and global histograms are used in [22, 28]. Since images are not always described by global intensity statistics, local intensity statistics have been used to group pixels (e.g., [29, 23, 11, 6]). Since these methods aim to group pixels, they do not capture texture in many cases. These energies are optimized using gradient descent, but more recently methods of convex relaxations have improved results in many cases [7, 5, 34].

Our Shape-Tailored descriptors are the solutions of PDE defined within regions. Thus, the energies we optimize involve integrals over the regions of functions of PDE that are dependent on the regions. While we use direct methods of calculus of variations to optimize these energies, one can also use shape gradients [12] (see also, [2, 15]). Our contribution lies in introducing new descriptors for texture segmentation, and not in the method of optimization.

2. Shape-Tailored Descriptors Formulation

In this section, we define Shape-Tailored Descriptors. We compute their gradient with respect to shape perturbations, and then the gradient of a region-based functional involving the descriptors. These results will be needed to optimize the energy for segmentation.

2.1. Defining Shape-Tailored Descriptors

Let $\Omega \subset \mathbb{R}^2$ be the domain of an image $I : \Omega \to \mathbb{R}^k$ ($k \geq 1$). Let $R \subset \Omega$ be an arbitrarily shaped region with non-zero area and smooth boundary ∂R. We compute local descriptors for each $x \in R$. The descriptor describes I in a neighborhood of x inside R. The descriptors at $x \in R$ will be aggregations of image data I and oriented gradients within multiple neighborhoods of x in R. This can be accomplished conveniently using scale-spaces [25] defined by PDE. This motivates the definition below.

Definition 1 (Shape-Tailored Local Descriptors)**.** *Let $R \subset \Omega$ be a bounded region with non-zero area and smooth boundary ∂R. Let $I : \Omega \to \mathbb{R}^k$. A* **Shape-Tailored Descriptor**, $\mathbf{u} : R \to \mathbb{R}^M$ *(where $M = n \times m$, $n, m \geq 1$) consists of components $u_{ij} : R \to \mathbb{R}$ so that $\mathbf{u} = (u_{11}, \ldots, u_{1m}, \ldots, u_{n1}, \ldots, u_{nm})^T$. The components are defined as:*

$$\begin{cases} u_{ij}(x) - \alpha_i \Delta u_{ij}(x) = J_j(x) & x \in R \\ \nabla u_{ij}(x) \cdot N = 0 & x \in \partial R \end{cases}, \tag{1}$$

where $1 \leq i \leq n$, $1 \leq j \leq m$, Δ denotes the Laplacian, ∇ denotes the gradient, N is the unit outward normal to R, $\alpha_i > 0$ are scales*, and $J_j : R \to \mathbb{R}$ are point-wise functions of the image I. In vector form, this is equivalent to*

$$\begin{cases} \mathbf{u}(x) - A\Delta \mathbf{u}(x) = \mathbf{J}(x) & x \in R \\ D\mathbf{u}(x)N = \mathbf{0} & x \in \partial R \end{cases}, \tag{2}$$

where $A = diag(\alpha_1 1_{1\times m}, \ldots, \alpha_n 1_{1\times m})$ (an $M \times M$ diagonal matrix), $1_{1\times m}$ is a $1 \times m$ matrix of ones, D denotes the spatial derivative operator, and $\mathbf{J} = (J_1, \ldots, J_m, \ldots, J_1, \ldots, J_m, \ldots)^T$.

Remark 1. *Possible choices for $\mathbf{J}$ can include oriented gradients of the gray-scale value of I, color channels of I, and the grayscale image I_g. Note oriented gradients of the grayscale image I_g, for an angles θ_i are defined as*

$I_{\theta_i}(x) := \int_{\theta_i}^{\theta_i+\Delta\theta} |\nabla I_g(x) \cdot e_{\theta'}| \, \mathrm{d}\theta'$ *where* e_θ *indicates a unit direction vector in the direction of* θ, $|\cdot|$ *is absolute value, and* $\Delta\theta > 0$ *is the angle bin size. Unless otherwise specified, we choose* J_j*'s to be the color channels and oriented gradients at angles* $\theta = \{0, \pi/8, 2\pi/8, \ldots, 7\pi/8\}$.

Remark 2. *The PDE (1), for each* θ*, form a scale space with scale parameter* α_i*. The PDE is the minimizer of*

$$E(u) = \int_R (J_j(x) - u(x))^2 \, \mathrm{d}x + \alpha_i \int_R |\nabla u(x)|^2 \, \mathrm{d}x.$$

Thus, u_{ij} *is a smoothing of* J_j *and* α_i *controls the amount of smoothing. Using the Green's function* K_{α_i}*, to be introduced in Section 2.2,* $u_{ij}(x) = \int_R K_{\alpha_i}(x, y) J_j(y) \, \mathrm{d}y$*, where* $K_\alpha(x, .)$ *is a weight function. It has weight concentrated near* x*, and therefore defines an effective neighborhood around* x *in which to aggregate data. An advantage of solving the PDE (1) is that* K_{α_i}*, i.e., the neighborhood, does not need to be computed explicitly, and the PDE can be solved in faster computational time than integrating the kernel (Green's function) directly.*

Remark 3. *The key property in defining Shape-Tailored Local Descriptors is the scale space defined within a region of arbitrary shape. Any other PDE besides the Poisson-like PDE (1) could also be a valid choice.*

Remark 4. *The descriptor* $\mathbf{u}$ *is motivated by its covariance / robustness properties. Indeed, the descriptor is covariant to planar rotations and translations. This follows from the covariance of the Laplacian. Further, the descriptor is robust to small deformations of the set* R*. This can be seen since locally any deformation is a translation, and the solution of the PDE can be approximated by taking local averages, which is robust to small translations. This robustness is useful for textures since textons (especially in textures in nature) within regions vary by small deformations.*

2.2. Shape-Tailored Descriptor Gradient

We now compute the variation of the descriptor $\mathbf{u}_R$ as the boundary ∂R is perturbed. The gradient with respect to the boundary can then be computed. Since the computations (proofs of Lemmas and Propositions) are involved, they are left to Supplementary Materials.

Since $\mathbf{u}$ has components u_{ij}, we compute the variation of u_{ij}. For simplicity of notation, we suppress ij and write u. We denote by h, a vector field defined on ∂R. This is a perturbation of ∂R. Thus, $h : \mathbb{S}^1 \to \mathbb{R}^2$ where $\mathbb{S}^1$ is the unit interval. We denote by $u_h(x) := \mathrm{d}u(x) \cdot h$ the variation of u at x with respect to perturbation of the boundary by h.

We first show that u_h satisfies a PDE that is the same as the descriptor PDE (1) but with a different boundary condition and forcing term:

Lemma 1 (PDE for Descriptor Variation)**.** *Let* u *satisfy the PDE (1),* h *be a perturbation of* ∂R*, and* u_h *denote the variation of* u *with respect to the perturbation* h*. Then*

$$\begin{cases} u_h(x) - \alpha_i \Delta u_h(x) = 0 & x \in R \\ \nabla u_h(x) \cdot N = u_s(x)(h_s \cdot N) - N^T H u(x) \cdot h & x \in \partial R \end{cases} \tag{3}$$

where s *is the arc-length parameter of* ∂R*,* h_s *denotes the derivative with respect to arc-length, and* $Hu(x)$ *denotes the Hessian matrix.*

One can now use the previous result to compute the gradient of u, $\nabla_c u$, with respect to $c = \partial R$. To do this, we express the solution of (3) using the Green's function [13], i.e., the fundamental solution, defined on R. The Green's function for (3) depends only on the structure of the PDE, i.e., left hand sides of (3), and not the particular forcing function or the right hand side of the boundary condition. Hence the Green's function for (3) is the same as the Green's function for (1). The Green's function is defined as follows:

Definition 2 (Green's Function for (3))**.** *The Green's function,* $K_{\alpha_i} : R \times R \to \mathbb{R}$*, for the problem (3) (and (1)) satisfies*

$$\begin{cases} K_{\alpha_i}(x, y) - \alpha_i \Delta_x K_{\alpha_i}(x, y) = \delta(x - y) & x, y \in R \\ \nabla_x K_{\alpha_i}(x, y) \cdot N = 0 & x \in \partial R, y \in R \end{cases} \tag{4}$$

where Δ_x *(*∇_x*) is the Laplacian (gradient) with respect to* x*, and* δ *is the Delta function.*

The gradient $\nabla_c \mathbf{u}(x)$ can now be computed:

Proposition 1 (Descriptor Gradient)**.** *The gradient with respect to* $c = \partial R$ *of* $u_{ij}(x)$ *(one component of* $\mathbf{u}(x)$*), which satisfies the PDE (1), is* $\nabla_c u_{ij}(x) =$

$$\left[\nabla u_{ij} \cdot \nabla_y K_{\alpha_i}(x, \cdot) + \frac{1}{\alpha_i} K_{\alpha_i}(x, \cdot)(u_{ij} - J_j) \right] N \tag{5}$$

where N *is the outward normal,* ∇_y *denotes the gradient wrt the second argument of* K_{α_i}*, and* Du *indicates the spatial derivative of* u*. We define* $\nabla_c \mathbf{u}(x)$ *to be the* $2 \times M$ *matrix with columns as the components* $\nabla_c u_{ij}(x)$*.*

Remark 5. *Note that* $\nabla_c \mathbf{u}(x)$ *is defined at each point of* c *for each* x*, and all the terms in expression (5) are evaluated at a point of the curve* $c(s)$*, which is suppressed for simplicity of notation.*

The Green's function is not expressible in analytic form for arbitrary shapes R. We will see that we will need to only compute region integrals of the gradient multiplied by a function. This, fortunately, may be expressed as a solution to a PDE, and thus does not require the Green's function. The integrals of descriptor gradients can be computed as:

Proposition 2 (Integrals of Descriptor Gradient)**.** *Let* $\mathbf{f}, \mathbf{g} : R \to \mathbb{R}^M$ *and* $\mathbf{u}$ *be the Shape-Tailored Descriptor in* R *(as in (2)). Define* $\mathbb{I}_d[R, \mathbf{u}, \mathbf{f}, \mathbf{g}]$ *as the quantity*

$$-\int_{\partial R} \nabla_c \mathbf{u}(x)\mathbf{g}(x)\,\mathrm{d}s(x) + \int_R \nabla_c \mathbf{u}(x)\mathbf{f}(x)\,\mathrm{d}x.$$

where $\mathrm{d}x$ *and* $\mathrm{d}s$ *are the area and arclength measure. Then*

$$\mathbb{I}_d[R, \mathbf{u}, \mathbf{f}, \mathbf{g}] = \left(tr[(D\mathbf{u})^T D\hat{\mathbf{u}}] + (\mathbf{u} - \mathbf{J})^T A^{-1}\hat{\mathbf{u}}\right) N \quad (6)$$

where N *is the outward normal to the boundary of* R*,* tr *denotes matrix trace, and*

$$\begin{cases} \hat{\mathbf{u}}(x) - A\Delta\hat{\mathbf{u}}(x) = \mathbf{f}(x) & x \in R \\ D\hat{\mathbf{u}}(x)N = \mathbf{g}(x) & x \in \partial R \end{cases}. \quad (7)$$

We now compute the gradient of a weighted area functional involving Shape-Tailored Descriptors. This result will be useful for computing gradients of energies designed for segmentation in Section 3.

Proposition 3 (Weighted Area Gradient)**.** *Let* $F : \mathbb{R}^M \to \mathbb{R}$ *and* $\mathbf{u} : R \to \mathbb{R}^M$ *be the Shape-Tailored Descriptor on* R*. Define the weighted area functionals as* $A_F = \int_R F(\mathbf{u}(x))\,\mathrm{d}x$*. Then*

$$\nabla_c A_F = (F \circ \mathbf{u})N + \mathbb{I}_d[R, \mathbf{u}, (\nabla F) \circ \mathbf{u}, \mathbf{0}] \quad (8)$$

where $\mathbb{I}_d$ *is defined as in Proposition 2.*

The dependence of the descriptor on the region induces the terms involving $\mathbb{I}_d$ in the above gradient. Those terms depend on $\hat{\mathbf{u}}$ defined in (7), which is the solution to another PDE defined on R. Thus, when performing a gradient descent of A_F, $\mathbf{u}$ and $\hat{\mathbf{u}}$ must be updated as the region evolves.

3. Segmentation of Shape-Tailored Descriptors

To illustrate the use of Shape-Tailored Descriptors in segmentation, we incorporate the descriptors into the Mumford-Shah energy [29], and then use the results of the previous section to compute its gradient.

Let $I : \Omega \to \mathbb{R}^k$ be the image, and $\mathbf{J} : \Omega \to \mathbb{R}^M$ be the vector of channels computed from I. We assume that the region R that we wish to segment and the background $R^c = \Omega \backslash R$ each consist of Shape-Tailored Descriptors that are mostly constant within neighborhoods of R and R^c following the Mumford-Shah model. We denote by $\mathbf{u} : R \to \mathbb{R}^M$ (resp., $\mathbf{v} : R^c \to \mathbb{R}^M$) the Shape-Tailored Descriptor in region R (resp., R^c) computed from $\mathbf{J}$. Note that $\mathbf{u}$ and $\mathbf{v}$ are both computed from $\mathbf{J}$ at the same scales α_i. The piecewise smooth Mumford-Shah [29, 41, 40] applied to $\mathbf{u}$ and $\mathbf{v}$ is

$$E(\mathbf{a}_i, \mathbf{a}_o, R) = \int_R (|\mathbf{u}(x) - \mathbf{a}_i(x)|^2 + \beta|D\mathbf{a}_i(x)|^2)\,\mathrm{d}x + \int_{R^c} (|\mathbf{v}(x) - \mathbf{a}_o(x)|^2 + \beta|D\mathbf{a}_o(x)|^2)\,\mathrm{d}x + \gamma L, \quad (9)$$

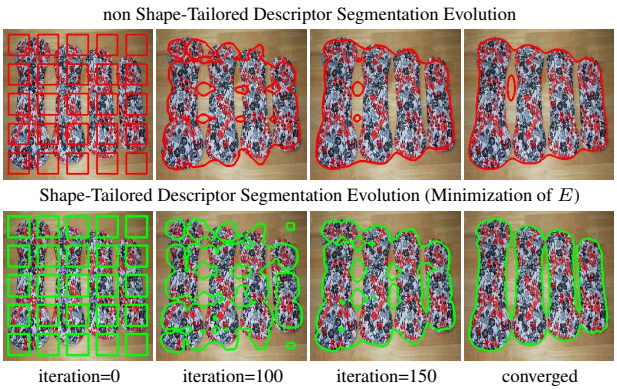

Figure 2. [Top]: non shape-tailored (traditional) local descriptors segmented with Chan-Vese. [Bottom]: segmentation of Shape-Tailored Descriptors with the piecewise constant model.

where $\mathbf{a}_i : R \to \mathbb{R}^M$ and $\mathbf{a}_o : R^c \to \mathbb{R}^M$ are functions that vary smoothly within their respective regions. In other words, they are roughly constant within local neighborhoods of their respective regions. Note that D indicates the Jacobian, and the two terms involving D enforce a smoothness penalty on $\mathbf{a}_i$ and $\mathbf{a}_o$. $\beta > 0$ controls the size of the neighborhoods for which the descriptors are assumed constant. $\beta \to \infty$ implies the whole region is assumed to have a constant descriptor (as in the simplified piecewise constant Mumford-Shah or Chan-Vese model [8]). Smaller β assumes that descriptors are constant within smaller neighborhoods. The functions $\mathbf{a}_i, \mathbf{a}_o$ are also solved as part of the optimization problem. Regularity of the region boundary is induced by the penalty on the length L of ∂R, where $\gamma > 0$.

We use alternating minimization in R and $\mathbf{a}_i, \mathbf{a}_o$. One can optimize for $\mathbf{a}_i$ and $\mathbf{a}_o$ given $\mathbf{u}$, $\mathbf{v}$ and R to find

$$\begin{cases} \mathbf{a}_i(x) - \beta\Delta\mathbf{a}_i(x) = \mathbf{u}(x) & x \in R \\ \mathbf{a}_o(x) - \beta\Delta\mathbf{a}_o(x) = \mathbf{v}(x) & x \in R^c \end{cases}. \quad (10)$$

Optimization in the region is performed using gradient descent, and the gradient can be computed using results of the previous section:

$$\nabla E = (|\mathbf{u} - \mathbf{a}_i|^2 - |\mathbf{v} - \mathbf{a}_o|^2 + \beta|D\mathbf{a}_i|^2 - \beta|D\mathbf{a}_o|^2)N + 2(\mathbb{I}_d[R, \mathbf{u}, \mathbf{u} - \mathbf{a}_i, \mathbf{0}] + \mathbb{I}_d[R^c, \mathbf{v}, \mathbf{v} - \mathbf{a}_o, \mathbf{0}]). \quad (11)$$

Figure 2 shows the gradient descent of E to segment a sample texture for the case that $\mathbf{a}_i, \mathbf{a}_o$ are assumed constant, i.e., the Chan-Vese model. To illustrate the motivation for segmentation with Shape-Tailored Descriptors, we show comparison to non-shape tailored descriptors (choosing the full image domain Ω to compute descriptors by solving (1) once on Ω, and using the standard Chan-Vese algorithm to segment these descriptors).

4. Numerical Implementation

We use level set methods [31] to implement the gradient descent of E. Discretization follows the standard schemes of level sets. Let Ψ be the level set function, F be the normal component of the gradient of energy ∇E, $\Delta t > 0$ be the step size, and t the iteration number. Steps 2-5 below are iterated until convergence of Ψ:

1. Initialize Ψ_0, $R_0 = \{\Psi_0 < 0\}$, $R_0^c = \Omega \backslash R_0$.
2. Solve for the Shape-Tailored Descriptors $\mathbf{u}_t : R_t \rightarrow \mathbb{R}^M$, $\mathbf{v}_t : R_t^c \rightarrow \mathbb{R}^M$ by solving (2) using an iterative scheme initialized with the Shape-Tailored Descriptors from the previous iteration $(\mathbf{u}_{t-1}, \mathbf{v}_{t-1}) : \Omega \rightarrow \mathbb{R}$ (zero for $t = 0$).
3. Solve for $\mathbf{a}_{i,t} : R_t \rightarrow \mathbb{R}^M$, $\mathbf{a}_{o,t} : R_t^c \rightarrow \mathbb{R}^M$ by solving (10) using an iterative scheme with initialization $\mathbf{a}_{i,t-1}, \mathbf{a}_{o,t-1}$. For the piecewise constant model, $\mathbf{a}_{i,t}$ and $\mathbf{a}_{o,t}$ are the averages of $\mathbf{u}_t$ and $\mathbf{v}_t$, respectively.
4. Solve for the "hat" descriptors $\hat{\mathbf{u}}_t : R_t \rightarrow \mathbb{R}^M$, $\hat{\mathbf{v}}_t : R_t^c \rightarrow \mathbb{R}^M$ by solving (7) (with the corresponding forcing and boundary functions determined by the arguments of $\mathbb{I}_d$ in (11)) using an iterative scheme with initialization $(\hat{\mathbf{u}}_{t-1}, \hat{\mathbf{v}}_{t-1})$.
5. Solve for F using (11). Then $\Psi_t = \Psi_{t-1} - \Delta t F |\nabla \Psi_{t-1}|$, and $R_t = \{\Psi_t < 0\}$, $R_t^c = \Omega \backslash R_t$.

The multigrid algorithm is used to solve for $\mathbf{u}_t$, $\mathbf{v}_t$, $\mathbf{a}_{i,t}$, $\mathbf{a}_{o,t}$, $\hat{\mathbf{u}}_t$, and $\hat{\mathbf{v}}_t$. After the first iteration, the update of these descriptors is fast since the solution changes only slightly between $t-1$ and t. Details of the numerical scheme is left to Supplementary Materials.

Updates for each of the components of $\mathbf{u}_t, \mathbf{v}_t$ can be done in parallel as the components are independent. Similarly for $\mathbf{a}_{i,t}, \mathbf{a}_{o,t}$ and $\hat{\mathbf{u}}_t, \hat{\mathbf{v}}_t$. Using an 12 core processor, our implementation to minimize E on a 1024×1024 image roughly takes 18 seconds for the piecewise constant model. This is with a box tessellation initialization, and the number of descriptor components is $M = 55$.

5. Experiments

The first set of experiments tests the ability of Shape-Tailored Descriptors to discriminate a variety of real-world textures. To this end, we compare Shape-Tailored Descriptors to a variety of descriptors for segmenting textured images based on the piecewise constant model. We compare on both a standard synthetic dataset and then on a dataset of real world images. The second set of experiments shows sample application of Shape-Tailored Descriptors to the problem of disocclusions in object tracking where objects consist of multiple textured regions. We thus use the piecewise smooth model. This shows that a state-of-the-art method in object tracking can be improved using Shape-Tailored Descriptors.

5.1. Robustness to Scale

Before we proceed to the main set of experiments, we show that Shape-Tailored Descriptors (STLD) are more robust to choices of scales α_i than the non shape-tailored descriptor (non-STLD). The scales control the locality of image data in the computation of $\mathbf{u}(x)$. Small α_i aggregate in small neighborhoods, and larger α_i aggregates in larger neighborhoods. Note that non-STLD is the solution of (2) on the whole domain of the image $R = \Omega$. non-STLD are computed before segmentation, and never updated.

We experiment on the Brodatz texture dataset (see details in the next sub-section). These images contain two textures. We choose five scales $\alpha_0 + (10, 20, 30, 40, 50)$ where α_0 is varied. The scales are based on a 256×256 image size, and the α_i's are multiplied by a factor of $(s/256)^2$ where s is the size of the smallest dimension. Segmentation is performed on both STLD and non-STLD using the piecewise constant model. A typical result is shown in the left of Fig. 3. A typical profile versus scale is shown on the right of Fig. 3. non-STLD with small α_0 gives the least accurate results. As α_0 increases, the results improve until the "right-scale" is chosen, and then the results degrade. This behavior is expected since large neighborhoods mix data from different textured regions. STLD retains the highest accuracy over many scales, and degrades slower with increasing scale.

The maximum scale should be chosen based on the size of the texton. In our experiments in the next sub-sections, we choose α_0 from a training set by creating a profile similar to Fig. 3. From experiments, 5 scales is a good tradeoff between accuracy and computational cost.

5.2. Performance of STLD in Segmentation

We test the performance of our new STLD by testing its ability to discriminate textures on two datasets, and then compare to other descriptors. Code and datasets will be available [1].

Datasets: The first dataset is a synthetic data set. It consists of images constructed from the textured images in the Brodatz dataset. Each is composed of two different textures. One texture is used as background and the other texture is masked with a shape from the MPEG 7 shape dataset and used as the foreground. The dataset consists of 50 images (5 different masks times 10 different foreground/background pairs). The second dataset consists of images obtained from Flickr that have two dominant textures. A variety of real textures (man-made and natural) have been chosen with common nuisances (e.g., small deformations of the domain,

[1] https://site.kaust.edu.sa/ac/frg/vision

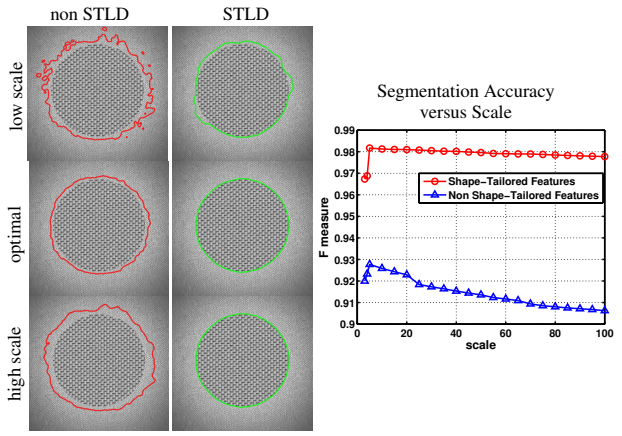

Figure 3. **Robustness of Shape-Tailored Features to the Choice of Scale.** [Left]: a synthetic image is segmented using low, optimum and large scales for non-STLD and STLD. [Right]: the accuracy of segmentation as the scale of the descriptor is increased. Larger F-measure indicates a more accurate segmentation.

some illumination variation). The size of the dataset is 256 images. We have hand segmented these images to facilitate quantitative comparison.

Methods Compared: We compare STLD to various other recent descriptors that are used for texture segmentation. Descriptors include simple global means used in Chan-Vese [8], global histograms (*Global Hist* [28]), local means (LAC [23]), more advanced descriptors based on local histograms in predefined neighborhood sizes (*Hist* [30]), SIFT descriptors (*SIFT*), the entropy profile (*Entropy* [17]), and non-STLD. For methods that can be formulated with convex relaxations, we use the segmentation based on global convex methods [5], which are more robust than gradient descent. This does not include our method, which uses gradient descent. We also compare to the hierarchical segmentation approach (*gPb* [1]). Note that gPb is not a descriptor, but uses several descriptors (e.g., Gabor filtering, and local histograms) to build a segmentation after edge detection. It is also for more general image segmentation, which is not the goal of our work, but we compare to it since it uses several descriptors. We also compare to [19] (CB), a recent texture segmentation method build on gPb, but using different edge detection.

Parameters: For all the methods, the training images were used to obtain the best regularity parameter γ, and that same parameter was used for the rest of the images. For STLD, the scales $\alpha = (5 + (10, 20, 30, 40, 50)) \times (s/256)^2$ where s is the size of the image, and $\theta = 0, \pi/8, \ldots, 7\pi/8$ are kept fixed on the whole datasets. All methods that require initialization are initialized with a box tessellation pattern that is standard in these types of methods.

Discussion of Qualitative Results: Figure 4 shows sample visualizations of results on the Brodatz dataset. Figure 5 shows sample results on our Real Texture dataset. Results are shown only for the top performing methods tested, and ground truth is displayed. Refer to Supplementary Material for more visualizations. STLD consistently performs well, clearly performing better than or at least as good as other methods. One can see that the boundaries are more accurate for STLD than non-STLD, and in many cases, the smoothing of data across textured regions also leads to more severe errors beyond overshooting the boundaries. The other region-based methods many times cannot capture the intrinsic texture differences on the datasets. The edge-based segmentation approach of gPb and CB works well detecting brightness edges, but in many cases does not detect texture boundaries. This maybe because sometimes texture boundaries are faint edges, and many times gPb and CB detect edges inside textons.

Discussion of Quantitative Results: Table 1 shows quantitative evaluation. We evaluate the algorithms using the evaluation protocol developed in [1]. The algorithms are evaluated both in terms of boundary and region accuracy by comparing to ground truth. For all metrics (except variation of information), a higher value indicates better fit to ground truth. ODS and OIS are the best values of results of the algorithm tuned with respect to a threshold on the entire dataset (ODS) and each image individually (OIS), and the difference applies only to gPb and CB. Our method out-performs all methods on all metrics.

5.3. Application of STLD to Disocclusions

We now show application of STLD to the problem of disocclusion detection in object tracking. One can track objects in a video by propagating an initial segmentation across frames, but two difficulties are self-occlusions and disocclusions of the object. Recently, [43] addressed the problem of self-occlusions and removed them from the segmentation propagation. This propagation and self-occlusion removal step does not obtain the full object segmentation since there may be parts of the object that become disoccluded. [43] detects disocclusions by comparing pixel intensities outside the propagated segmentation to local color histograms of the propagated segmentation. Pixels that match the local distributions are classified as disocclusion and included as part of the object segmentation. Our descriptors are more descriptive than local color histograms and are thus able to deal with more challenging object appearances, especially textured objects. Thus, we now use STLD to perform the disocclusion detection by segmentation of STLD based on the piecewise smooth Mumford-Shah. This is initialized with the propagation of the segmentation from the previous frame based on [43]. This detects as disocclusions those pixels that have similar STLDs locally to the segmentation propagation. Note that a piecewise constant STLD model of two regions is not adequate since

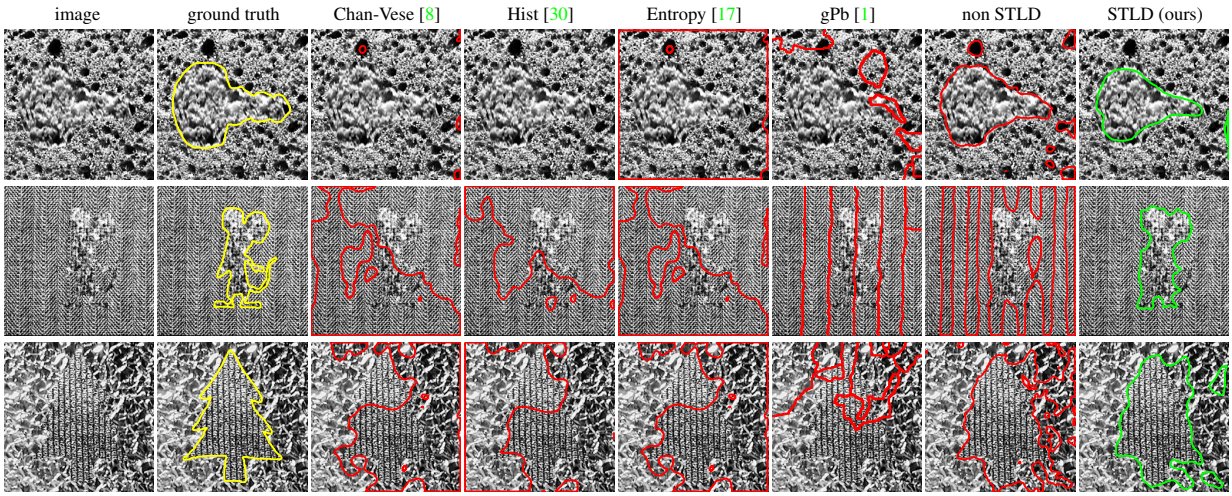

Figure 4. Sample Results on the Synthetic (Brodatz) Texture Dataset.

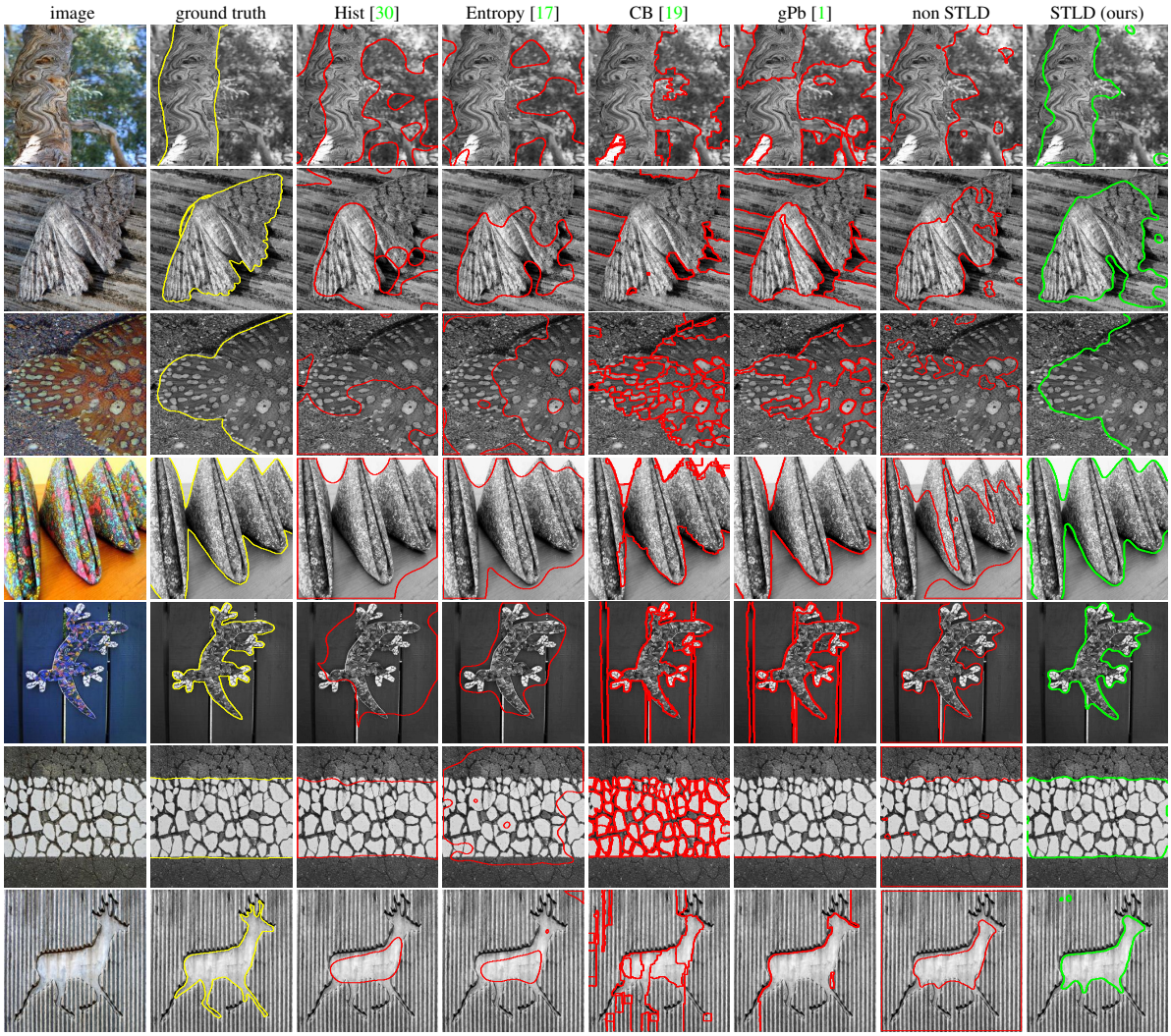

Figure 5. Sample Results on the Real Texture Dataset. Segmentation boundaries are displayed for various methods.

Brodatz Synthetic Dataset

	Contour		Region metrics					
	F-meas.		GT-cov.		Rand. Index		Var. Info.	
	ODS	OIS	ODS	OIS	ODS	OIS	ODS	OIS
STLD	**0.30**	**0.30**	**0.81**	**0.81**	**0.81**	**0.81**	**0.88**	**0.88**
non STLD	**0.28**	**0.28**	**0.78**	**0.78**	**0.77**	**0.77**	**0.98**	**0.98**
gPb [1]	0.20	0.20	0.56	0.56	0.57	0.57	1.17	1.17
SIFT	0.10	0.11	0.66	0.66	0.66	0.66	1.20	1.20
Entropy [17]	0.09	0.09	0.61	0.61	0.61	0.61	1.17	1.17
Hist-5 [30]	0.12	0.12	0.56	0.56	0.63	0.63	1.09	1.09
Hist-10 [30]	0.11	0.11	0.60	0.60	0.64	0.64	1.01	1.01
Chan-Vese [8]	0.09	0.09	0.61	0.61	0.61	0.61	1.17	1.17
LAC [23]	0.07	0.07	0.66	0.66	0.68	0.68	1.16	1.16
Global Hist [28]	0.10	0.10	0.38	0.38	0.52	0.52	2.41	2.41

Real Texture Dataset

	Contour		Region metrics					
	F-meas.		GT-cov.		Rand. Index		Var. Info.	
	ODS	OIS	ODS	OIS	ODS	OIS	ODS	OIS
STLD	**0.58**	**0.58**	**0.87**	**0.87**	**0.87**	**0.87**	**0.59**	**0.59**
non-STLD	0.17	0.17	**0.81**	0.81	**0.82**	0.82	**0.77**	0.77
gPb [1]	**0.50**	**0.54**	0.74	**0.84**	0.78	**0.86**	0.80	**0.65**
CB [19]	0.48	0.52	0.64	0.70	0.66	0.75	0.89	0.78
SIFT	0.10	0.10	0.55	0.55	0.59	0.59	1.44	1.44
Entropy [17]	0.08	0.08	0.74	0.74	0.75	0.75	0.95	0.95
Hist-5 [30]	0.14	0.14	0.66	0.66	0.70	0.70	1.18	1.18
Hist-10 [30]	0.13	0.13	0.66	0.66	0.70	0.70	1.19	1.19
Chan-Vese [8]	0.14	0.14	0.71	0.71	0.73	0.73	1.04	1.04
LAC [23]	0.09	0.09	0.55	0.55	0.58	0.58	1.41	1.41
Global Hist [28]	0.12	0.12	0.65	0.65	0.67	0.67	1.12	1.12

Table 1. **Summary of Results on Texture Segmentation Datasets**. Algorithms are evaluated using contour and region metrics (see text for details). Higher F-measure for the contour metric, ground truth covering (GT-cov), and rand index indicate better fit to the ground truth, and lower variation of information (Var. Info) indicates a better fit to ground truth. Bold red indicate best results and bold black indicates second-best results.

	Cheetah	CowFish	Turtle	WG Fish
Occlusion Tracker [43]	0.222	0.658	0.493	0.705
STLD	**0.937**	**0.929**	**0.958**	**0.909**

Table 2. Quantitative Evaluation of Object Tracking Results. Ground-Truth covering is used to evaluate results (higher means better fit to ground truth).

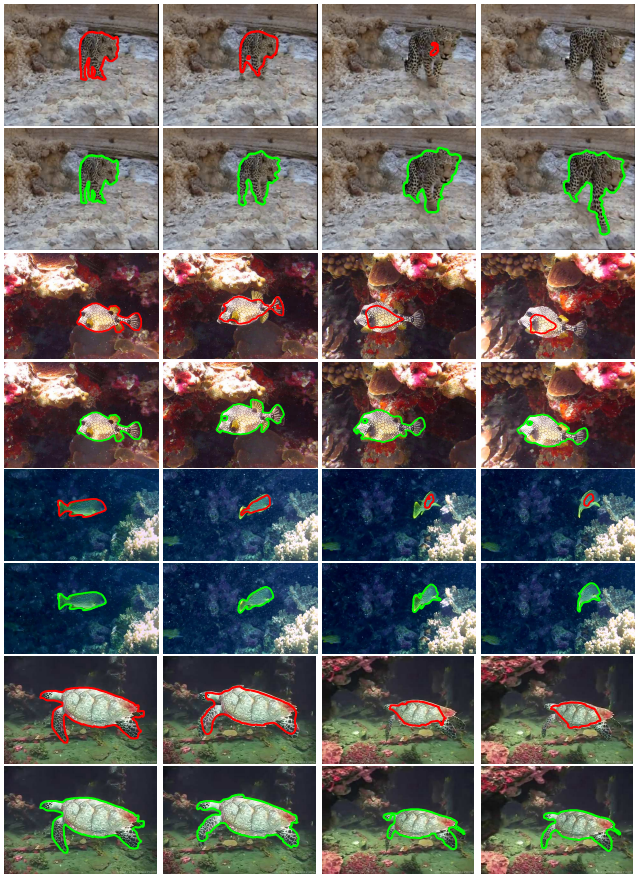

Figure 6. Results on Textured Object Tracking. [Top]: Results of a state-of-the-art method [43] (red). The method fails early since the disocclusion detection is based on local color histogram descriptors, which fail to capture textures. [Bottom]: Results of [43] by replacing local color histograms in disocclusion detection with STLD based on piecewise smooth Mumford-Shah.

the object and background consist of multiple textures.

Results on four challenging videos are shown in Figure 6 and compared against [43]. Table 2 gives quantitative analysis. The videos contain objects with multiple textures, and the backgrounds also consist of multiple textures. In all sequences, $\beta = 10$, the scales α_i are chosen the same as in the previous section. Shape-Tailored Descriptors capture the textured object of interest accurately. [43] fails to capture disoccluded regions that are textured. These errors, slight at first as only small parts are disoccluded between frames, are then propagated forward and the method fails to segment the object accurately. Only 4 out of 50 frames are shown; videos are in Supplementary Materials.

6. Conclusion

We have introduced *Shape-Tailored Local Descriptors*, dense descriptors of oriented gradients that are tailored to arbitrarily shaped regions by the use of shape-dependent scale spaces. Existing local descriptors that are based on oriented gradients aggregate data from neighborhoods that could cross texture boundaries. We have shown that STLD leads to more accurate segmentation of textures than non-STLD and other common descriptors. We have shown this through sample application of these descriptors in a Mumford-Shah segmentation framework. We also showed application of these descriptors in object tracking, specifically addressing the issues of disocclusions. This improves a state-of-the-art object tracking technique. Although the STLD proved useful, much work remains in the design of descriptors for segmentation, in particular to address issues of shading and shadows, and scale-invariance.

Acknowledgements

This research was funded by KAUST Baseline funding, and KAUST OCRF-2014-CRG3-62140401. Anthony Yezzi was supported by NSF CCF-1347191.

References

[1] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik. Contour detection and hierarchical image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(5):898–916, 2011. 2, 6, 7, 8

[2] G. Aubert, M. Barlaud, O. Faugeras, and S. Jehan-Besson. Image segmentation using active contours: Calculus of variations or shape gradients? *SIAM Journal on Applied Mathematics*, 63(6):2128–2154, 2003. 2

[3] S. P. Awate, T. Tasdizen, and R. T. Whitaker. Unsupervised texture segmentation with nonparametric neighborhood statistics. In *Computer Vision–ECCV 2006*, pages 494–507. Springer, 2006. 2

[4] S. Boltz, F. Nielsen, and S. Soatto. Texture regimes for entropy-based multiscale image analysis. In *Computer Vision–ECCV 2010*, pages 692–705. Springer, 2010. 2

[5] X. Bresson, S. Esedolu, P. Vandergheynst, J.-P. Thiran, and S. Osher. Fast global minimization of the active contour/snake model. *Journal of Mathematical Imaging and vision*, 28(2):151–167, 2007. 2, 6

[6] T. Brox and D. Cremers. On local region models and a statistical interpretation of the piecewise smooth mumford-shah functional. *International journal of computer vision*, 84(2):184–193, 2009. 2

[7] T. F. Chan, S. Esedoglu, and M. Nikolova. Algorithms for finding global minimizers of image segmentation and denoising models. *SIAM Journal on Applied Mathematics*, 66(5):1632–1648, 2006. 2

[8] T. F. Chan and L. A. Vese. Active contours without edges. *Image processing, IEEE transactions on*, 10(2):266–277, 2001. 2, 4, 6, 7, 8

[9] D. Cremers, M. Rousson, and R. Deriche. A review of statistical approaches to level set segmentation: integrating color, texture, motion and shape. *International journal of computer vision*, 72(2):195–215, 2007. 2

[10] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886–893. IEEE, 2005. 1

[11] C. Darolti, A. Mertins, C. Bodensteiner, and U. G. Hofmann. Local region descriptors for active contours evolution. *Image Processing, IEEE Transactions on*, 17(12):2275–2288, 2008. 2

[12] M. C. Delfour and J.-P. Zolésio. *Shapes and geometries: metrics, analysis, differential calculus, and optimization*, volume 22. Siam, 2011. 2

[13] L. C. Evans. Partial differential equations. 1998. 3

[14] M. Galun, E. Sharon, R. Basri, and A. Brandt. Texture segmentation by multiscale aggregation of filter responses and shape elements. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 716–723. IEEE, 2003. 2

[15] A. Herbulot, S. Jehan-Besson, S. Duffner, M. Barlaud, and G. Aubert. Segmentation of vectorial image features using shape gradients and information measures. *Journal of Mathematical Imaging and Vision*, 25(3):365–386, 2006. 2

[16] B.-W. Hong, K. Ni, and S. Soatto. Entropy-scale profiles for texture segmentation. In *Scale Space and Variational Methods in Computer Vision*, pages 243–254. Springer, 2012. 2

[17] B.-W. Hong, S. Soatto, K. Ni, and T. Chan. The scale of a texture and its application to segmentation. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008. 2, 6, 7, 8

[18] N. Houhou, J. Thiran, and X. Bresson. Fast texture segmentation model based on the shape operator and active contour. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008. 2

[19] P. Isola, D. Zoran, D. Krishnan, and E. H. Adelson. Crisp boundary detection using pointwise mutual information. In *Computer Vision–ECCV 2014*, pages 799–814. Springer, 2014. 2, 6, 7, 8

[20] B. Julesz. Textons, the elements of texture perception, and their interactions. *Nature*, 290(5802):91–97, 1981. 1

[21] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International journal of computer vision*, 1(4):321–331, 1988. 2

[22] J. Kim, J. W. Fisher, A. Yezzi, M. Çetin, and A. S. Willsky. A nonparametric statistical method for image segmentation using information theory and curve evolution. *Image Processing, IEEE Transactions on*, 14(10):1486–1502, 2005. 2

[23] S. Lankton and A. Tannenbaum. Localizing region-based active contours. *Image Processing, IEEE Transactions on*, 17(11):2029–2039, 2008. 2, 6, 8

[24] T. S. Lee, D. Mumford, and A. Yuille. Texture segmentation by minimizing vector-valued energy functionals: The coupled-membrane model. In *Computer VisionECCV'92*, pages 165–173. Springer, 1992. 2

[25] T. Lindeberg. *Scale-space theory in computer vision*. Springer, 1993. 2

[26] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004. 1

[27] J. Malik and P. Perona. Preattentive texture discrimination with early vision mechanisms. *JOSA A*, 7(5):923–932, 1990. 1, 2

[28] O. Michailovich, Y. Rathi, and A. Tannenbaum. Image segmentation using active contours driven by the bhattacharyya gradient flow. *Image Processing, IEEE Transactions on*, 16(11):2787–2801, 2007. 2, 6, 8

[29] D. Mumford and J. Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on pure and applied mathematics*, 42(5):577–685, 1989. 1, 2, 4

[30] K. Ni, X. Bresson, T. Chan, and S. Esedoglu. Local histogram based segmentation using the wasserstein distance. *International Journal of Computer Vision*, 84(1):97–111, 2009. 2, 6, 7, 8

[31] S. Osher and J. A. Sethian. Fronts propagating with curvature-dependent speed: algorithms based on hamilton-jacobi formulations. *Journal of computational physics*, 79(1):12–49, 1988. 5

[32] N. Paragios and R. Deriche. Geodesic active regions and level set methods for supervised texture segmentation. *Inter-*

national Journal of Computer Vision, 46(3):223–247, 2002. 2

[33] G. Peyré, J. Fadili, and J. Rabin. Wasserstein active contours. In *Image Processing (ICIP), 2012 19th IEEE International Conference on*, pages 2541–2544. IEEE, 2012. 2

[34] T. Pock, D. Cremers, H. Bischof, and A. Chambolle. An algorithm for minimizing the mumford-shah functional. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 1133–1140. IEEE, 2009. 2

[35] M. Rousson, T. Brox, and R. Deriche. Active unsupervised texture segmentation on a diffusion based feature space. In *Computer vision and pattern recognition, 2003. Proceedings. 2003 IEEE computer society conference on*, volume 2, pages II–699. IEEE, 2003. 2

[36] C. Sagiv, N. A. Sochen, and Y. Y. Zeevi. Integrated active contours for texture segmentation. *Image Processing, IEEE Transactions on*, 15(6):1633–1646, 2006. 2

[37] L. Sifre and S. Mallat. Rotation, scaling and deformation invariant scattering for texture discrimination. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 1233–1240. IEEE, 2013. 1

[38] S. Todorovic and N. Ahuja. Texel-based texture segmentation. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 841–848. IEEE, 2009. 2

[39] E. Tola, V. Lepetit, and P. Fua. Daisy: An efficient dense descriptor applied to wide-baseline stereo. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(5):815–830, 2010. 1

[40] A. Tsai, A. Yezzi Jr, and A. S. Willsky. Curve evolution implementation of the mumford-shah functional for image segmentation, denoising, interpolation, and magnification. *Image Processing, IEEE Transactions on*, 10(8):1169–1186, 2001. 4

[41] L. A. Vese and T. F. Chan. A multiphase level set framework for image segmentation using the mumford and shah model. *International journal of computer vision*, 50(3):271–293, 2002. 4

[42] A. Y. Yang, J. Wright, Y. Ma, and S. S. Sastry. Unsupervised segmentation of natural images via lossy data compression. *Computer Vision and Image Understanding*, 110(2):212–225, 2008. 2

[43] Y. Yang and G. Sundaramoorthi. Modeling self-occlusions in dynamic shape and appearance tracking. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 201–208. IEEE, 2013. 2, 6, 8

[44] A. Yezzi Jr, A. Tsai, and A. Willsky. A statistical approach to snakes for bimodal and trimodal imagery. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 2, pages 898–903. IEEE, 1999. 2

[45] S. C. Zhu and A. Yuille. Region competition: Unifying snakes, region growing, and bayes/mdl for multiband image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 18(9):884–900, 1996. 2