Segmentation of Scanning Tunneling Microscopy Images Using Variational Methods and Empirical Wavelets

Kevin Bui · Jacob Fauman · David Kes · Leticia Torres Mandiola · Adina Ciomaga · Ricardo Salazar · Andrea L. Bertozzi · Jérôme Gilles · Andrew I. Guttentag · Paul S. Weiss

April 17, 2018

K. Bui
Department of Industrial Engineering and Management Sciences, Northwestern University, 2145 Sheridan Road, Evanston, IL 60208, United States. E-mail: kbui1993@gmail.com

J. Fauman
Department of Applied Mathematics, 526 UCB, ECOT 225, University of Colorado Boulder, Boulder, Colorado 80309-0526, United States. E-mail: jacob.fauman@colorado.edu

D. Kes
College of Arts and Sciences, University of San Francisco, USF Downtown Campus, 101 Howard Street, San Francisco, CA 94105, United States. E-mail: ddkes@dons.usfca.edu

L. Torres Mandiola
Department of Mathematics, South Kensington Campus, Imperial College London, London, United Kingdom SW7 2AZ. E-mail: leticia.torres-mandiola16@imperial.co.uk

A. Ciomaga
Univ. Paris Diderot, Sorbonne Paris Cité, Laboratoire Jacques-Louis Lions, UMR 7598, UPMC, CNRS, F-75205 Paris, France. E-mail: adina@ljll.univ-paris-diderot.fr

R. Salazar
Department of Mathematics, University of California, Los Angeles, 520 Portola Plaza, Los Angeles, CA 90095-1555, United States. E-mail: rsalazar@math.ucla.edu

A.L. Bertozzi
Department of Mathematics, University of California, Los Angeles, 520 Portola Plaza, Los Angeles, CA 90095-1555, United States. E-mail: bertozzi@math.ucla.edu

J. Gilles
Department of Mathematics & Statistics, San Diego State University, 5500 Campanile Dr, San Diego, CA 92181-7720, United States. E-mail: jgilles@mail.sdsu.edu

A.I. Guttentag
Department of Chemistry and Biochemistry, University of California, Los Angeles, Los Angeles, California 90095, United States.

California NanoSystems Institute, University of California, Los Angeles, Los Angeles, California 90095, United States. E-mail: aguttentag@ucla.edu

P.S. Weiss
Department of Chemistry and Biochemistry, University of California, Los Angeles, Los Angeles, California 90095, United States.

Department of Materials Science and Engineering, University of California, Los Angeles, Los Angeles, California 90095, United States. · California NanoSystems Institute, University of California, Los Angeles, Los Angeles, California 90095, United States. E-mail: psw@cnsi.ucla.edu

Abstract In the fields of nanoscience and nanotechnology, it is important to be able to functionalize surfaces chemically for a wide variety of applications. Scanning tunneling microscopes (STMs) are important instruments in this area used to measure the surface structure and chemistry with better than molecular resolution. Self-assembly is frequently used to create monolayers that redefine the surface chemistry in just a single-molecule-thick layer (Nuzzo and Allara 1983; Smith et al 2004; Love et al 2005). Indeed, STM images reveal rich information about the structure of self-assembled monolayers since they convey chemical and physical properties of the studied material.

In order to assist in and to enhance the analysis of STM and other images (Thomas et al 2015, 2016), we propose and demonstrate an image-processing framework that produces two image segmentations: one is based on intensities (apparent heights in STM images) and the other is based on textural patterns. The proposed framework begins with a cartoon+texture decomposition, which separates an image into its cartoon and texture components. Afterward, the cartoon image is segmented by a modified multiphase version of the local Chan-Vese model (Wang et al 2010), while the texture image is segmented by a combination of 2D empirical wavelet transform and a clustering algorithm. Overall, our proposed framework contains several new features, specifically in presenting a new application of cartoon+texture decomposition and of the empirical wavelet transforms and in developing a specialized framework to segment STM images and other data. To demonstrate the potential of our approach, we apply it to actual STM images of cyanide monolayers on Au$\{111\}$ and present their corresponding segmentation results.

Keywords Scanning Tunneling Microscopy · Segmentation · Chan-Vese · Empirical Wavelets · Textures

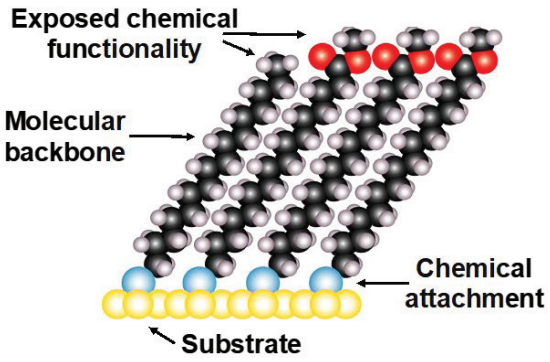


Fig. 1 In self-assembled monolayers, a single layer of molecules is chemically bound to a solid or liquid substrate. The wide range of substrates (*e.g.*, metals, semiconductors, insulators, glasses, superconductors, nanoparticles) that can be used call for complementary chemistries of attachment of the molecular layers. The exposed functional group at the ends of the molecules typically dominates the interactions of the substrate with the surrounding chemical, physical, and biological environment

1 Introduction

Self-assembled monolayers (SAMs) have been extensively studied and applied in nanoscience, nanotechnology, and beyond (Poirier 1997; Gooding et al 2003; Smith et al 2004; Love et al 2005). These SAMs are formed by molecules that have a head group (*e.g.*, sulfur, selenium, carboxylate, phosphonate) that is chemically bound to a substrate (*e.g.*, gold, silver, copper, platinum, germanium), and often form two-dimensional crystalline lattices (see Figure 1). The strong affinity between the head group and the substrate and the intermolecular interactions between the molecular backbones and tail groups lead to self-assembly of the monolayers via exposure of the substrate to the molecules in solution, in vapor, or by contact with a supporting structure such as a polymer stamp. The organization of the monolayer structure depends on the chemical properties and structures of the molecular monolayer components (Claridge et al 2013). Controlling the design of a SAM requires properly tuning the chemical and physical properties of the assembled molecules (Dameron et al 2005). As a result, control of basic parameters and external stimuli (*e.g.*, deposition conditions, temperature, electrochemical potential, and illumination) on SAMs has been examined to target specific assemblies for nanotechnology research and applications (Guttentag et al 2016a). One method of analyzing the chemical and physical properties of SAM is by examining molecular resolution images obtained by scanning tunneling microscopy.

In order to obtain a scanning tunneling microsopy image of a SAM, an atomically sharp conducting probe tip is brought within one or two atomic diameters of the surface of the sample so that electrons can tunnel from the surface to the tip. A voltage bias is applied between the two and the tip-sample separation is typically adjusted while scanning to maintain a constant tunneling current of electrons. Since the current is extremely sensitive to the tip-sample separation, better than atomic resolution is often obtained and apparent height differences across the surface are recorded, thereby acquiring nanoscale images with molecular features. The scanning procedure is shown in Figure 2 and examples of STM images of cyanide (CN) monolayers on Au$\{111\}$ are shown in Figure 3. These images show the varying textures and different apparent heights (displayed as intensities) as a result of the structure and chemical properties of the SAM. Partitioning the images according to the apparent heights and texture patterns would help facilitate the understanding and analyses of SAMs and other surfaces studied. We note that not only are the ordered regions important but also are the boundaries between them (Poirier 1997) since these domain boundaries determine access of other molecules to the substrate and can be used to isolate single molecules, or pairs, lines, or clusters of molecules (Bumm et al 1996; Kim et al 2011; Claridge et al 2013).

Here, we propose a novel framework to analyze STM images that produces segmentation based on intensities and segmentation based on texture features. The proposed method consists of three main steps. The first step performs a cartoon+texture decomposition of the STM image. Its cartoon and texture components are then analyzed separately. The cartoon image is the component containing only the edges or boundaries of homogeneous regions in the images and it is devoid of any oscillatory patterns. On the other hand, the texture image consists of oscillatory patterns. We propose two parallel steps: the cartoon component is segmented using a variant of the Chan-Vese model (Chan et al 2000) and the texture component is segmented by feeding some classifier with features based on empirical wavelets (Gilles 2013; Gilles et al 2014).

First, we propose a novel multiphase version of the local Chan-Vese model (Wang et al 2010) and develop an efficient algorithm based on the MBO scheme (Merriman et al 1992, 1994) to solve it. The proposed model creates a more accurate segmentation result that is robust against intensity inhomogeneities compared to the original multiphase Chan-Vese model (Vese and Chan 2002). Secondly, we improve the 2D empirical wavelet transform, specifically the empirical curvelet transform, so that it provides a filter bank of empirical curvelets extracting meaningful textural information. From the filter bank outputs, we design a texture feature matrix to feed a clustering algorithm in order to identify regions of different textural patterns on the STM image.

The paper outline is as follows. In Section 2, we describe the nonlinear decomposition algorithm we use in our approach to decompose an STM image into its cartoon and textures components. In Section 3, we propose a modified version of the local multiphase model and determine its diffuse interface approximation in order to develop an algorithm based on the MBO scheme to solve it. In Section 4, we review the empirical wavelet transform (particularly the empirical curvelet transform) and propose modifications

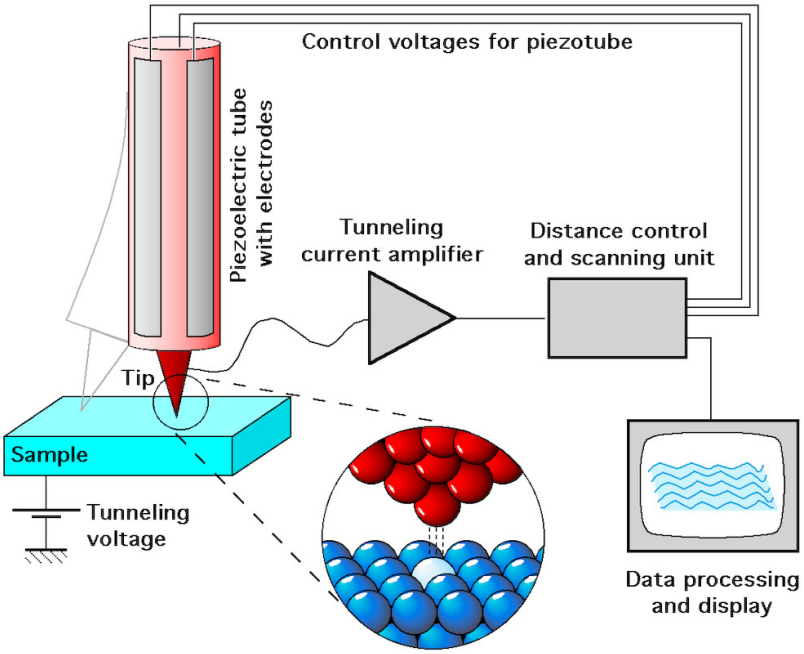


Fig. 2 To obtain scanning tunneling microscope images, constant current is held as the tip moves across the surface, experiencing voltage drops over bumps. Figure provided by Michael Schmid, TU Wien, at `https://commons.wikimedia.org/w/index.php?curid=180388`

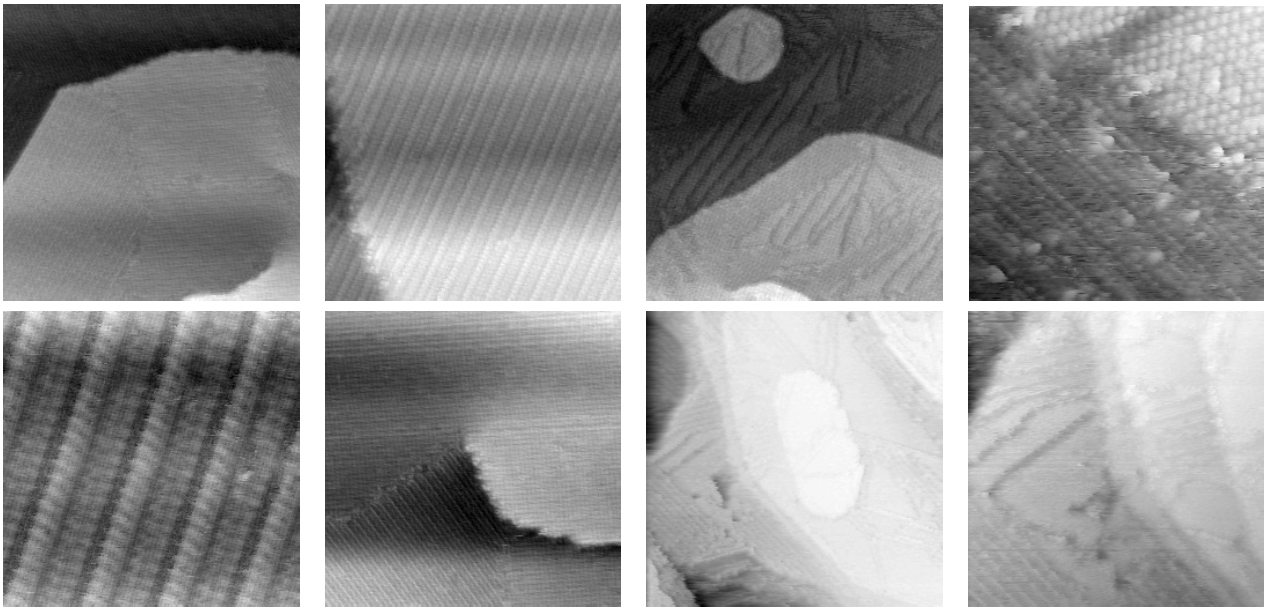


Fig. 3 Raw scanning tunneling microscope images of cyanide on Au$\{111\}$, reproduced from Guttentag et al (2016a) with permissions. Images copyright American Chemical Society

on how the curvelets' supports are detected. We then explain the design of the texture feature matrix based on the empirical curvelet coefficients. In Section 5, our proposed framework is applied to the images in Figure 3 in order to evaluate its efficacy in partitioning STM images by intensities and texture patterns. Finally, in Section 6, we summarize our results and discuss possible research directions to further improve the proposed framework.

2 The Cartoon+Texture Decomposition

As discussed in the introduction, the first step of our algorithm involves the decomposition of an image into its cartoon and textural components.

The cartoon+texture decomposition can be posed as an inverse problem and it consists of decomposing an image $f : \Omega \to \mathbb{R}$, where Ω is the image domain, into

$$f = u + v$$

where u and v are the cartoon and the texture components, respectively. The cartoon image carries broad information about the image and is usually modeled by a function of bounded variation (piecewise smooth with possibly a discontinuity set). The texture image contains oscillatory information and is thus usually modeled by oscillating functions.

One of the earliest variational models that inspired cartoon+texture decomposition is the Rudin-Osher-Fatemi total variation minimization model (Rudin et al 1992)

$$\inf_{\substack{u\in BV(\Omega)\\ v\in L^2(\Omega)}} \Big\{\sigma\|u\|_{TV} + \|v\|_{L^2}^2,\ f = u+v\Big\}, \tag{1}$$

where $\|u\|_{TV}$ is the total variation of u and σ is a tuning parameter that controls the regularization strength. The model was originally used for denoising purposes because of the functional spaces that u and v belong to. The function u belongs to the space of functions of bounded variations

$$BV(\Omega) = \left\{u \in L^1(\Omega) : \int_\Omega |\nabla u| < \infty\right\},$$

which penalizes oscillations such as noise and textures but allows for piecewise-smooth functions made of homogeneous regions with sharp boundaries. However, the decomposition is well-posed only in a multiresolution setup since image features can be considered as texture in one scale and cartoon at a different scale. One of the most popular models in cartoon+texture decomposition is the $TV-L^1$ model proposed by Chan and Esedoglu (2005), for which Chambolle (2004) provided a fast projection algorithm.

Although models, such as (1) and the above mentioned, are able to perform cartoon + texture decomposition, Meyer (2001) argued that the texture image extracted by these models does not fully characterize the oscillatory patterns of the original image. As a result, he proposed to replace the L^2 norm in (1) by weaker norms (associated to spaces of oscillatory distributions) in order to better capture the oscillatory patterns. In practice, some of these norms are difficult to compute. To remedy this drawback, Buades et al (2011) developed a nonlinear version of the linearized Meyer's model.

The linearized Meyer's model is

$$\inf_{\substack{u\in H^1(\Omega)\\ v\in H^{-1}(\Omega)}} \Big\{\sigma^4\|\nabla u\|_{L^2}^2 + \|v\|_{H^{-1}}^2,\ f = u+v\Big\}, \tag{2}$$

where $H^1(\Omega) = \{u \in L^2(\Omega) : \nabla u \in L^2(\Omega)\}$ and $H^{-1}(\Omega)$ is the dual space of the homogeneous version of $H^1(\Omega)$. They are defined in the Fourier domain by

$$H^1(\Omega)$$
$$= \{u \in L^2(\Omega) : \int_\Omega [1+(2\pi|\xi|)^2]|\mathscr{F}_2(u)(\xi)|^2 d\xi < \infty\}$$

and

$$H^{-1}(\Omega)$$
$$= \{v \in L^2(\Omega) : \int_\Omega [1+(2\pi|\xi|)^2]^{-1}|\mathscr{F}_2(v)(\xi)|^2 d\xi < \infty\},$$

where $\mathscr{F}_2(\cdot)$ is the 2D Fourier Transform. Minimizing the quadratic functional (2) yields the solution

$$u = L_\sigma * f \text{ with } \mathscr{F}_2(L_\sigma)(\xi) = \frac{1}{1+(2\pi\sigma|\xi|)^4},$$

where the convolution kernel is given in the Fourier domain. Since $\mathscr{F}_2(L_\sigma)(\xi)$ defines a low-pass filter, any frequency ξ that is significantly smaller than $\frac{1}{2\pi\sigma}$ is kept in u. Otherwise, it is kept in v. Thus, the cartoon + texture decomposition is performed by applying a low-pass/high-pass filter onto f,

$$(u,v) = (L_\sigma * f, (Id - L_\sigma) * f).$$

The drawback of the low-pass filter is that sharp edges are altered, while they should be preserved as much as possible since they are necessary for cartoon segmentation.

The nonlinear version of (2) relies on a classifier which determines whether a pixel of the original image belongs to the cartoon or the texture component (Buades et al 2011). The idea consists of measuring the rate of change of the local total variation (LTV) between the original image and its lowpass filtered version, defined by

$$\lambda_\sigma(x) = \frac{LTV_\sigma(f)(x) - LTV_\sigma(L_\sigma * f)(x)}{LTV_\sigma(f)(x)}, \tag{3}$$

where

$$LTV_\sigma(f)(x) = L_\sigma * |\nabla f|(x). \tag{4}$$

If a neighbourhood of a pixel x does not contain any textures, then f and $L_\sigma * f$ will be similar within the neighbourhood, so $\lambda_\sigma(x)$ is close to 0. If some textures are present within the neighborhood, then the total variation of the neighborhood in the filtered image will be smaller and $\lambda_\sigma(x)$ is close to 1. Therefore, the cartoon component u is computed as the weighted average between the original image f and the filtered image $L_\sigma * f$ depending on λ_σ, i.e.

$$u(x) = w\big(\lambda_\sigma(x)\big)(L_\sigma * f)(x) + \big(1 - w(\lambda_\sigma(x))\big)f(x),$$

with the soft threshold function $w : [0,1] \to [0,1]$ given by

$$w(x) = \begin{cases} 0 & \text{if} \quad x < a_1 \\ (x-a_1)/(a_2-a_1) & \text{if} \quad a_1 \le x \le a_2 \\ 1 & \text{if} \quad a_2 < x \end{cases},$$

where $0 < a_1 < a_2 < 1$. For our experiments in Section 5, as suggested by Buades et al (2011), we fixed $a_1 = 0.25$ and $a_2 = 0.50$. The texture component is easily obtained by computing the difference between the original image and its cartoon component. The algorithm for the nonlinear version is summarized in Algorithm 1.

In Figure 4, we present results for both linear and nonlinear decompositions for one of the images in Figure 3 to illustrate why choosing the nonlinear version is more interesting. Although both decompositions succeed in separating the image into its cartoon and texture components, as shown in Figures 4b and 4d, the nonlinear decomposition gives sharper edges in the cartoon component than does the linear decomposition. Since in general these edges do not correspond to textures associated with molecular orientations but rather with topographic transitions of the molecular layers, the nonlinear version is therefore preferable.

Algorithm 1: Cartoon+Texture Decomposition

Input: Original image f, parameter $\sigma > 0$

1: Compute the LTV reduction rate using (3) at each pixel.

2: Obtain the cartoon component:

$$u(x) \leftarrow w\big(\lambda_\sigma(x)\big)(L_\sigma * f)(x) + \big(1 - w(\lambda_\sigma(x))\big) f(x)$$

3: Obtain the texture component:

$$v(x) \leftarrow f(x) - u(x)$$

Output: Cartoon and texture components u, v

3 Segmentation of the Cartoon Image

The cartoon component of an STM image provides information about the stratification of scanned molecules. The lighter the gray intensity is, the greater the apparent height. Here, the largest variations are due to monoatomic steps in the substrate; a single layer of molecules is present everywhere on the surface. Identifying different gray level regions is therefore crucial for finding the topographic properties of the substrate and chemical layers.

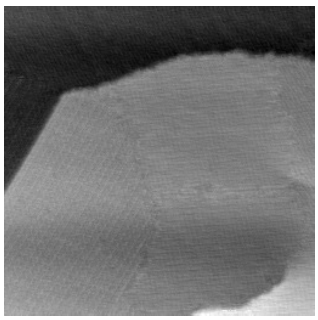
We propose to use a level-set approach based on the multiphase Chan-Vese (multiphase CV) model to perform the segmentation of the cartoon component. The Chan-Vese model is inspired by the pioneering work of Mumford and Shah (1989), who suggested performing image segmentation by solving the minimization problem

$$\inf_{u,\Gamma}\left\{\int_{\Omega\setminus\Gamma}|\nabla u|^2 + \lambda\int_\Omega (u-u_0)^2 + \mu\int_\Gamma ds\right\}, \tag{5}$$

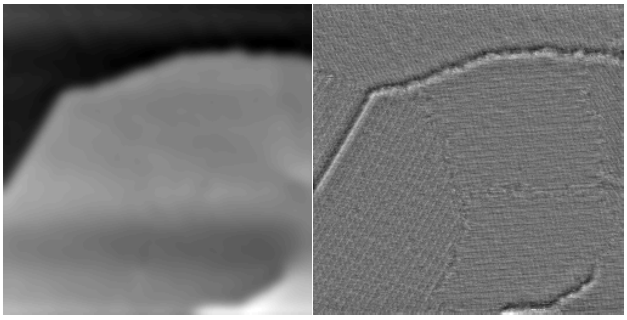
where $u_0 : \Omega \to \mathbb{R}$ is the cartoon image to be segmented and μ and λ are weighing parameters. The function u is a piecewise smooth approximation of u_0, which is allowed to have jumps across Γ – a closed subset of Ω given by a finite union of rectifiable curves.

Although several algorithms have been proposed to compute the solution (Ambrosio and Tortorelli 1990), its computation (Vese and Chan 1997; March 1992) is relatively complicated and numerically expensive. To overcome this drawback, simplifications of the energy functional (5) to piecewise constant functions have been proposed.

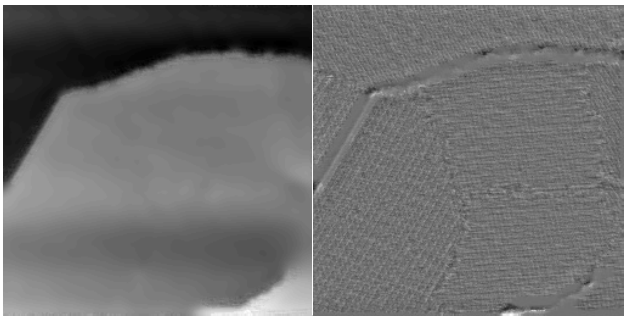
Based on the level-set approach (Osher and Sethian 1988), the multiphase CV segmentation model (Vese and Chan 2002) remains one of the most popular models. In this setup, the function u is allowed to have only four values c_1, c_2, c_3, and c_4, one for each of the four distinct regions. Furthermore, using two level-set functions ϕ_1 and ϕ_2, thus producing four phases $\{\phi_1 > 0, \phi_2 > 0\}, \{\phi_1 > 0, \phi_2 < 0\}, \{\phi_1 < 0, \phi_2 > 0\}$, and $\{\phi_1 < 0, \phi_2 > 0\}$, one can show it is sufficient to generate a partition of the domain Ω into regions of different intensities having triple junctions or T-junctions such that $\Gamma = \{\phi_1 = 0\} \cup \{\phi_2 = 0\}$ (See Vese and Chan (2002) for details).



(a) Original STM image



(b) Linear filtering: cartoon **(c)** Linear filtering: texture



(d) Nonlinear filtering: cartoon **(e)** Nonlinear filtering: texture

Fig. 4 Results of linear and nonlinear cartoon texture decompositions with $\sigma = 3$, reproduced from Guttentag et al (2016a) with permission. Image in (a) copyright American Chemical Society

Let $\mathbf{u} = (u_1, u_2)$ be a vector-valued function and $\mathbf{c} = (c_1, c_2, c_3, c_4)$ be a vector of constants. The fidelity term is defined as

$$\begin{aligned}\mathscr{E}_{fid}(\mathbf{c},\mathbf{u}) = &\int_\Omega (c_1-u_0)^2 u_1 u_2 \\ &+ \int_\Omega (c_2-u_0)^2 u_1 (1-u_2) \\ &+ \int_\Omega (c_3-u_0)^2 (1-u_1) u_2 \\ &+ \int_\Omega (c_4-u_0)^2 (1-u_1)(1-u_2).\end{aligned} \tag{6}$$

The perimeter term is defined by

$$\mathscr{E}_{per}(\mathbf{u}) = \int_\Omega |\nabla u_1| + \int_\Omega |\nabla u_2|. \tag{7}$$

Using the standard Heaviside function H with the notations $\Phi = (\phi_1, \phi_2)$ and $H(\Phi) = (H(\phi_1), H(\phi_2))$, the four-phase piecewise constant Mumford-Shah model can be written in terms of the level set functions ϕ_1 and ϕ_2 as

$$\mathscr{E}_{CV}(\mathbf{c}, H(\Phi)) = \lambda \mathscr{E}_{fid}(\mathbf{c}, H(\Phi)) + \mu \mathscr{E}_{per}(H(\Phi)). \tag{8}$$

3.1 Local Multiphase Chan-Vese Model

The model (8) is, however, not robust against illumination bias, such as shadows, or intensity inhomogeneities, such as weak edges, usually present in STM images. For example, if a region of an image is partially overlapped by a shadow, it might be segmented into two regions. Or, on the contrary, regions with inhomogeneous gray-level intensities, may not be segmented at all, even though they might provide valuable information.

Wang et al (2010) proposed a local term that can be added to (8) to counteract the lighting issues. We define, as before, a local term

$$\begin{aligned}\mathscr{E}_{loc}(\mathbf{d},\mathbf{u}) = &\int_\Omega (g_k * u_0 - u_0 - d_1)^2 u_1 u_2 \\ &+ \int_\Omega (g_k * u_0 - u_0 - d_2)^2 u_1 (1-u_2) \\ &+ \int_\Omega (g_k * u_0 - u_0 - d_3)^2 (1-u_1) u_2 \\ &+ \int_\Omega (g_k * u_0 - u_0 - d_4)^2 (1-u_1)(1-u_2),\end{aligned} \tag{9}$$

where g_k is a convolution kernel with $(k \times k)$-size window and $\mathbf{d} = (d_1, d_2, d_3, d_4)$ is a vector-valued function. The local multiphase Chan-Vese model (local MCV) requires minimizing the energy functional

$$\mathscr{E}_{CVloc}(\mathbf{c},\mathbf{d},H(\Phi)) = \mathscr{E}_{CV}(\mathbf{c},H(\Phi)) + \beta \mathscr{E}_{loc}(\mathbf{d},H(\Phi)). \tag{10}$$

With g_k as a low-pass filter, the image difference between the filtered cartoon image and the original cartoon image will have its edges properly identified and the areas with slowly varying intensities disregarded. By incorporating the image difference, the model would take into account weak edges and illumination bias.

The effectiveness of incorporating (9) depends on the choice of the convolution kernel g_k. Since g_k needs to be a low-pass filter, one issue to be aware of is over-smoothing of the edges. For the purpose of our paper, we use a Gaussian filter, which weighs more pixels close to the center than pixels distant from it. This property effectively preserves the edges of the image as the image is being smoothed. Moreover, the choice of standard deviation for the Gaussian filter enables greater control in the amount of smoothing.

3.2 Ginzburg-Landau Approximation for the Local MCV Model

Computing the minimizer for the energy functional (10) can be computationally expensive. For example, see the work of Getreuer (2012). Instead, one could alternatively obtain an approximate solution by threshold dynamics, suggested by Esedoglu and Tsai (2006). To this end, we approximate the energy functional (10) by a sequence of energies

$$\mathscr{E}^{\varepsilon}_{CVloc}(\mathbf{c},\mathbf{d},\mathbf{u}) = \lambda \mathscr{E}_{fid}(\mathbf{c},\mathbf{u}) + \mu \mathscr{E}^{\varepsilon}_{GL}(\mathbf{u}) + \beta \mathscr{E}_{loc}(\mathbf{d},\mathbf{u}),$$

where the perimeter term $\mathscr{E}_{per}$ in the multiphase CV model (8) is replaced by the Ginzburg-Landau functional

$$\mathscr{E}^{\varepsilon}_{GL}(\mathbf{u}) = \varepsilon \int_\Omega (|\nabla u_1|^2 + |\nabla u_2|^2) + \frac{1}{\varepsilon}\int_\Omega (W(u_1) + W(u_2)),$$

where $W(u) = u^2(1-u)^2$. Applying the results of Modica and Mortola (1977); Modica (1987), it can be shown that $\mathscr{E}^{\varepsilon}_{GL}(\mathbf{u})$ Γ-converges to $\mathscr{E}_{per}(\mathbf{u})$ as $\varepsilon \to 0^+$. Note that everything in the new energy functional is now expressed in terms of the new functions u_i instead of $H(\phi_i)$ for $i = 1, 2$.

By calculus of variations, keeping $\mathbf{d}$ and $\mathbf{u}$ fixed and minimizing with respect to $\mathbf{c}$, we obtain

$$\begin{aligned} c_1 &= \mathscr{A}[u_0, u_1, u_2] \\ c_2 &= \mathscr{A}[u_0, u_1, 1-u_2] \\ c_3 &= \mathscr{A}[u_0, 1-u_1, u_2] \\ c_4 &= \mathscr{A}[u_0, 1-u_1, 1-u_2], \end{aligned} \tag{12}$$

where

$$\mathscr{A}[u_0,u_1,u_2] = \frac{\int_\Omega u_0 u_1 u_2}{\int_\Omega u_1 u_2}.$$

Similarly, keeping $\mathbf{c}$ and $\mathbf{u}$ fixed and minimizing with respect to $\mathbf{d}$, we obtain

$$\begin{aligned} d_1 &= \mathscr{A}[g_k * u_0 - u_0, u_1, u_2] \\ d_2 &= \mathscr{A}[g_k * u_0 - u_0, u_1, 1-u_2] \\ d_3 &= \mathscr{A}[g_k * u_0 - u_0, 1-u_1, u_2] \\ d_4 &= \mathscr{A}[g_k * u_0 - u_0, 1-u_1, 1-u_2]. \end{aligned} \tag{13}$$

Keeping now $\mathbf{c}$ and $\mathbf{d}$ fixed and minimizing $\mathscr{E}^{\varepsilon}_{CVloc}$ with respect to $\mathbf{u}$, we deduce the Euler-Lagrange equations for $\mathbf{u}$. It is a common technique to parametrize the descent direction by an artificial time variable $t \geq 0$ and initialize $\mathbf{u}(x,0)$. The equations for $\mathbf{u}(x,t) = (u_1(x,t), u_2(x,t))$ are

$$\begin{aligned} \frac{\partial u_1}{\partial t} = &-\lambda \mathscr{L}(u_0,\mathbf{c},u_2)[u_1] \\ &-\beta \mathscr{L}(g_k * u_0 - u_0, \mathbf{c}, u_2)[u_1] \\ &+\mu\left(2\varepsilon \Delta u_1 - \frac{1}{\varepsilon} W'(u_1)\right) \end{aligned} \tag{14}$$

$$\begin{aligned} \frac{\partial u_2}{\partial t} = &-\lambda \mathscr{L}(u_0,\bar{\mathbf{c}},u_1)[u_2] \\ &-\beta \mathscr{L}(g_k * u_0 - u_0, \bar{\mathbf{c}}, u_1)[u_2] \\ &+\mu\left(2\varepsilon \Delta u_2 - \frac{1}{\varepsilon} W'(u_2)\right), \end{aligned} \tag{15}$$

where $\bar{\mathbf{c}} = (c_1, c_3, c_2, c_4)$ and the operator $\mathscr{L}(u_0,\mathbf{c},u_2)[u_1]$ is constant with respect to u_1, is linear in u_2, and is given by

$$\begin{aligned} \mathscr{L}(u_0,\mathbf{c},u_2)[u_1] = &(c_1-u_0)^2 u_2 + (c_2-u_0)^2(1-u_2) \\ &-(c_3-u_0)^2 u_2 - (c_4-u_0)^2(1-u_2). \end{aligned}$$

At $t = 0$, we initialize u_1 and u_2 as the checkerboard functions

$$\begin{aligned} u_1(x,0) &= \mathbb{1}_{\{\sin\frac{\pi x_1}{3}\sin\frac{\pi x_2}{3}>0\}} \\ u_2(x,0) &= \mathbb{1}_{\{\sin\frac{\pi x_1}{10}\sin\frac{\pi x_2}{10}>0\}}, \end{aligned} \tag{16}$$

where $\mathbb{1}$ is the characteristic function, since initialization of the checkerboard function was observed to have faster convergence to the solution for the two-phase model (Getreuer 2012).

3.3 MBO scheme for Solving the Local MCV

In order to solve the above system of parabolic PDEs, we implement the MBO scheme (Merriman et al 1992, 1994). The underlying idea is to solve the PDE in three steps: solve the linear ODE

$$\frac{\partial w}{\partial t} = -\lambda \mathscr{L}(u_0, \mathbf{c}, u)[w] - \beta \mathscr{L}(g_k * u_0 - u_0, \mathbf{c}, u)[w], \tag{17}$$

solve the heat equation,

$$\frac{\partial v}{\partial t} = 2\mu\varepsilon\Delta v \tag{18}$$

and apply thresholding, which corresponds to solving the nonlinear ODE

$$\frac{\partial u}{\partial t} = -\frac{\mu}{\varepsilon}W'(u). \tag{19}$$

We use the fact that as $\varepsilon \to 0$, the solution is approaching one of the steady states ($u = 0$ and $u = 1$). Since we are dealing with a system of equations, we need to alternate the steps between the two functions u_1 and u_2.

Using the MBO scheme, we develop an iterative algorithm that computes a sequence of solutions $\{\mathbf{c}^n, \mathbf{d}^n, \mathbf{u}^n\}$. After initialization of $\mathbf{u}^0 = (u_1^0, u_2^0)$ as the checkerboard functions (16) and the computation of

$$\tilde{u}_0 = g_k * u_0 - u_0,$$

for each iteration $n \in \mathbb{N}$ we proceed as follows:

1. Compute the average intensities

$$\begin{aligned} c_1^n &= \mathscr{A}[u_0, u_1^n, u_2^n] \\ c_2^n &= \mathscr{A}[u_0, u_1^n, 1-u_2^n] \\ c_3^n &= \mathscr{A}[u_0, 1-u_1^n, u_2^n] \\ c_4^n &= \mathscr{A}[u_0, 1-u_1^n, 1-u_2^n]. \end{aligned} \tag{20}$$

and average differences of intensities

$$\begin{aligned} d_1^n &= \mathscr{A}[\tilde{u}_0, u_1^n, u_2^n] \\ d_2^n &= \mathscr{A}[\tilde{u}_0, u_1^n, 1-u_2^n] \\ d_3^n &= \mathscr{A}[\tilde{u}_0, 1-u_1^n, u_2^n] \\ d_4^n &= \mathscr{A}[\tilde{u}_0, 1-u_1^n, 1-u_2^n]. \end{aligned} \tag{21}$$

2. Let v_1^{n+1} be the solution of the ODE (17), with initial data u_1^n, computed at time dt for the operator $\mathscr{L}(\cdot, \mathbf{c}, u_2^n)$. This can be easily solved via a finite difference scheme of the form

$$\begin{aligned} \frac{v_1^{n+1} - u_1^n}{dt} = &-\lambda \mathscr{L}(u_0, \mathbf{c}, u_2^n)[u_1^n] \\ &-\beta \mathscr{L}(\tilde{u}_0, \mathbf{c}, u_2^n)[u_1^n]. \end{aligned} \tag{22}$$

3. Let w_1^{n+1} be the solution of the heat equation (18), with initial data v_1^{n+1}, computed at time dt. The equation can be solved in the Fourier domain as

$$\mathscr{F}_2(w_1^{n+1}) = \frac{1}{1 + 2\mu\, dt|\xi|^2}\mathscr{F}_2(v_1^{n+1}). \tag{23}$$

4. Threshold to approach steady-state solutions of equation (19) as follows:

$$u_1^{n+1} = \begin{cases} 0 & \text{if} \quad w_1^{n+1} \in (-\infty, \frac{1}{2}] \\ 1 & \text{if} \quad w_1^{n+1} \in (\frac{1}{2}, +\infty). \end{cases} \tag{24}$$

5. Set v_2^{n+1} to be the solution of the ODE (17), with initial data u_2^n, computed at time dt, for the operator $\mathscr{L}(\cdot, \bar{\mathbf{c}}, u_1^{n+1})$. This can be solved, as before, via a finite difference scheme of the form

$$\begin{aligned} \frac{v_2^{n+1} - u_2^n}{dt} = &-\lambda \mathscr{L}(u_0, \bar{\mathbf{c}}, u_1^{n+1})[u_2^n] \\ &-\beta \mathscr{L}(\tilde{u}_0, \bar{\mathbf{c}}, u_1^{n+1})[u_2^n]. \end{aligned} \tag{25}$$

6. Let w_2^{n+1} be the solution of the heat equation (18), with initial data v_2^{n+1}, computed at time dt. As before, we solve the equation in the Fourier domain as

$$\mathscr{F}_2(w_2^{n+1}) = \frac{1}{1 + 2\mu\, dt|\xi|^2}\mathscr{F}_2(v_2^{n+1}). \tag{26}$$

7. Threshold to approach steady-state solutions of equation (19) as follows:

$$u_2^{n+1} = \begin{cases} 0 & \text{if} \quad w_2^{n+1} \in (-\infty, \frac{1}{2}] \\ 1 & \text{if} \quad w_2^{n+1} \in (\frac{1}{2}, +\infty). \end{cases} \tag{27}$$

In order to obtain the segmented result $\tilde{u}$, we multiply u_2 by two and add it to u_1 to form at most four segmented regions. If we simply add u_1 and u_2 together, we would have at most three regions. The algorithm for minimizing (10) is summarized in Algorithm 2. Results are shown and discussed in Section 5.

4 Texture segmentation using Empirical Wavelet Transform

STM images may be comprised of various texture patterns, as shown in Figure 3. These patterns are not in general simple waves but rather a combination of several simple oscillations. The goal of this section is to partition the image textures into several components, each grouping together pixels belonging to a similar pattern. Texture segmentation is in general a difficult task because it needs to take into

Algorithm 2: MBO scheme for local MCV

Input: Image u_0, parameters λ, μ, β, dt

1: Compute
$$\tilde{u}_0 = g_k * u_0 - u_0$$
using the Gaussian filter, preferably imgaussfilt in MATLAB.
2: Initialize u_1 and u_2 as in (16).
3: **for** $i = 1$ to n **do**
4: Compute the average intensities $\mathbf{c}^i$ as in (20).
5: Compute the average intensities $\mathbf{d}^i$ as in (21).
6: Compute the solution $\mathbf{u}^i$ using equations (22)-(27).
7: **end for**
8: Combine u_1^n and u_2^n to obtain multiphase image $\tilde{u}^n$, i.e.
$$\tilde{u}^n = u_1^n + 2u_2^n$$

Output: Segmented Image $\tilde{u}^n$

account all types of variability within textures. Such difficulties can be leveraged in the case of particular textures, as for STM images, since the textures correspond to periodic patterns at different frequencies and with different orientations.

Directional image decomposition methods involve a decomposition of the Fourier spectrum into basis elements. These methods include Gabor filters (Jain and Farrokhnia 1991; Dunn and Higgins 1993; Dunn et al 1994; Dunn and Higgins 1995; Weldon et al 1996), wavelets (Strang 1993; Arivazhagan and Ganesan 2003; Unser 1995), curvelets (Arivazhagan et al 2006; Candes and Donoho 2005; Candes et al 2006; Shen and Yin 2009) and shearlets (Guo et al 2006). However, they are not adaptive and may result in incoherent partitions for STM-type images. Adaptive methods, on the other hand, provide better image decomposition since the basis elements are generated by the information contained in the image itself. Among these adaptive methods, the 2D variational mode decomposition (VMD) is a non-recursive, fully adaptive algorithm that sparsely decomposes signals/images into ensembles of constituent modes (Dragomiretskiy and Zosso 2014, 2015; Zosso et al 2017b). By minimizing an energy functional, this method simultaneously retrieves a given number of modes (texture patterns) together with their supports and the frequencies around which they are band-limited. To accommodate microscopy images, where texture patterns are combinations of simple modes, the energy functional is adapted to lattices by coupling several modes with a single support function. The solution to the energy functional is optimized using alternating direction method of multipliers (ADMM) (Gabay and Mercier 1976; Glowinski and Marroco 1975) and the MBO scheme, requiring several parameters to tune for the fidelity, penalty, and regularization terms involved as well as the convergence rate of ADMM (Boyd et al 2011). The VMD is an effective segmentation model, which has been demonstrated to work well on a broad variety of images. However, for our purpose, it

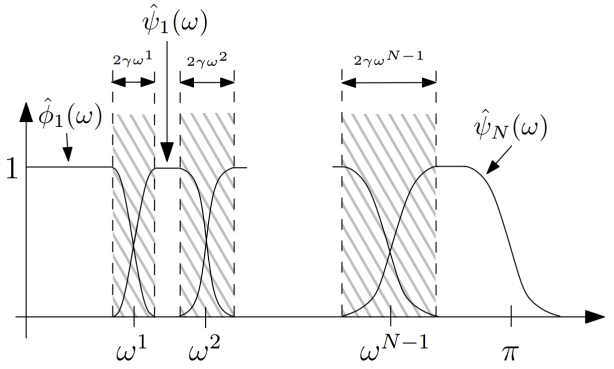


Fig. 5 Partitioning of the Fourier spectrum of a 1D signal

presents two major inconveniences: (1) the large number of parameters to be tuned, which can become time consuming when trying to achieve the best texture segmentation result, and (2) the explicit number of active modes required in the decomposition, which is restrictive.

An alternative method is the empirical wavelet transform (EWT) recently proposed by Gilles (2013). It is very well adapted to patterns specific to STM image textures, and it automatically finds the number of modes while requiring very few parameters to tune. Using EWT, we propose a texture segmentation algorithm adapted to microscopy image textures, which consists of three steps: (1) perform the EWT on the texture component, (2) construct a feature matrix based on the processed EWT coefficients, and (3) apply a clustering algorithm to the feature matrix in order to obtain the final segmentation. Detailed below, the main contributions in the EWT algorithm are the improvement of boundary detection and partitioning the Fourier spectrum and the selection of texture features based on their local energy.

4.1 The Empirical Wavelet Transform

The EWT was originally proposed as a signal decomposition method that detects and separates the signals' principal harmonic modes. The principal modes are modeled as amplitude modulated-frequency modulated (AM-FM) signals with compact support in the Fourier domain (Gilles 2013). The EWT consists of two steps: (1) it partitions the Fourier spectrum into N supports and it builds in the Fourier domain a filter bank, where each wavelet filter corresponds to a support, and (2) it filters the input signal with the obtained filter bank to produce the different components. The filter bank consists of N wavelet filters: one low-pass filter corresponding to the approximation component and $N-1$ bandpass filters corresponding to the details components.

The partitioning of the Fourier spectrum is as important as building the adaptive wavelets since it provides information about the principal harmonic components. Several approaches to perform the boundaries detection in the Fourier domain were investigated by Gilles (2013); Gilles and Heal (2014); Gilles et al (2014). In particular, Gilles and Heal

(2014) proposed a fully automatic algorithm based on a combination of a scale-space representation and Otsu's method. The advantage of this approach is in that it automatically finds the number N of expected modes as well as detects the boundaries of the Fourier supports. We used this approach in all experiments.

Assume that the Fourier spectrum is partitioned into N contiguous segments with boundaries $\{\omega_n\}_{n=0}^{N}$, where $\omega_0 = 0$ and $\omega_N = \pi$ (see Figure 5). Then, based on Meyer's wavelet formulation, we construct a filter bank of wavelets

$$\{\phi_1(x), \{\psi_n(x)\}_{n=1}^{N-1}\},$$

on the corresponding segments $[\omega_{n-1}, \omega_n]$. The Fourier transform (denoted as $\mathscr{F}_1$ in the 1D case) of the empirical scaling function is given by

$$\mathscr{F}_1(\phi_1)(\omega) = \begin{cases} 1 & \text{if } |\omega| \leq (1-\gamma)\omega_1 \\ \cos\left[\frac{\pi}{2}\mathscr{B}\left(\frac{1}{2\gamma\omega_1}(|\omega| - (1-\gamma)\omega_1)\right)\right] & \\ & \text{if } (1-\gamma)\omega_1 \leq |\omega| \leq (1+\gamma)\omega_1 \\ 0 & \text{if otherwise} \end{cases}, \tag{28}$$

whereas the Fourier transforms of the empirical wavelets are given by

$$\mathscr{F}_1(\psi_n)(\omega) = \begin{cases} 1 & \text{if } (1+\gamma)\omega_n \leq |\omega| \leq (1-\gamma)\omega_{n+1} \\ \cos\left[\frac{\pi}{2}\mathscr{B}\left(\frac{1}{2\gamma\omega_{n+1}}(|\omega| - (1-\gamma)(\omega_{n+1})\right)\right] & \\ & \text{if } (1-\gamma)\omega_{n+1} \leq |\omega| \leq (1+\gamma)\omega_{n+1} \\ \sin\left[\frac{\pi}{2}\mathscr{B}\left(\frac{1}{2\gamma\omega_n}(|\omega| - (1-\gamma)\omega_n)\right)\right] & \\ & \text{if } (1-\gamma)\omega_n \leq |\omega| \leq (1+\gamma)\omega_n \\ 0 & \text{if otherwise} \end{cases}, \tag{29}$$

for $n = 1, \ldots, N$. The function $\mathscr{B}$ is an arbitrary $\mathscr{C}^k([0,1])$ function satisfying the properties that $\mathscr{B}(t) = 0$ if $t \leq 0$, $\mathscr{B}(t) = 1$ if $t \geq 1$, and $\mathscr{B}(t) + \mathscr{B}(1-t) = 1$ and $\mathscr{B}(t) \in (0,1)$ for all $t \in [0,1]$. The parameter γ is chosen to ensure that two consecutive transition areas (shown as dashed areas in Figure 5) do not overlap. As shown by Gilles (2013), a proper selection of γ guarantees that the filter bank $\{\phi_1, \{\psi_n\}_{n=1}^{N-1}\}$ is a tight frame in $L^2(\mathbb{R})$. Then, the EWT is defined in the same way as the classical Wavelet Transform. For the signal function f, the details coefficients are given as

$$\mathscr{W}_f^{\mathscr{E}}(n,x) = \mathscr{F}_1^*\left(\mathscr{F}_1(f)(\omega)\overline{\mathscr{F}_1(\psi_n)(\omega)}\right)(x), \tag{30}$$

and the approximation coefficient as

$$\mathscr{W}_f^{\mathscr{E}}(0,x) = \mathscr{F}_1^*\left(\mathscr{F}_1(f)(\omega)\overline{\mathscr{F}_1(\phi_1)(\omega)}\right)(x), \tag{31}$$

where $\mathscr{F}_1^*$ stands for the inverse 1D Fourier Transform.

The EWT was later generalized to 2D images for various kinds of wavelet transform, specifically tensor wavelets, Littlewood-Paley wavelet transform, the ridgelet transform, and the curvelet transform (Gilles et al 2014).

4.2 The Empirical Curvelet Transform

Textures in STM images can be seen as oscillatory patterns with multiple orientations. Among all of the above mentioned variants, curvelets are wavelets that take into account various orientations (Candes and Donoho 2005). Therefore, its empirical counterpart, the Empirical Curvelet Transform (ECT), is the most appropriate adaptation of EWT to partition texture images. Similar to the EWT, the ECT builds a filter bank in the Fourier domain where each filter has its support on a polar wedge.

As shown in Figure 6b, the Fourier domain is partitioned in this case into a disk centered at the origin, which contains the low frequencies, and concentric annuli and angular sectors (polar wedges), which contain high frequencies. In order to build the filter bank adaptively, the ECT needs to detect the boundaries of the polar wedges empirically, which correspond to finding the scales for the angle and the radii. Following Gilles et al (2014), this step can be achieved by considering the pseudo-polar Fourier transform (Averbuch et al 2006, 2008) and performing the previously described 1D detection to 1D spectra corresponding to averaging with respect to the frequency magnitude and orientation, respectively. Gilles et al (2014) proposed three different cases: (1) scales and angles are detected independently, (2) scales are detected first and angles are detected per each scale, and (3) angles are detected first and scales are detected per each angular sector. In this work, we will consider the second option because STM images contain texture patterns with different main orientations having varying frequency magnitudes.

In the following, we will denote $\omega = (\omega_{x_1}, \omega_{x_2})$ the frequency coordinates in the Fourier plane, $|\omega|$ its magnitude and θ its angle. Otsu's boundary detection method will then provide N_θ number of angles and N_s^m number of scales per each angular sector $m = 1, \ldots, N_\theta$. This is equivalent to obtaining the set of angular boundaries $\{\theta_m\}_{m=1}^{N_\theta}$ and the set of scale boundaries $\{\omega_n^m\}_{n=1}^{N_s^m}$ per angular sector for $m = 1, \ldots, N_\theta$. Note that $\omega_1^1 = \ldots = \omega_1^{N_\theta}$ because all together they form the disk centered at the origin of the Fourier domain. The corresponding curvelet filters are then defined in the Fourier domain (we denote $\mathscr{F}_2$ the 2D Fourier transform) in the following way. The purely radial lowpass filter ϕ_1 is given by

$$\mathscr{F}_2(\phi_1)(\omega,\theta) = \begin{cases} 1 & \text{if } \quad |\omega| \leq (1-\gamma)\omega_1, \\ \cos\left[\frac{\pi}{2}\mathscr{B}\left(\frac{1}{2\gamma\omega_1}(|\omega| - (1-\gamma)\omega_1)\right)\right] & \\ & \text{if } \quad (1-\gamma)\omega_1 \leq |\omega| \leq (1+\gamma)\omega_1, \\ 0 & \text{if otherwise,} \end{cases} \tag{32}$$

where $\omega_1 = \omega_1^1 = \ldots = \omega_1^{N_\theta}$. The polar curvelet associated to the polar wedge $I_n^m := [\theta_m, \theta_{m+1}] \times [\omega_n^m, \omega_{n+1}^m]$ can be written as

$$\mathscr{F}_2(\psi_{m,n})(\omega,\theta) = W_n^m(|\omega|)V_m(\theta), \tag{33}$$

where the radial window W_n^m is

$$W_n^m(|\omega|) = \begin{cases} 1 & \text{if} \quad (1+\gamma)\omega_n^m \leq |\omega| \leq (1-\gamma)\omega_{n+1}^m, \\ \cos\left[\frac{\pi}{2}\mathscr{B}\left(\frac{1}{2\gamma\omega_{n+1}^m}\left(|\omega| - (1-\gamma)\omega_{n+1}^m\right)\right)\right] & \\ & \text{if} \quad (1-\gamma)\omega_{n+1}^m \leq |\omega| \leq (1+\gamma)\omega_{n+1}^m, \\ \sin\left[\frac{\pi}{2}\mathscr{B}\left(\frac{1}{2\gamma\omega_n^m}\left(|\omega| - (1-\gamma)\omega_n^m\right)\right)\right] & \\ & \text{if} \quad (1-\gamma)\omega_n^m \leq |\omega| \leq (1+\gamma)\omega_n^m, \\ 0 & \text{if otherwise,} \end{cases} \tag{34}$$

for $n \neq N_s^m - 1$ and

$$W_{N_s^m-1}^m(|\omega|) = \begin{cases} 1 & \text{if} \quad (1+\gamma)\omega_{N_s^m-1}^m \leq |\omega|, \\ \sin\left[\frac{\pi}{2}\mathscr{B}\left(\frac{1}{2\gamma\omega_{N_s^m-1}^m}\left(|\omega| - (1-\gamma)\omega_{N_s^m-1}^m\right)\right)\right] & \\ & \text{if} \quad (1-\gamma)\omega_{N_s^m-1}^m \leq |\omega| \leq (1+\gamma)\omega_{N_s^m-1}^m, \\ 0 & \text{if otherwise,} \end{cases} \tag{35}$$

for $n = N_s^m - 1$, while the angular window V_m is

$$V_m(\theta) = \begin{cases} 1 & \text{if} \quad \theta_m + \Delta\theta \leq \theta \leq \theta_{m+1} - \Delta\theta, \\ \cos\left[\frac{\pi}{2}\mathscr{B}\left(\frac{1}{2\Delta\theta}\left(\theta - \theta_{m+1} + \Delta\theta\right)\right)\right] & \\ & \text{if} \quad \theta_{m+1} - \Delta\theta \leq \theta \leq \theta_{m+1} + \Delta\theta, \\ \sin\left[\frac{\pi}{2}\mathscr{B}\left(\frac{1}{2\Delta\theta}\left(\theta - \theta_m + \Delta\theta\right)\right)\right] & \\ & \text{if} \quad \theta_m - \Delta\theta \leq \theta \leq \theta_m + \Delta\theta, \\ 0 & \text{if otherwise.} \end{cases} \tag{36}$$

The parameters γ and $\Delta\theta$ are chosen in order to guarantee the tight frame property (see Gilles et al (2014) for details). This leads to the construction of the filter bank of empirical curvelets

$$\left\{\phi_1(x), \{\psi_{m,n}(x)\}_{\substack{m=1,\ldots,N_\theta \\ n=1,\ldots,N_s^m-1}}\right\}. \tag{37}$$

From (37), the empirical curvelet transform of the texture component v is given by the detail coefficients as

$$\mathscr{W}_v^{\mathscr{E}\mathscr{C}}(m,n,x) = \mathscr{F}_2^*\left(\mathscr{F}_2(v)\overline{\mathscr{F}_2(\psi_{m,n})}\right)(x), \tag{38}$$

and the approximation coefficients as

$$\mathscr{W}_v^{\mathscr{E}\mathscr{C}}(0,0,x) = \mathscr{F}_2^*\left(\mathscr{F}_2(v)\overline{\mathscr{F}_2(\phi_1)}\right)(x), \tag{39}$$

where $\mathscr{F}_2^*$ stands for the inverse 2D Fourier Transform. We can reconstruct the image $v(x)$ by the inverse transform:

$$\begin{aligned} v(x) = \mathscr{F}_2^*\Bigg(&\mathscr{F}_2\left(\mathscr{W}_v^{\mathscr{E}\mathscr{C}}(0,0,\cdot)\right)\mathscr{F}_2(\phi_1) \\ &+ \sum_{m=1}^{N_\theta}\sum_{n=1}^{N_s^m-1}\mathscr{F}_2\left(\mathscr{W}_v^{\mathscr{E}\mathscr{C}}(m,n,\cdot)\right)\mathscr{F}_2(\psi_{m,n})\Bigg)(x). \end{aligned} \tag{40}$$

Each empirical curvelet subband $\mathscr{W}_v^{\mathscr{E}\mathscr{C}}(m,n,\cdot)$ contains either some textural pattern or noise. In many cases, some of these curvelets may have extremely weak oscillatory patterns or pure noise, deeming them useless in our texture analysis. Therefore, we propose a pre-processing step and a post-processing step in order to obtain only empirical curvelets with meaningful information.

4.3 Improved Boundaries Detection

In this section, we propose modifications in the boundary detection algorithm in order to reduce the number of irrelevant polar wedges. Since the detection is performed in the pseudo-polar Fourier domain, the following processing will also be performed in the pseudo-polar domain. Hereafter, we will denote $\mathscr{F}_P(v)$ as the pseudo-polar Fourier transform of the input texture image v.

Our first improvement is a preprocessing step before the actual detection. Since the goal is to isolate clusters of high magnitude frequencies in $\mathscr{F}_P(v)$ we suggest thresholding the Fourier coefficients in order to remove all frequencies that are not relevant. Let T be the hard-thresholding operator defined as

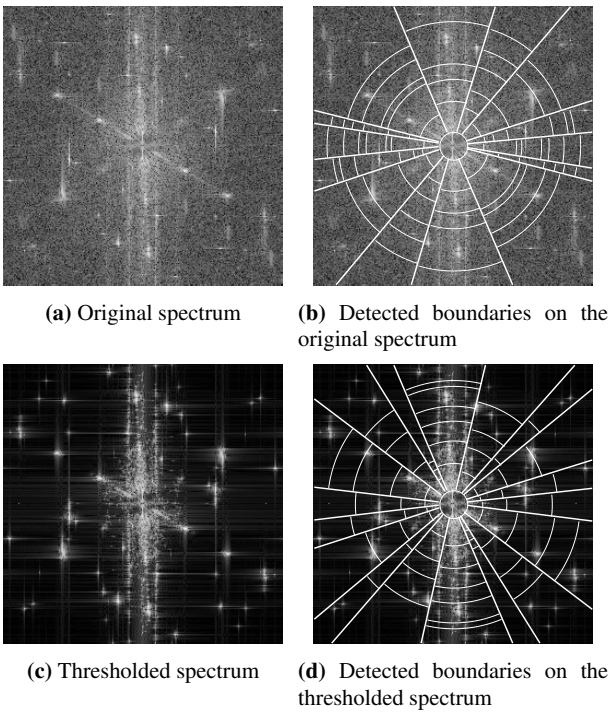
$$T(a,\tau) = \begin{cases} 0 & \text{if} \quad |a| \leq \tau \\ a & \text{if} \quad |a| > \tau \end{cases}. \tag{41}$$

The preprocessing step consists of performing the detection on $T(\mathscr{F}_P(v),\tau)$ instead of $\mathscr{F}_P(v)$. The threshold τ is chosen as a certain percentile of the magnitude of the Fourier coefficients: we create a vector whose entries are the magnitudes of the Fourier coefficients listed in increasing order and we set τ to be the value of the $n \times p - th$ entry of the vector, where n is the length of the vector and p the specified percentile (i.e. $p \in (0,1)$). Figure 6 illustrates the effects of performing such thresholding. Figures 6a and 6c show the original spectrum as well as its thresholded version (using a 92 percentile). Figures 6b and 6d provide the corresponding detected partitions. We note that the partition obtained from the thresholded spectrum provides a separate angular sector for the high magnitude clusters near the top right corner while these clusters are associated with another angular sector in the partition obtained from the original spectrum.

Unfortunately, sometimes some meaningless polar wedges are detected in each angular sector. Our second improvement aims at removing these useless polar wedges by merging them with their neighbors. Suppose that $\{\theta_m\}_{m=1}^{N_\theta}$ and $\left\{\{\omega_n^m\}_{n=1}^{N_s^m}\right\}_{m=1}^{N_\theta}$ are the set of angles and scales detected on $T(\mathscr{F}_P(v),\tau)$. In the pseudo-polar domain, a polar wedge corresponds to the rectangle I_n^m, whose area we denote by $A_n^m = area(I_n^m) = (\omega_{n+1}^m - \omega_n^m)(\theta_{m+1} - \theta_m)$. We define the information density per polar wedge by

$$M_n^m = \frac{\|T(\mathscr{F}_P(v),\tau)\mathbb{1}_{I_n^m}\|_1}{A_n^m}, \tag{42}$$

where $\mathbb{1}_{I_n^m}$ is the characteristic function over the domain I_n^m. As a reference, we use the largest density $\tilde{M} = \max_{m,n} M_n^m$ and we threshold at a certain fraction $\eta = 0.1$ as follows.



(a) Original spectrum

(b) Detected boundaries on the original spectrum

(c) Thresholded spectrum

(d) Detected boundaries on the thresholded spectrum

Fig. 6 Comparison of partitions obtained from the original spectrum and its thresholded version. The spectrum shown is reproduced and modified from Guttentag et al (2016a) with permission. STM image spectrum is copyright from American Chemical Society

For a given angular sector, we start from the polar wedge that is the farthest from the origin until we reach the one closer to the origin. If $M_n^m \leq \eta \tilde{M}$, then the I_n^m polar wedge is irrelevant and we merge it with I_{n-1}^m, i.e we remove ω_{n-1}^m from the list $\left\{\{\omega_n^m\}_{n=1}^{N_s^m}\right\}_{m=1}^{N_\theta}$ and update $N_s^m := N_s^m - 1$. If $M_n^m > \eta \tilde{M}$, we move to the next polar wedge I_{n-1}^m and repeat the procedure. The corresponding merging algorithm is summarized in Algorithm 3.

Partitions obtained before and after merging are illustrated in Figure 7. It is easy to see that some polar wedges having less information were merged to form a new set of polar wedges having a minimum amount of useful information (see for instance the most vertical angular sector, the thin outer wedge no longer exist in the updated partition).

Based on the updated partition, the empirical curvelet filter bank is then constructed accordingly to (32) and (33). The full modified empirical curvelets transform is summarized in Algorithm 4.

As shown in Gilles et al (2014), the tight frame property depends only on the construction process of the curvelet filters themselves and does not depend on how the supports detection is done. Therefore, the original proof remains valid even within the framework of our modified support detection algorithm, implying that the tight frame property is preserved.

Algorithm 3: Merging Curvelet partition

Input: Thresholded Fourier domain $T(\mathscr{F}_P(v), \tau)$, original boundaries $\{\theta_m\}_{m=1}^{N_\theta}$ and $\left\{\{\omega_n^m\}_{n=1}^{N_s^m}\right\}_{m=1}^{N_\theta}$

Compute M_n^m for $m = 1 \ldots N_\theta$ and $n = 1 \ldots N_s^m$ according to (42).
Compute $\tilde{M} = \max_{m,n} M_n^m$.
for $i = 1$ to N_θ **do**
  Set $j := N_s^i$.
  while $j \geq 3$ **do**
    if $M_{j-1}^i < 0.10\tilde{M}$ **then**
      Remove ω_{j-1}^i from $\left\{\{\omega_n^m\}_{n=1}^{N_s^m}\right\}_{m=1}^{N_\theta}$.
      Set $N_s^i := N_s^i - 1$.
    end if
    $j := j - 1$.
  end while
end for

Output: Updated boundaries $\left\{\{\theta_m\}_{m=1}^{N_\theta}, \left\{\{\omega_n^m\}_{n=1}^{N_s^m}\right\}_{m=1}^{N_\theta}\right\}$.

4.4 Texture Features

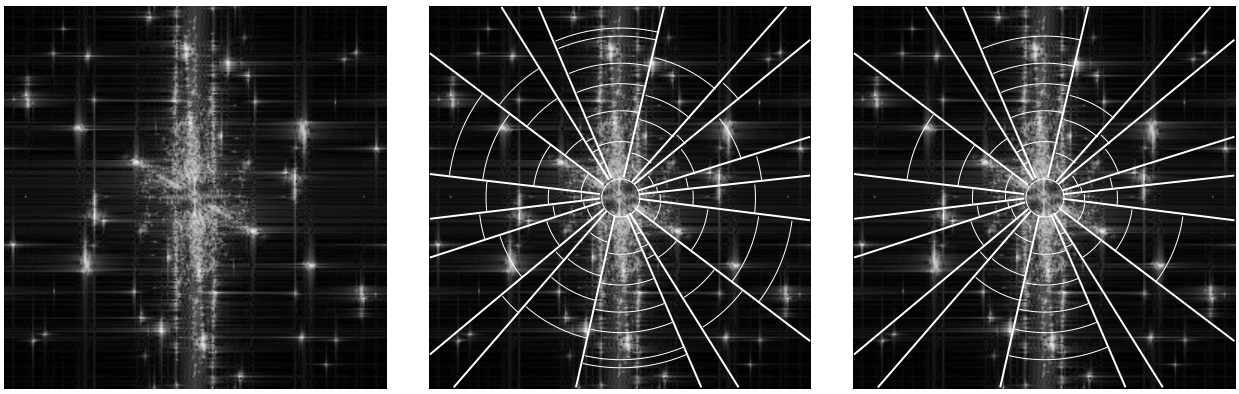
After applying our modified empirical curvelet transform to the texture component of the image, we finally construct the relevant information to characterize different textures. This information could be directly given by the empirical curvelets coefficients by reshaping each one into a vector corresponding to each pixel. However, this kind of feature vector does not have any inherent spatial information of the local neighborhood of its corresponding pixel. Instead, we consider computing the local "energy" of the curvelet coefficients. Then, we build a set of feature vectors associated to each pixel of the texture component. This set will form together a feature matrix whereby a clustering algorithm can be applied to it.

Define the local energy at (m,n,x) as

$$E_{m,n}^{\mathscr{MEC}}(x) = \frac{\|\mathbb{1}_{B(x,r_{m,n})} \mathscr{W}_v^{\mathscr{MEC}}(m,n,\cdot)\|_2}{|B(x,r_{m,n})|} \tag{43}$$

where $B(x,r_{m,n})$ is a $(2r_{m,n}+1)\times(2r_{m,n}+1)$ neighborhood window centered at x. The radius $r_{m,n}$ is determined by the frequency of the texture pattern so that $E_{m,n}^{\mathscr{MEC}}(x)$ captures enough information around x. Hence, with ω_n^m related to frequency, we choose $r_{m,n}$ proportional to $\lceil \frac{1}{\omega_n^m} \rceil$ so that the radius relates to the period of the texture pattern. In practice, we set $r_{m,n} = \lceil \frac{\pi}{\omega_n^m} \rceil$. Note that for pixels near or at the border of the image, we apply symmetric padding to align with our assumption that texture patterns are periodic.

After calculating (43) for each pixel for each empirical curvelet subband, we construct the feature matrix $\mathscr{D}$ by casting all energy matrices $E_{m,n}^{\mathscr{MEC}}$ as vectors so that they



(a) Thresholded spectrum **(b)** Partition before merging **(c)** Partition after merging

Fig. 7 Influence of the merging algorithm on the obtained partitions. The spectrum shown is reproduced and modified from Guttentag et al (2016a) with permission. STM image spectrum is copyright American Chemical Society

Algorithm 4: Modified Empirical Curvelet Transform

Input: Image $v(x)$, Threshold Value τ

1: Compute the Pseudo-Polar FFT $\mathscr{F}_P(v)$.
2: Threshold the Fourier coefficients to obtain $T(\mathscr{F}_P(v),\tau)$.
3: Detect the original partition $\left\{\{\theta_m\}_{m=1}^{N_\theta}, \left\{\{\omega_n^m\}_{n=1}^{N_s^m}\right\}_{m=1}^{N_\theta}\right\}$ using Otsu's Method as described in Gilles et al (2014); Gilles and Heal (2014).
4: Compute the updated set $\left\{\{\theta_m\}_{m=1}^{N_\theta}, \left\{\{\omega_n^m\}_{n=1}^{N_s^m}\right\}_{m=1}^{N_\theta}\right\}$ by applying the merging algorithm Algorithm 3.
5: Construct the corresponding curvelet filter bank $\mathscr{B}^{\mathscr{MECC}} = \left\{\phi_1(x), \{\psi_{m,n}(x)\}_{\substack{m=1,\dots,N_\theta \\ n=1,\dots,N_s^m-1}}\right\}$ accordingly to (32)-(33).
6: Filter $v(x)$ using (38)-(39) to obtain $\mathscr{W}_v^{\mathscr{MECC}} = \left\{\mathscr{W}_v^{\mathscr{MECC}}(0,0,x), \{\mathscr{W}_v^{\mathscr{MECC}}(m,n,x)\}_{\substack{m=1,\dots,N_\theta \\ n=1,\dots,N_s^m-1}}\right\}$

Output: Spectrum boundaries $\left\{\{\theta_m\}_{m=1}^{N_\theta}, \left\{\{\omega_n^m\}_{n=1}^{N_s^m}\right\}_{m=1}^{N_\theta}\right\}$, empirical curvelet filter bank $\mathscr{B}^{\mathscr{MECC}}$, empirical curvelets coefficients $\mathscr{W}_v^{\mathscr{MECC}}$.

form the columns of $\mathscr{D}$. Thus, $\mathscr{D}$ has the form

$$\mathscr{D} = \begin{pmatrix} | & \dots & | & | & \dots & | \\ E_{1,1}^{\mathscr{MECC}} & \dots & E_{1,N_s^1-1}^{\mathscr{MECC}} & E_{2,1}^{\mathscr{MECC}} & \dots & E_{N_\theta,N_s^{N_\theta}-1}^{\mathscr{MECC}} \\ | & \dots & | & | & \dots & | \end{pmatrix}. \quad (44)$$

Because texture patterns are locally periodic in an image, pixels belonging to the same texture pattern should both have similar energies as defined by (43). Hence, we could group the column vectors as belonging to the same class. We can then apply any clustering algorithm, such as k-means or multiclass MBO clustering (Garcia-Cardona et al 2014; Merkurjev et al 2014), on $\mathscr{D}$ to identify the texture patterns of the image.

5 Experimental Results

In this section, we present results of our framework applied to the images in Figure 3. All algorithms in this paper were implemented in MATLAB R2016a. The codes and the results are available at `https://github.com/kbui1993/Microscopy-Codes`.

In order to segment the images according to intensities or texture patterns, we first perform the cartoon+texture decomposition on each of them to obtain their cartoon and texture components. In our experiments, we select $\sigma = 3$ in Algorithm 1. This parameter leads to more appealing segmentation results compared to other values of σ. Moreover, $\sigma = 3$ is the minimum value for which humans could perceive region as textures (Buades et al 2011).

After obtaining the cartoon and texture components for each image, we apply our proposed methods to each component. Our results are divided into two subsections. The cartoon segmentation results are given in Section 5.1 while the texture segmentation results are given in Section 5.2.

5.1 Cartoon Segmentation Results

For comparison, we apply the multiphase algorithm based on the MBO scheme (Esedoglu and Tsai 2006) and the proposed local multiphase segmentation algorithm (Algorithm 2) to the cartoon component of each image in Figure 3. Convergence for both methods is reached when both phase fields u_1 and u_2 do not change within the same iteration. Otherwise, the maximum number of iterations is set at 200. For each image, the values for λ and μ are set the same between both methods, but the time step dt might be different because both methods are minimizing different energy functionals. Finally, the standard deviation for the Gaussian filter in Algorithm 2 is set to 10.

Here, we discuss the parameter selection for both the multiphase and local multiphase methods. In our experi-

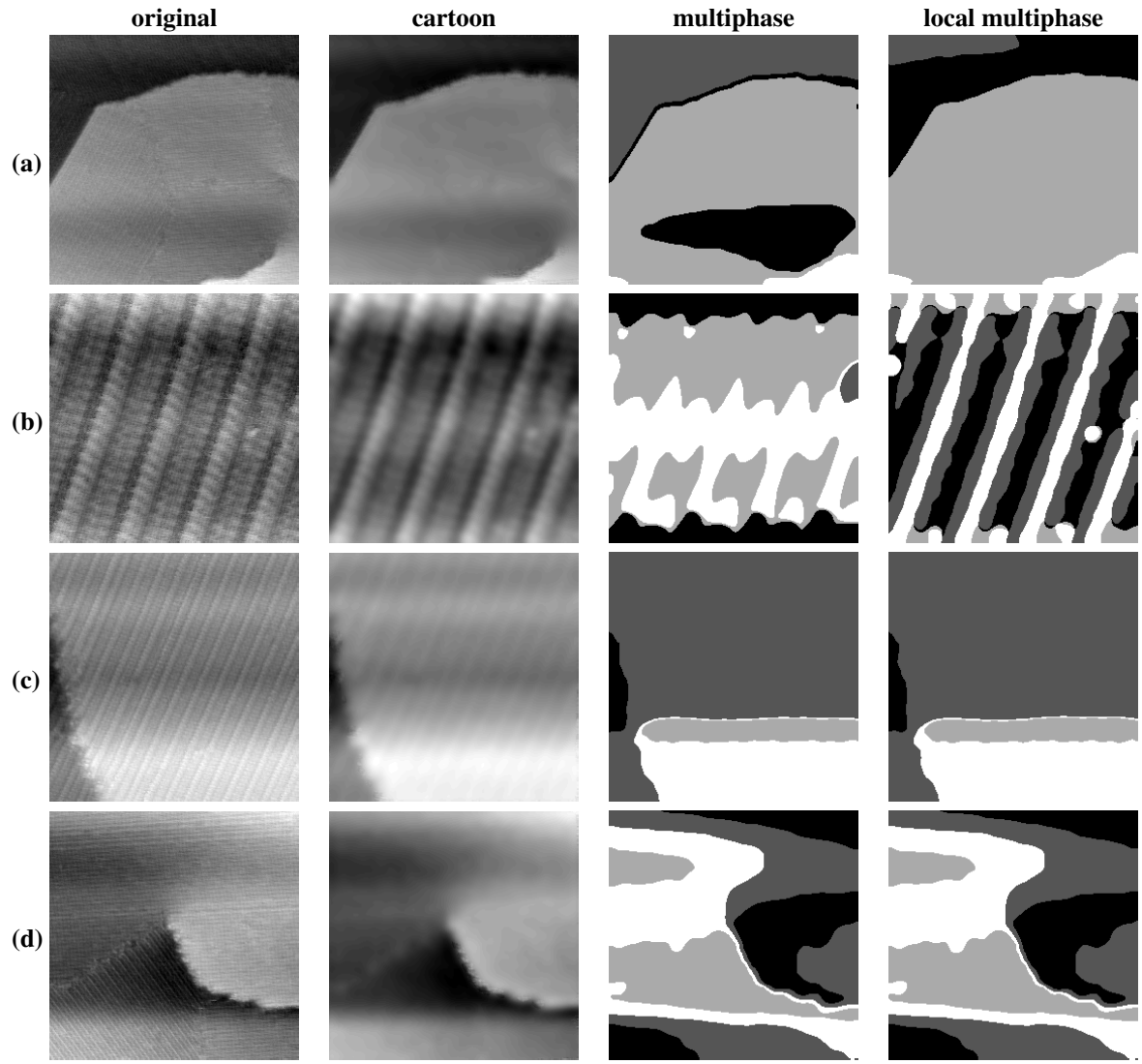


Fig. 8 Comparison between the multiphase CV and local MCV algorithms. The parameters are **(a)** $\lambda = 10$, $\mu = 10^{-3} \times 255^2$, $\beta = 10$, $dt_{CV} = 0.75$, $dt_{CVloc} = 3.2$ **(b)** $\lambda = 10$, $\mu = 10^{-3} \times 255^2$, $\beta = 300$, $dt_{CV} = 4$, $dt_{CVloc} = 4$ **(c)** $\lambda = 10$, $\mu = 10^{-3} \times 255^2$, $\beta = 60$, $dt_{CV} = 2$, $dt_{CVloc} = 2$ **(d)** $\lambda = 10$, $\mu = 10^{-3} \times 255^2$, $\beta = 10$, $dt_{CV} = 2.5$, $dt_{CVloc} = 2$. Raw scanning tunneling microscope images of cyanide on Au$\{111\}$, reproduced from Guttentag et al (2016a) with permission. Images copyright American Chemical Society

ments, the values of λ in (8) are set to 5, 7, or 10. If there are multiple regions or regions of varying sizes in the image, the value μ should be small. Otherwise, it should be large. For our images of size 255×255, we adopt four values for μ: $10^{-4} \times 255^2$, $10^{-3} \times 255^2$, $10^{-2} \times 255^2$, and $10^{-1} \times 255^2$. The parameter β from (10) is set as the same value as λ if there is intensity homogeneity. Otherwise, if there is intensity inhomogeneity, β is set to be larger than λ to ensure that potential regions with weak edges are detected. To ensure Γ-convergence of (11) when running Algorithm 2, the time step dt needs to be small (we set here $dt < 20$), and the number of iterations needs to be large, which is why the maximum is set to 200.

The results of each method are shown along with the original and the cartoon images in Figures 8 and 9. We observe that the results differ between the two methods. In general, local multiphase is robust against illumination bias and intensity inhomogeneity, thus leading to segmentation results that look similar to the cartoon images. On the other hand, multiphase produces results with artificial regions, *e.g.*, Figure 8a-c, or results that ignore weak-edged regions, *e.g.*, Figure 8d and Figure 9a-d.

In Figure 8a, the multiphase result detects the streak of shadow in the middle of the image along with an edge as a region. However, these should not appear as distinct regions at all. On the other hand, the segmentation result of the local multiphase result works properly. Moreover, it detects smaller and more subtle regions of the cartoon image, which are located in the top left and bottom right.

Figure 8b is heavily affected by illumination bias. Hence, we have a poor segmentation result from the multiphase algorithm. On the other hand, the result from local multiphase is able to identify the natural regions of its corre-

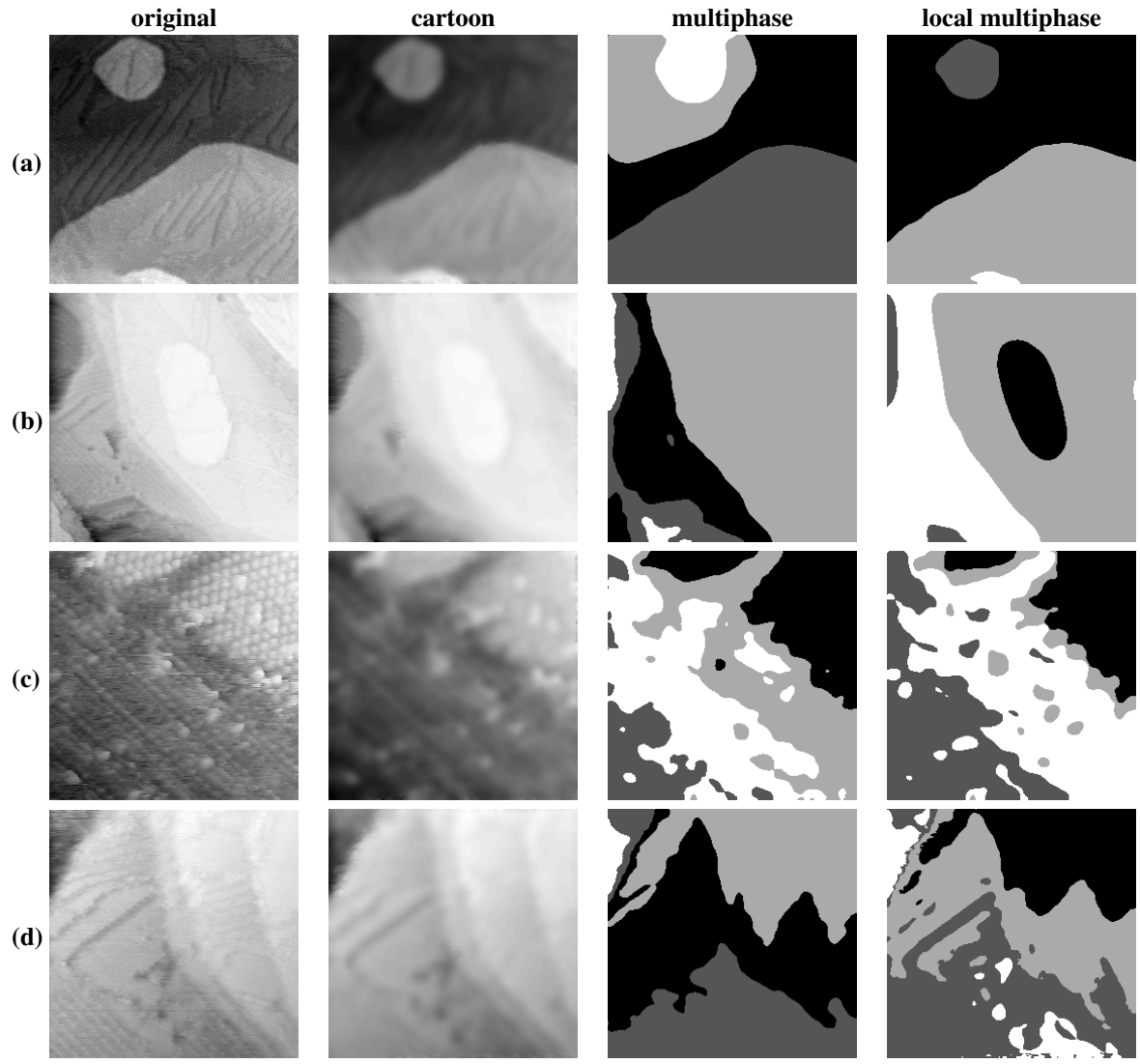


Fig. 9 Comparison between the multiphase CV and local MCV algorithms. The parameters are **(a)** $\lambda = 5$, $\mu = 10^{-1} \times 255^2$, $\beta = 70$, $dt_{CV} = 0.35$, $dt_{CVloc} = 0.10$ **(b)** $\lambda = 10$, $\mu = 10^{-2} \times 255^2$, $\beta = 30$, $dt_{CV} = 0.1$, $dt_{CVloc} = 6.5$ **(c)** $\lambda = 7$, $\mu = 10^{-4} \times 255^2$, $\beta = 65$, $dt_{CV} = 5$, $dt_{CVloc} = 18$ **(d)** $\lambda = 5$, $\mu = 10^{-4} \times 255^2$, $\beta = 50$, $dt_{CV} = 12$, $dt_{CVloc} = 0.6$. Raw scanning tunneling microscope images of cyanide on Au$\{111\}$, reproduced from Guttentag et al (2016a) with permission. Images copyright American Chemical Society

sponding cartoon image, thus appearing extremely similar to it.

In Figure 8c-d, both multiphase and local multiphase produce highly similar results. The results capture regions corresponding to the different light intensities in their corresponding cartoon images. In Figure 8c, the results for both multiphase and local multiphase look similar to their cartoon image. In Figure 8d, the segmentation results by multiphase and local multiphase appear to be dissimilar to the cartoon image, but upon closer inspection, the results detect regions of similar intensities that might be difficult to discern from the cartoon image.

In Figure 9a, the local multiphase result resembles the cartoon image more than does the multiphase result. The local multiphase result is able to capture the "island" region on the top right and the small streak at the very bottom of the image. The multiphase result is unable to capture the small streak and it attempts to segment the "island" region.

In Figure 9b, unlike multiphase result, the local multiphase result is able to identify the oval region at the right of the image. The oval region has weak edges because of the apparent intensity inhomogeneity, which the multiphase algorithm is unable to detect it.

As for Figure 9c, both multiphase and local multiphase results appear similar to the cartoon image, but local multiphase is able to identify the small "islands" in the middle of the image. The multiphase result tends to identify larger regions. but for one or two regions of the result, some parts of one region do not have similar gray-level intensities according to the cartoon image.

In Figure 9d, both results provide a segmentation based on the gray-level intensity. Local multiphase is able to detect shadows, such as those in the top left corner and at the

bottom of the image, with better precision. The multiphase segmentation captures wider transition regions. However, the slow varying vertical edges are not localized by neither of the two methods, in which case, most likely, a nonlocal version of the segmentation algorithms above would help improve the results.

In summary, because of the local term, the local multiphase algorithm provides better segmentation results than does the multiphase algorithm, The local term enables the local multiphase algorithm to capture with better precision regions with shadows and intensity inhomogeneities. The multiphase algorithm, on the other hand, fails to detect such regions, and its results are significantly worse under the presence of illumination bias and intensity inhomogeneity. Therefore, the local multiphase segmentation is preferred to the multiphase segmentation in the case of STM images.

5.2 Texture Segmentation Results

In order to segment the texture component of the image, we apply Algorithm 4 to obtain its ECT coefficients. Next we build the texture feature matrix accordingly to (44). Finally we apply a clustering algorithm to the energy matrix. The number of clusters is determined by the user. In our experiments, the clustering algorithms we use are k-means, already implemented in MATLAB, and multiclass MBO clustering (Garcia-Cardona et al 2014; Merkurjev et al 2014). Both methods use the cityblock metric to measure the similarity between data points of the energy matrix in order to determine the clusters.

As an unsupervised method, k-means utilizes random initialization to determine the initial centroids. In order to obtain high-quality clusters, Arthur and Vassilvitskii (2007) devised a heuristic to ensure that every two initial centroids are dissimilar to each other. As a semi-supervised method, the multiclass MBO clustering randomly selects 25% of the labels determined by k-means as its initialization. Note that because of random initialization of the centroids, results may differ for every run. Hence, for each image, we run 10 replications of k-means and select the best result based on minimum within-cluster sums of point-to-centroid distances. The best result is only shown and is used as an initialization to multiclass MBO clustering.

The multiclass MBO is a graph-based method, so it requires constructing the graph Laplacian matrix which is expensive both in computation and in memory. Instead of computing the matrix exactly, we use Nyström extension to approximate its eigen-decomposition (Fowlkes et al 2004). The method requires from the user the number of data points to sample and the number of eigenvectors to compute. In our experiments, we sample 300 data points and compute 30 eigenvectors.

For all images, the multiclass MBO clustering is set with the following parameters: $\mu = 30$ (fidelity parameter) and $\eta = 10^{-7}$ (tolerance), following the notations in the work of Garcia-Cardona et al (2014). The thresholding step is performed after every three iterations of the diffusion step. The time step dt is the only parameter that differs between the images.

Texture segmentation results obtained from k-means and multiclass MBO clustering are shown in Figures 10 and 11. We observe that both clustering algorithms are able to identify most of the recognizable texture patterns of the images. Although both clustering results are similar, the results obtained from multiclass MBO clustering tend to have smoother clusters and appear less noisy than the results obtained from k means. In other words, the multiclass MBO results are cleaner than the k-means results.

In Figure 10a, we select $k = 5$. Two of the clusters correspond to the material interfaces: one interface is a deep indentation and the other appears perforated. The three clusters correspond to the most periodic texture patterns: one that appears to have horizontal direction (the middle cluster), one that is slanted upward (the left cluster), and one that is finer and is slanted downward (top cluster).

In Figure 10b, we select $k = 2$. One cluster corresponds to the perforated lines while the other corresponds to the regions between the lines. However, some of the pixels appear to be misclassified. For example, pixels in the region between the perforated lines are classified as the same cluster. This result may be attributed to the noise or unexpected anomalies within the texture patterns. Since the clustering result from multiclass MBO appears to have less misclassified pixels than does k-means, it affirms that multiclass MBO is more resistant to noise, even with 25% of the labels provided by k-means.

In Figure 10c, we select $k = 2$ again. One cluster corresponds to the uniform texture pattern that has an upward, slanted direction while the other corresponds to the break dividing the uniform texture pattern. We observe that pixels near the break tend to be misclassified because of the break is relatively small compared to the uniform texture pattern.

In Figure 10d, we select $k = 3$: one corresponding to the perforated or indented edge, the other one corresponding to the slightly horizontal texture pattern (top and bottom right cluster), and the last one corresponding to the texture pattern slanted downward (bottom left cluster). Again, as for the previous image discussed, pixels near the edge tend to be misclassified.

In Figure 11a, the methods have difficulty clustering the texture patterns because the texture image itself has various texture patterns occurring at different sizes in various locations. For example, our results have a rectangular cluster located at the bottom of the image, but from the texture component, we observe that many types of texture patterns exist there. Furthermore, another cluster seems to correspond to the circular region in the texture pattern, but it also extends outside of it. Nevertheless, the other three clusters do correspond to identifiable texture patterns in the image. Two of the clusters correspond to texture patterns that slant upward, but one has finer texture while the other has sharper edges. The third cluster identifies with the horizontal texture pattern (top and bottom cluster).

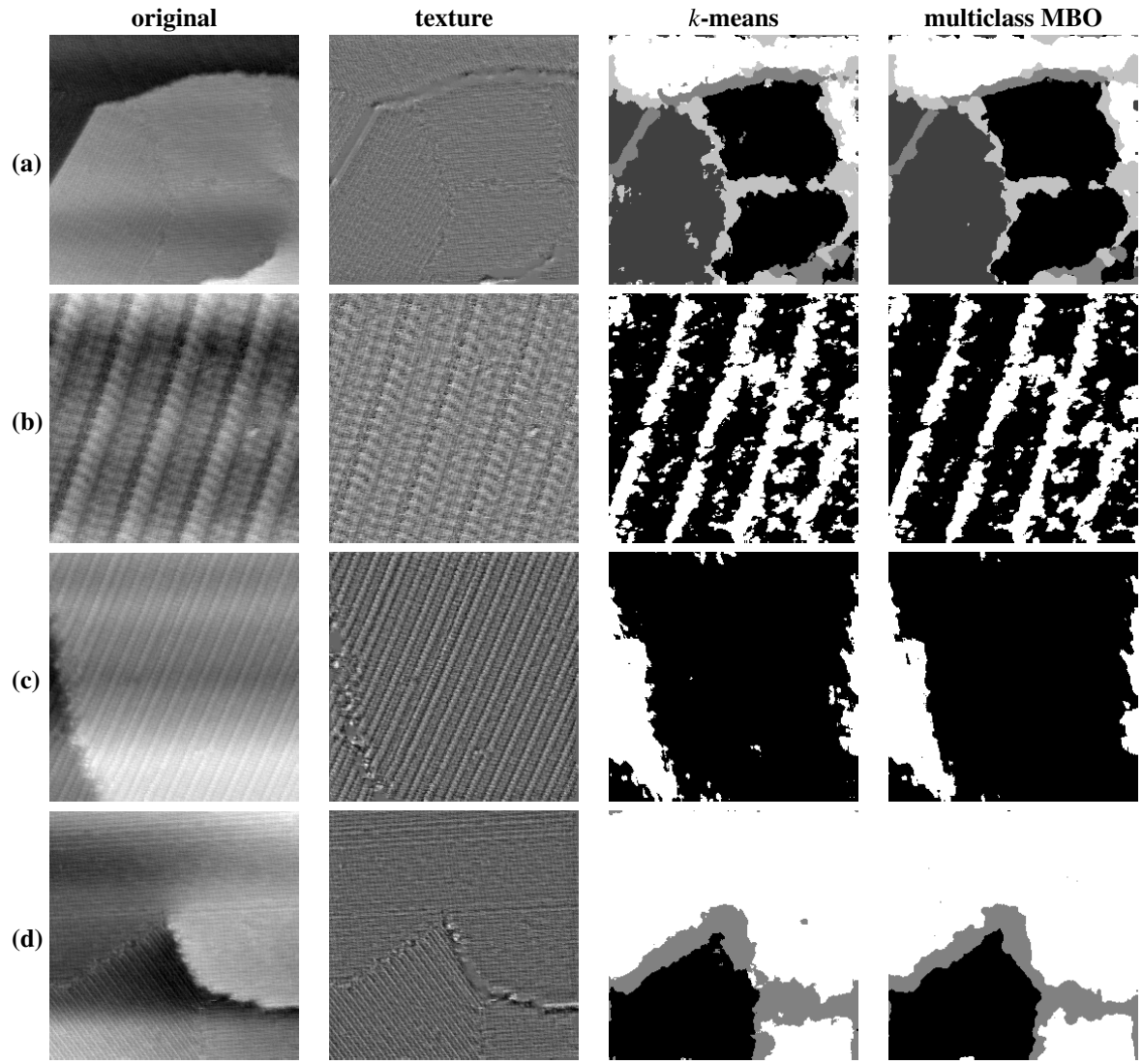


Fig. 10 Texture segmentation results. The used parameters are: (a) $\tau =$ 92nd percentile, k = 5, $dt = 0.03$. (b) $\tau =$ 99.50th percentile, k = 2, $dt = 0.10$. (c) $\tau =$ 98.80th percentile, $k = 2$, $dt = 0.05$. (d) $\tau =$ 85th percentile, $k = 3$, $dt = 0.05$. Raw scanning tunneling microscope images of cyanide on Au$\{111\}$, reproduced from Guttentag et al (2016a) with permission. Texture segmentation results in (a) were reproduced with permission from Guttentag et al (2016a). Images and segmentation results copyright American Chemical Society

In Figure 11b, we select $k = 4$. One cluster corresponds to the smoothest regions of the texture component. Another corresponds to the texture pattern slanted downward and with rough edges (bottom left cluster). The third cluster corresponds to the finer texture pattern slanted downward (top right, bottom left, and along the right edge of the elliptical smooth region). However, the last cluster seems to correspond to the miscellaneous region that does not have any apparent pattern.

In Figure 11c, we select $k = 2$, one cluster corresponding to the scaly texture pattern in the top right and the other one corresponding to the rough texture pattern that have some lines slanted downward. We observe that some pixels outside of the top right region belong to the same cluster, and most of them do look similar to the top right texture pattern as they look like scales.

In Figure 11d, we select $k = 4$. One cluster corresponds to the texture slanted downward and it is located at the bottom left and bottom middle of the image. Another cluster corresponds to the smooth regions of the image, which are located at the top left corner and most of the right side of the image. The other cluster seems to correspond to the rough edges, which are located mostly in the left of the image. The last cluster seems to correspond to miscellaneous patterns like in Figure 11b.

6 Conclusions

In this paper, we proposed a framework to segment STM images, combining variational methods and a clustering algorithm based on features extracted by the empirical wavelet transform. The expected information of microscopy images led us to first apply a cartoon+texture decomposi-

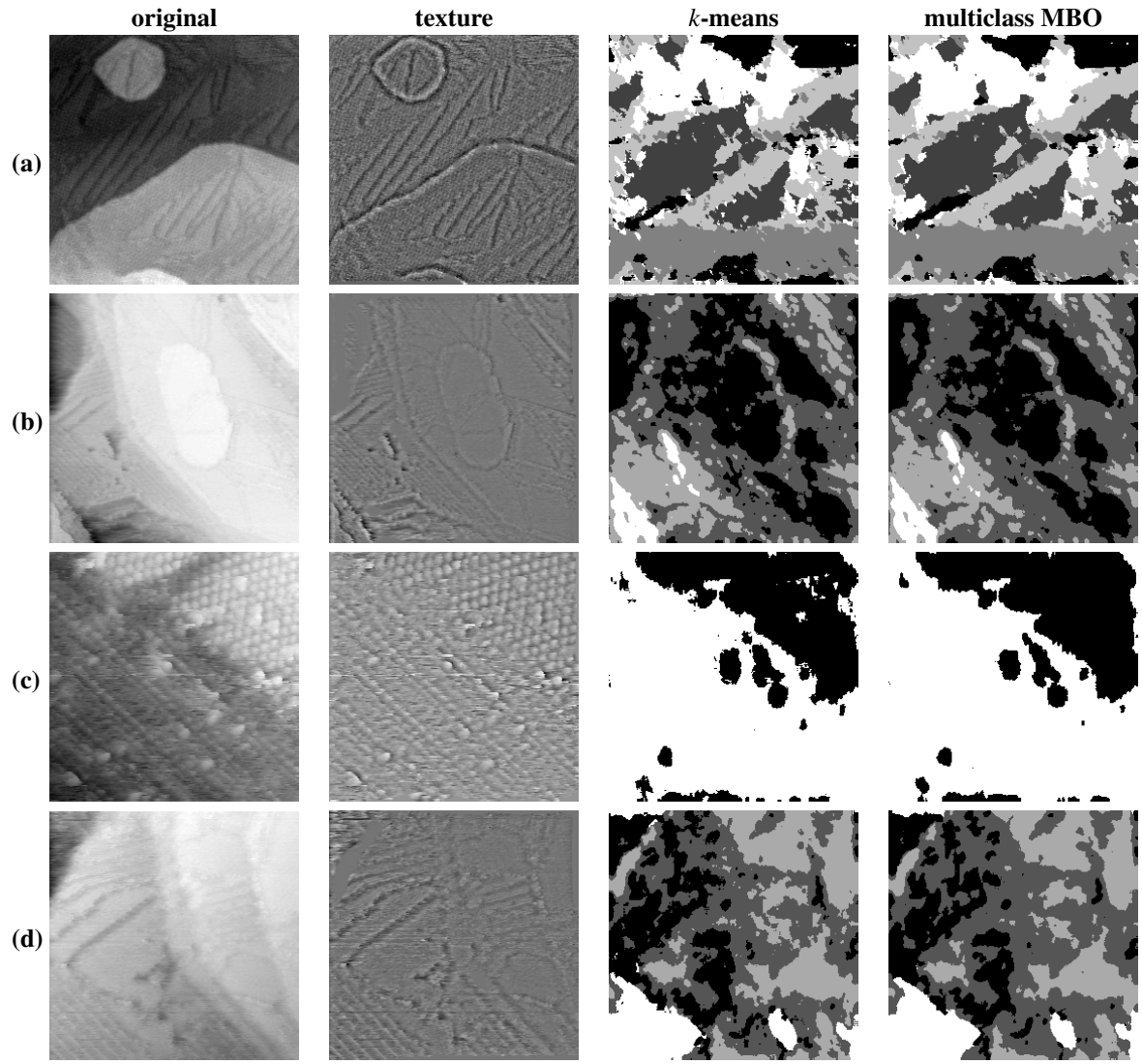


Fig. 11 Texture segmentation results. The parameters are **(a)** $\tau = 95.15$th percentile, $k = 5$, $dt = 0.10$ **(b)** $\tau = 45$th percentile, $k = 4$, $\tau = 0.10$ **(c)** $\tau = 85$th percentile, $k = 2$, $dt = 0.05$ **(d)** $\tau = 72.50$th percentile, $k = 4$, $dt = 0.05$. Raw scanning tunneling microscope images of cyanide on Au$\{111\}$, reproduced from Guttentag et al (2016a) with permission. Images copyright American Chemical Society

tion and then run a modified version of the multiphase CV model on the cartoon part, and a clustering of features extracted by the empirical curvelet transform on the texture part. The results in Section 5 demonstrate the proficiency of this framework to analyze STM images. Guttentag et al (2016a) have already used the proposed approach to characterize patterns of cyanide molecules on Au$\{111\}$, complementing the results in another related work (Guttentag et al 2016b).

There are several directions to investigate in order to improve the proposed framework. One direction is employing directional filters (Buades and Lisani 2016a,b) instead of isotropic filters L_σ in the cartoon+texture decomposition, which are designed to separate noise and micro-texture better in the proximity of edges. As such, in the proposed local MCV model, the Gaussian filter could be upgraded to a nonlocal filter, incorporating direction to better characterize weak edges. To gain robustness against low contrast and impulse noise, one could work with the L^1 fidelity term, but since the latter is non-differentiable at the origin, one would have to use convex optimization algorithms like primal-dual methods. Finally, in order to overcome artifacts (*e.g.*, scars, occluions, scratches), one could incorporate an indicator function as done by Zosso et al (2017a). As for the segmentation of the texture component, the use of other features than the local energy of the curvelet coefficient, such as the co-ocurrence matrices (Haralick et al 1973) constructed from the empirical curvelets, is likely to lead to more adapted clustering results.

Overall, the proposed framework has produced remarkable segmentation results of STM images using variants of the state-of-the-art image processing algorithms. One could modify and apply this framework to other kinds of images, thus leading to more interesting contributions and applications in other scientific areas beyond nanoscience.

Acknowledgements This work was supported by the W.M. Keck Foundation Center for Leveraging Sparsity. The self-assembly and STM imaging were supported by the U.S. Department of Energy (Grant #DE-SC-1037004). L. Torres Mandiola was partially supported by NSF DMS-1312361. A. Ciomaga, K. Bui, J. Fauman, and D. Kes were partially supported by NSF DMS-1045536. A. Bertozzi was supported by Simons Math + X Investigator Award number 510776.

References

Ambrosio L, Tortorelli VM (1990) Approximation of functional depending on jumps by elliptic functional via t-convergence. Communications on Pure and Applied Mathematics 43(8):999–1036

Arivazhagan S, Ganesan L (2003) Texture segmentation using wavelet transform. Pattern Recognition Letters 24(16):3197–3203

Arivazhagan S, Ganesan L, Kumar TS (2006) Texture classification using curvelet statistical and co-occurrence features. In: Pattern Recognition, 2006. ICPR 2006. 18th International Conference on, IEEE, vol 2, pp 938–941

Arthur D, Vassilvitskii S (2007) K-means++: the advantages of careful seeding. In: In Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms

Averbuch A, Coifman RR, Donoho DL, Elad M, Israeli M (2006) Fast and accurate polar Fourier transform. Applied and Computational Harmonic Analysis 21(2):145–167

Averbuch A, Coifman RR, Donoho DL, Israeli M, Shkolnisky Y (2008) A framework for discrete integral transformations I-The pseudopolar Fourier transform. SIAM Journal on Scientific Computing 30(2):764–784

Boyd S, Parikh N, Chu E, Peleato B, Eckstein J, et al (2011) Distributed optimization and statistical learning via the alternating direction method of multipliers. Foundations and Trends® in Machine Learning 3(1):1–122

Buades A, Lisani JL (2016a) Directional filters for cartoon+ texture image decomposition. Image Processing On Line 6:75–88

Buades A, Lisani JL (2016b) Directional filters for color cartoon+ texture image and video decomposition. Journal of Mathematical Imaging and Vision 55(1):125–135

Buades A, Le T, Morel JM, Vese L (2011) Cartoon+ texture image decomposition. Image Processing On Line 1

Bumm LA, Arnold JJ, Cygan MT, Dunbar TD, Burgin TP, Jones L, Allara DL, Tour JM, Weiss PS (1996) Are single molecular wires conducting? SCIENCE-NEW YORK THEN WASHINGTON- pp 1705–1707

Candes E, Demanet L, Donoho D, Ying L (2006) Fast discrete curvelet transforms. Multiscale Modeling & Simulation 5(3):861–899

Candes EJ, Donoho DL (2005) Continuous curvelet transform: I. resolution of the wavefront set. Applied and Computational Harmonic Analysis 19(2):162–197

Chambolle A (2004) An algorithm for total variation minimization and applications. Journal of Mathematical Imaging and Vision 20(1):89–97

Chan TF, Esedoglu S (2005) Aspects of total variation regularized L_1 function approximation. SIAM Journal on Applied Mathematics

Chan TF, Sandberg BY, Vese LA (2000) Active contours without edges for vector-valued images. Journal of Visual Communication and Image Representation 11(2):130–141

Claridge SA, Liao WS, Thomas JC, Zhao Y, Cao HH, Cheunkar S, Serino AC, Andrews AM, Weiss PS (2013) From the bottom up: dimensional control and characterization in molecular monolayers. Chemical Society Reviews 42(7):2725–2745

Dameron AA, Charles LF, Weiss PS (2005) Structures and displacement of 1-adamantanethiol self-assembled monolayers on Au $\{111\}$. Journal of the American Chemical Society 127(24):8697–8704

Dragomiretskiy K, Zosso D (2014) Variational mode decomposition. IEEE transactions on signal processing 62(3):531–544

Dragomiretskiy K, Zosso D (2015) Two-dimensional variational mode decomposition. In: International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition, Springer, pp 197–208

Dunn D, Higgins WE (1995) Optimal gabor filters for texture segmentation. IEEE Transactions on image processing 4(7):947–964

Dunn D, Higgins WE, Wakeley J (1994) Texture segmentation using 2-d gabor elementary functions. IEEE Transactions on Pattern Analysis and Machine Intelligence 16(2):130–149

Dunn DF, Higgins WE (1993) Optimal gabor-filter design for texture segmentation. In: Acoustics, Speech, and Signal Processing, 1993. ICASSP-93., 1993 IEEE International Conference on, IEEE, vol 5, pp 37–40

Esedoglu S, Tsai YHR (2006) Threshold dynamics for the piecewise constant Mumford–Shah functional. Journal of Computational Physics 211(1):367–384

Fowlkes C, Belongie S, Chung F, Malik J (2004) Spectral grouping using the Nyström method. IEEE Transactions on Pattern Analysis and Machine Intelligence 26(2):214–225

Gabay D, Mercier B (1976) A dual algorithm for the solution of nonlinear variational problems via finite element approximation. Computers & Mathematics with Applications 2(1):17–40

Garcia-Cardona C, Merkurjev E, Bertozzi AL, Flenner A, Percus AG (2014) Multiclass data segmentation using diffuse interface methods on graphs. IEEE Transactions on Pattern Analysis and Machine Intelligence 36(8):1600–1613

Getreuer P (2012) Chan-Vese segmentation. Image Processing On Line 2:214–224

Gilles J (2013) Empirical Wavelet Transform. IEEE Transactions on Signal Processing 61(16):3999–4010, DOI 10.1109/TSP.2013.2265222

Gilles J, Heal K (2014) A parameterless scale-space approach to find meaningful modes in histograms - application to image and spectrum segmentation. Interna-

tional Journal of Wavelets, Multiresolution and Information Processing 12(6):1450,044–1–1450,044–17, DOI 10.1142/S0219691314500441

Gilles J, Tran G, Osher S (2014) 2D empirical transforms. Wavelets, ridgelets and curvelets revisited. SIAM Journal on Imaging Sciences 7(1):157–186, DOI 10.1137/130923774

Glowinski R, Marroco A (1975) Sur l'approximation, par éléments finis d'ordre un, et la résolution, par pénalisation-dualité d'une classe de problèmes de dirichlet non linéaires. Revue française d'automatique, informatique, recherche opérationnelle Analyse numérique 9(R2):41–76

Gooding JJ, Mearns F, Yang W, Liu J (2003) Self-assembled monolayers into the 21st century: recent advances and applications. Electroanalysis 15(2):81–96

Guo K, Kutyniok G, Labate D (2006) Sparse multidimensional representations using anisotropic dilation and shear operators

Guttentag AI, Barr KK, Song TB, Bui KV, Fauman JN, Torres LF, Kes DD, Ciomaga A, Gilles J, Sullivan NF, Yang Y, Allara DL, Zharnikov M, Weiss PS (2016a) Hexagons to ribbons: Flipping cyanide on Au $\{111\}$. Journal of the American Chemical Society 138(48):15,580–15,586

Guttentag AI, Wachter T, Barr KK, Abendroth JM, Song TB, Sullivan NF, Yang Y, Allara DL, Zharnikov M, Weiss PS (2016b) Surface structure and electron transfer dynamics of the self-assembly of cyanide on Au $\{111\}$. The Journal of Physical Chemistry C 120(47):26,736–26,746

Haralick RM, Shanmugam K, Dinstein I (1973) Textural features for image classification. IEEE Transactions on Systems, Man, and Cybernetics 3(6):610–621

Jain AK, Farrokhnia F (1991) Unsupervised texture segmentation using gabor filters. Pattern recognition 24(12):1167–1186

Kim M, Hohman JN, Cao Y, Houk KN, Ma H, Jen AKY, Weiss PS (2011) Creating favorable geometries for directing organic photoreactions in alkanethiolate monolayers. Science 331(6022):1312–1315

Love JC, Estroff LA, Kriebel JK, Nuzzo RG, Whitesides GM (2005) Self-assembled monolayers of thiolates on metals as a form of nanotechnology. Chemical Reviews 105(4):1103–1170

March R (1992) Visual reconstruction with discontinuities using variational methods. Image and Vision Computing 10(1):30–38

Merkurjev E, Garcia-Cardona C, Bertozzi AL, Flenner A, Percus AG (2014) Diffuse interface methods for multiclass segmentation of high-dimensional data. Applied Mathematics Letters 33:29–34, DOI 10.1016/j.aml.2014.02.008

Merriman B, Bence JK, Osher S (1992) Diffusion generated motion by mean curvature. In: Proceedings of the Geometry Center Workshop

Merriman B, Bence JK, Osher SJ (1994) Motion of multiple junctions: A level set approach. Journal of Computational Physics 112(2):334–363

Meyer Y (2001) Oscillating Patterns in Image Processing and Nonlinear Evolution Equations: The Fifteenth Dean Jacqueline B. Lewis Memorial Lectures. American Mathematical Society, Boston, MA, USA

Modica L (1987) The gradient theory of phase transitions and the minimal interface criterion. Archive for Rational Mechanics and Analysis 98(2):123–142

Modica L, Mortola S (1977) Un esempio di Γ–convergenza. Bollettino della Unione Matematica Italiana B 14(5):285–299

Mumford D, Shah J (1989) Optimal approximations by piecewise smooth functions and associated variational problems. Communications on Pure and Applied mathematics 42(5):577–685

Nuzzo RG, Allara DL (1983) Adsorption of bifunctional organic disulfides on gold surfaces. Journal of the American Chemical Society 105(13):4481–4483

Osher S, Sethian JA (1988) Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations. Journal of Computational Physics 79(1):12–49

Poirier GE (1997) Characterization of organosulfur molecular monolayers on Au (111) using scanning tunneling microscopy. Chemical Reviews 97(4):1117–1128

Rudin LI, Osher S, Fatemi E (1992) Nonlinear total variation based noise removal algorithms. Physica D: Nonlinear Phenomena 60(1-4):259–268

Shen L, Yin Q (2009) Texture classification using curvelet transform. In: Proceedings of the International Symposium on Information Processing (ISIP09), Citeseer, pp 319–324

Smith RK, Lewis PA, Weiss PS (2004) Patterning self-assembled monolayers. Progress in Surface Science 75(1):1–68

Strang G (1993) Wavelet transforms versus fourier transforms. Bulletin of the American Mathematical Society 28(2):288–305

Thomas JC, Schwartz JJ, Hohman JN, Claridge SA, Auluck HS, Serino AC, Spokoyny AM, Tran G, Kelly KF, Mirkin CA, Gilles J, Osher SJ, Weiss PS (2015) Defect-tolerant aligned dipoles within two-dimensional plastic lattices. ACS Nano 9(5):4734–4742

Thomas JC, Goronzy DP, Dragomiretskiy K, Zosso D, Gilles J, Osher SJ, Bertozzi AL, Weiss PS (2016) Mapping buried hydrogen-bonding networks. ACS Nano 10(5):5446–5451

Unser M (1995) Texture classification and segmentation using wavelet frames. IEEE Transactions on image processing 4(11):1549–1560

Vese LA, Chan TF (1997) Reduced non-convex functional approximations for image restoration & segmentation. UCLA CAM Reports 97-56,

Vese LA, Chan TF (2002) A multiphase level set framework for image segmentation using the Mumford and Shah model. International Journal of Computer Vision 50(3):271–293

Wang XF, Huang DS, Xu H (2010) An efficient local Chan-Vese model for image segmentation. Pattern Recognition 43(3):603–618

Weldon TP, Higgins WE, Dunn DF (1996) Efficient gabor filter design for texture segmentation. Pattern Recognition 29(12):2005–2015

Zosso D, An J, Stevick J, Takaki N, Weiss M, Slaughter LS, Cao HH, Weiss PS, Bertozzi AL (2017a) Image segmentation with dynamic artifacts detection and bias correction. Inverse Problems & Imaging 11(3)

Zosso D, Dragomiretskiy K, Bertozzi AL, Weiss PS (2017b) Two-dimensional compact variational mode decomposition. Journal of Mathematical Imaging and Vision 58(2):294–320