

Research Article

Extracting Filaments Based on Morphology Components Analysis from Radio Astronomical Images

M. Zhu,¹ W. Liu,¹ B. Y. Wang,¹ M. F. Zhang,² W. W. Tian,³ X. C. Yu ,¹ T. H. Liang,¹ D. Wu,² D. Hu,¹ and F. Q. Duan ¹

¹College of Information Science and Technology, Beijing Normal University, Beijing, China

²Key Laboratory of Optical Astronomy, National Astronomical Observatory of China, Beijing, China

³The University of Chinese Academy of Sciences, Beijing, China

Correspondence should be addressed to X. C. Yu; yuxianchuan@163.com

Received 17 October 2018; Accepted 3 March 2019; Published 2 June 2019

Guest Editor: Junhui Fan

Copyright © 2019 M. Zhu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Filaments are a type of wide-existing astronomical structure. It is a challenge to separate filaments from radio astronomical images, because their radiation is usually weak. What is more, filaments often mix with bright objects, e.g., stars, which makes it difficult to separate them. In order to extract filaments, A. Men'shchikov proposed a method “getfilaments” to find filaments automatically. However, the algorithm removed tiny structures by counting connected pixels number simply. Removing tiny structures based on local information might remove some part of the filaments because filaments in radio astronomical image are usually weak. In order to solve this problem, we applied morphology components analysis (MCA) to process each single spatial scale image and proposed a filaments extraction algorithm based on MCA. MCA uses a dictionary whose elements can be wavelet translation function, curvelet translation function, or ridgelet translation function to decompose images. Different selection of elements in the dictionary can get different morphology components of the spatial scale image. By using MCA, we can get line structure, gauss sources, and other structures in spatial scale images and exclude the components that are not related to filaments. Experimental results showed that our proposed method based on MCA is effective in extracting filaments from real radio astronomical images, and images processed by our method have higher peak signal-to-noise ratio (PSNR).

1. Introduction

A substantial part of interstellar medium exists in the form of a fascinating web of omnipresent filamentary structures [1], called filaments. The astronomical filament is first discovered in the Milky Way. Along with the development of telescopes, various filaments come into sight. Among them the filaments in star-forming regions are the most fascinating, many magnetohydrodynamic (MHD) simulations have shown that giant molecular clouds (GMCs) primarily evolve into filaments before they collapse to form stars [2, 3]. Recent observations also confirm these simulations [4, 5]. Since the formation of massive stellar objects is still unclear, further research on filaments is essential. Filaments in Galactic and cosmological fields are also important. Studies have argued that low mass galaxies got their gas through “cold accretion”, which is often directed along filaments [6, 7].

Some researchers paid more attention to the large-scale filaments of the universe [8], which may give clues to better understand the slightly nonuniform cosmic microwave background (CMB) and the birth of the first generation of stars. In addition, filaments have been observed in other objects, such as supernova remnants (SNR) [9] and protoplanetary disk [10].

The fact that many filaments are fuzzy in images causes difficulty to distinguish them from background and surrounding objects. Schneider et al. [11] investigated spatial and density structure of the Rosette molecular cloud, by applying a curvelet analysis, a filament-tracing algorithm (DisPerSE), and probability density functions (PDFs) on Herschel column density maps. Hennebelle et al. [12] showed a method based on adaptive mesh refinement magneto hydrodynamic simulations, which treat self-consistently cooling and self-gravity. Tugay [13] proposed a layer smoothing method,

which described cellular large-scale structure of the universe (LSS) as a grid of clusters with density larger than a limited value, to detect extragalactic filaments. Men'shchikov [14] proposed a multi-scale filaments extraction method named getfilaments, which decomposed a simulated astronomical image containing filaments into spatial images at different scales to prevent interaction influence of different spatial scale structures. The getfilaments works well in simulated images and has been used to identify filaments for real astronomical images, e.g., the far-infrared images of Musca cloud observed with Herschel [15]. However, getfilaments might exclude some tiny structure of filaments in astronomy images, because it removes tiny structures just by counting connected pixels number, and filaments in astronomy images are usually weak.

In this paper, we develop an improved method based on morphology components analysis (MCA) and getfilaments. MCA is able to decompose the image into morphological components based on different features from the perspective of mathematical morphology and is often used in image restoration, separation, and decomposition [16–19]. The basic idea of MCA decomposition algorithm is to choose two dictionaries: smooth dictionary and texture dictionary, to represent morphology components [20]. We can design different dictionaries to represent different sparse components in the image. Smooth dictionary produces the decomposed smooth component which carries the geometric and piecewise smooth information of the image, and texture dictionary produces the decomposed texture component which carries the marginal and edge information.

The paper is structured as follows. The improved method named filament extraction algorithm based on MCA is described in Section 2. Section 3 is devoted to discussing experimental results of our method and comparing our method with the getfilaments method by employing data from GALFA-HI of Arecibo.

2. Filament Extraction Algorithm Based on MCA

2.1. MCA Model. MCA was proposed by Starck et al. [17]. MCA is a kind of decomposition algorithm based on signal sparsity and morphological diversity. MCA assumes that signals are linear combinations of several morphological components, and each morphological component can be sparsely represented on its own dictionary.

We assume that image \mathbf{x} comprises M different morphological components: $\mathbf{x} = \mathbf{x}_1 + \mathbf{x}_2 + \dots + \mathbf{x}_M$. We design different dictionaries \mathbf{D}_i for different morphological components \mathbf{x}_i and assume all components mix together linearly. The image \mathbf{x} as an one-dimensional vector of length M can then be represented as follows:

$$\mathbf{x} = \mathbf{D}\boldsymbol{\alpha}, \quad (1)$$

where the matrix $\mathbf{D} = [D_1, \dots, D_p] \in R^{M \times P}$ (typically, $M \ll P$) is a dictionary. $\boldsymbol{\alpha} \in R^P$ is the vector of sparse coefficients.

The equivalent constrained optimization problem is as follows:

$$\begin{aligned} & \{\boldsymbol{\alpha}_1^{\text{opt}}, \boldsymbol{\alpha}_2^{\text{opt}}, \dots, \boldsymbol{\alpha}_p^{\text{opt}}\} \\ & = \arg \min_{\{\boldsymbol{\alpha}_1, \dots, \boldsymbol{\alpha}_p\}} \sum_{k=1}^P \|\boldsymbol{\alpha}_k\|_1, \quad (2) \\ \text{subject to : } & \mathbf{x} = \sum_{k=1}^P \mathbf{D}_k \boldsymbol{\alpha}_k. \end{aligned}$$

However, this model does not take into account factors that may lead to the failure of the image decomposition, such as noise. When noise exists in the image \mathbf{x} , the vector $\{\boldsymbol{\alpha}_1^{\text{opt}}, \boldsymbol{\alpha}_2^{\text{opt}}, \dots, \boldsymbol{\alpha}_p^{\text{opt}}\}$ might be not sparse since noise cannot be sparsely represented. For this kind of noise, we put the noise in the error item to achieve the sparse decomposition of the image \mathbf{x} . The constraint in (2) is modified as follows:

$$\begin{aligned} & \{\boldsymbol{\alpha}_1^{\text{opt}}, \boldsymbol{\alpha}_2^{\text{opt}}, \dots, \boldsymbol{\alpha}_p^{\text{opt}}\} \\ & = \arg \min_{\{\boldsymbol{\alpha}_1, \dots, \boldsymbol{\alpha}_p\}} \sum_{k=1}^P \|\boldsymbol{\alpha}_k\|_1, \quad (3) \\ \text{subject to : } & \left\| \mathbf{x} - \sum_{k=1}^P \mathbf{D}_k \boldsymbol{\alpha}_k \right\|_1 \leq \epsilon. \end{aligned}$$

where ϵ represents the noise level in the image \mathbf{x} .

2.2. MCA Decomposition Algorithm. In this paper, we focus on the image decomposition into two components: cartoon layer and texture layer. Cartoon layer contains cartoon and piecewise smooth information, and texture layer may contain other texture information, marginal information, and noises [21, 22]. Studies [23, 24] have shown that noises exist in both cartoon and texture layer. In other words, the smooth part not only contains the majority of the useful information, but also contains a small part of the noise. If we set the same threshold of noise variance for the whole image, rather than calculating the threshold for each part of the image, some useful information might be removed. We therefore introduce the MCA decomposition algorithm to process an image into smooth (cartoon) layer and texture layer.

We assume that matrix \mathbf{D}_t is the dictionary matrix of the texture layer and that \mathbf{D}_c is the dictionary matrix of the cartoon layer. A solution for the decomposition could be obtained by relaxing the constraint in (3) to become an approximate one:

$$\begin{aligned} \{\boldsymbol{\alpha}_t^{\text{opt}}, \boldsymbol{\alpha}_c^{\text{opt}}\} & = \arg \min_{\{\boldsymbol{\alpha}_t, \boldsymbol{\alpha}_c\}} \|\boldsymbol{\alpha}_t\|_1 + \|\boldsymbol{\alpha}_c\|_1 \\ & + \lambda \|\mathbf{x} - \mathbf{D}_t \boldsymbol{\alpha}_t - \mathbf{D}_c \boldsymbol{\alpha}_c\|_2^2, \quad (4) \end{aligned}$$

where λ is a Lagrange operators. Define $\mathbf{x}_t = \mathbf{D}_t \boldsymbol{\alpha}_t$ and $\mathbf{x}_c = \mathbf{D}_c \boldsymbol{\alpha}_c$. Given \mathbf{x}_t , we can recover $\boldsymbol{\alpha}_t$ as $\boldsymbol{\alpha}_t = \mathbf{D}_t^+ \mathbf{x}_t$, where \mathbf{D}_t^+ is the Moore-Penrose pseudoinverse of \mathbf{D}_t . In order to get

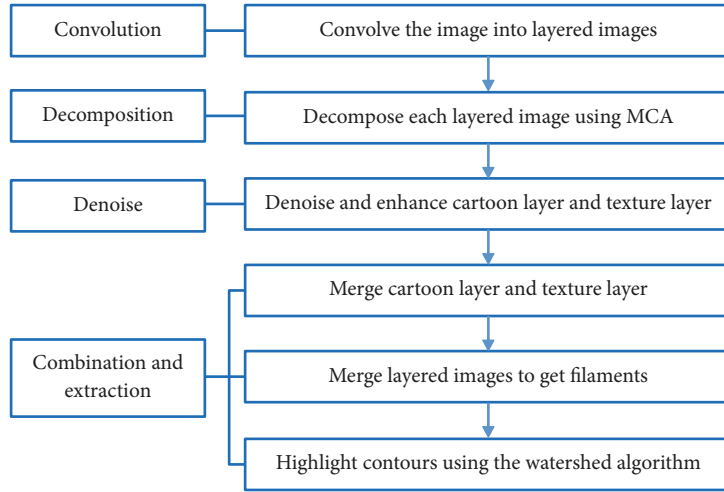


FIGURE 1: Flow of the filament extraction algorithm.

piecewise smooth component, add a TV (Total Variation) penalty [25] to fit the smooth layer. TV is used to damp ringing artifacts near edges and oscillating. Put these back into (4), and, thus, we obtain the following:

$$\{\mathbf{x}_t^{\text{opt}}, \mathbf{x}_c^{\text{opt}}\} = \arg \min_{\{\mathbf{x}_t, \mathbf{x}_c\}} \|\mathbf{D}_t^+ \mathbf{x}_t\|_1 + \|\mathbf{D}_c^+ \mathbf{x}_c\|_1 + \lambda \|\mathbf{x} - \mathbf{x}_t - \mathbf{x}_c\|_2^2 + \gamma \|TV(\mathbf{x}_c)\|. \quad (5)$$

γ is the TV regularization parameter, in multiscale method, a lower γ is able to remove the artefacts caused by curvelet. $TV(\mathbf{x}_c)$ is a measure of the amount of oscillations in the cartoon layer. Penalizing with TV, the cartoon layer is closer to the piecewise smooth image. However, TV suffers from the so-called staircase effect that impacts the quality of images reconstruction. The adaptive TV [26] and the higher order derivative [27] are solutions to reduce the staircase effect.

Then we discuss the choice of dictionaries for the cartoon and the texture layer. Appropriate dictionaries are very important for sparse representations over the image. Generally, the choice of dictionaries depends on experiences. The structure in the dictionary is more matched with the image easier to form a sparse representation. The commonly used dictionary of MCA includes wavelet transform, ridgelet transform, curvelet transform, discrete cosine transform (DCT), and so on. The two dictionaries used in this paper are described as follows.

First, we choose curvelet as the dictionary for cartoon layer. The curvelet transform based on the multiscale ridgelet transform was proposed by Cand & Donoho [28]. It first decomposes the image into a set of wavelet bands and then analyzes each band with the ridgelet transform at different scale levels. The curvelet transform performs well at the detection of anisotropic structures, smooth curves, and edges of different lengths [29].

We next choose the local DCT as the dictionary for texture layer. DCT is a variant of the discrete Fourier transform (DFT). It uses a symmetric signal extension to replace

complex analysis with real numbers and is appropriate for the sparse representation of the texture and periodic part of the image.

2.3. Filament Extraction Algorithm Based on MCA. The shape of the filaments can be obtained by applying the above filament extraction algorithm to radio astronomical images. We finally use the watershed algorithm to highlight filaments.

The watershed algorithm [30, 31] is based on mathematical morphology. The watershed algorithm segments an image into nonoverlapping regions and gets a pixel width and continuous boundary for the purpose of extracting and identifying a specific area [32, 33]. A grayscale image can be viewed as a topographic surface. A high grayscale value of a pixel denotes a peak or hill while a low grayscale denotes a valley. Each local minimum of pixels and the affected region are called a catchment basin, and the boundary of catchment basins forms the watershed. By filling each isolated valley (local minimum) with differently colored water (labels), the region of influence of each local minimum gradually expands outwards. The adjacent regions then converge, and the boundaries that form the watershed appear.

The whole filament extraction algorithm (<https://github.com/MiWBY/MCA>) can be roughly divided into four steps (as shown in Figure 1).

(1) *Convolution.* First, using a Gaussian filter, we convolve the original images into a series of layered images. Different full widths at half maximum can be set for different image layers:

$$X_j = G_{j-1} * X - G_j * X \quad (j = 1, 2, \dots, N_s). \quad (6)$$

where X is the original image, X_j is the j th subimage after convolution, G_{j-1} and G_j are different Gaussian beams for different spatial components, $*$ is the convolution operation, and N_s is the number of the layers.

In this process, structures at different scales in the astronomical images can be separated into different layers (subimages), and each layer contains similar scales, which

make the input sources become simpler in the later denoising and extraction process.

(2) *Decomposition*. We apply the MCA algorithm to each layered image so that each layered image is decomposed into a cartoon layer and a texture layer. Here we use the curvelet as the dictionary for the cartoon layer and local DCT as the dictionary for the texture layer as described in Section 2.2. The cartoon layer contains most of filaments and low-frequency noise, and the texture layer contains sources, high-frequency noise, and small part of filaments.

Starck et al. [17] proposed the MCA decomposition algorithm based on BCR algorithm (Block Coordinate Relaxation). The algorithm is given as follows. Input: The subimage X_j after convolution, which is described as the input image \mathbf{x} here, dictionary \mathbf{D}_c of the cartoon layer, dictionary \mathbf{D}_t of the texture layer, number of iterations L_{\max} , and the threshold $\delta = \lambda \cdot L_{\max}$.

Output: Cartoon layer \mathbf{x}_c and texture layer \mathbf{x}_t .

(1) Initialize L_{\max} , and $\lambda = k * \epsilon$ (typically, $k = 3$), where ϵ is the value of noise level. Then the threshold $\delta = \lambda \cdot L_{\max}$.

(2) For $j = 1 : L_{\max}$

For $k = 1 : P$

(i) Update \mathbf{x}_c assuming \mathbf{x}_t is fixed:

(a) Calculate the residual $\mathbf{r} = \mathbf{x} - \mathbf{x}_c - \mathbf{x}_t$.

(b) Calculate $\boldsymbol{\alpha}_c = \mathbf{D}_c^+(\mathbf{x}_c + \mathbf{r})$.

(c) Soft thresholding the coefficient $\boldsymbol{\alpha}_c$ with the δ threshold and obtain $\hat{\boldsymbol{\alpha}}_c$.

(d) Reconstruct \mathbf{x}_c by $\mathbf{x}_c = \mathbf{D}_c \hat{\boldsymbol{\alpha}}_c$.

(ii) Update \mathbf{x}_t with the above method

Apply the TV correction by $\mathbf{x}_c = \mathbf{x}_c - \mu \gamma (TV(\mathbf{x}_c)/\mathbf{x}_c)$, where μ is the minimum parameter, and is chosen by a line-decreasing the overall penalty function, or as a fixed step-size of moderate value that guarantees convergence.

(3) Update the threshold by $\gamma = \gamma - \lambda$.

If $\gamma > \lambda$, return step 2. Else, finish.

(3) *Denoise*. First, we denoise each layer using the iterative cleaning algorithm proposed by Men'shchikov et al. [34]. The cleaning algorithm employs a global intensity threshold for single-scale images, as the larger-scale background has been effectively filtered out by the spatial decomposition. This iterative algorithm automatically finds a cut-off level that separates the signal of important sources from the noise and background at each scale. Next, we enhance details for both the cartoon layer and texture layer.

(4) *Combination and Extraction*. To get the extracted filaments, we first merge the cartoon layer and textual layer for each layered image. Because filaments are irregular, and structures of filaments exist in both cartoon layer and texture layer, it is not appropriate to use just one component to

represent the filaments. For example, if we just use cartoon layer to represent filaments, filaments may lose some texture. Thus, we merge the cartoon layer and textual layer to represent filaments better. Next, the layered subimages are added together to produce the filaments. Finally, we apply the watershed algorithm to highlight the contours of filaments.

By applying MCA to decompose a real image, new features (components) can be obtained. This leads to better image separability. Furthermore, the smooth components have a better signal-to-noise ratio than the original image.

3. Extraction Results

3.1. *Results for a Simulated Image*. Before applying our method to real radio astronomical images, we simulated an image that is composed of a straight filament with $37''$ size of FWHM, a string of sources with $24''$ size of FWHM, a simple background with $4000''$ size of FWHM, and a moderate-level noise with noise level=1.05 to test the improved algorithm (Figure 2(a)). The simulation method is the same as that mentioned in Men'shchikov et al. [14]. In the simulated image, there is only one spatial component, while our method assumes there are many spatial components, which is similar to real astronomical images. In other words, even if there is only one spatial component, our method will also treat it as many components.

We first extract filaments using MCA method without convolution and denoising (Figure 2). In Figure 2(c), texture layer (especially in the area marked by red box) still contains part of filaments structures, which means just using cartoon layer to represent filaments is insufficient. So it is necessary to combine cartoon layer and texture layer. However, noises and sources also exist in texture layer. If the two layers are combined directly, the reconstructed filaments contains noises and sources (Figure 2(d)), so denoising is necessary before combination.

Next, extracted results obtained using our improved method are shown in Figure 3. Compared to Figure 2(d), the reconstructed filaments in Figure 3(e) contain less noise. The edge of the filament is unreal as the result of decomposition.

3.2. Results for Astronomical Images

3.2.1. *Decomposition and Denosing Results*. Aiming at real radio astronomical images, we compare the extraction results of our method with those of the getfilaments method. We employ data from GALFA-HI of Arecibo as example images. The equatorial coordinates of the objects are (12.00h, +10.35°), and the object name is 'GALFA-HI RA+DEC Tile 004.00+02.35'. The data cube contains 2048 images at different velocity (with respect to the local standard stationary system). Here we select the 715th image from the 2048 original images as the experimental image (Figure 4). In the experimental images, filaments describe significantly elongated structures. After convolution of the 715th image, we obtain 99 layers (subimages) at different scales (Figure 5) and select the 40th subimage as a comparison example (Figure 5(b)). In order to display the image properly and improve visual

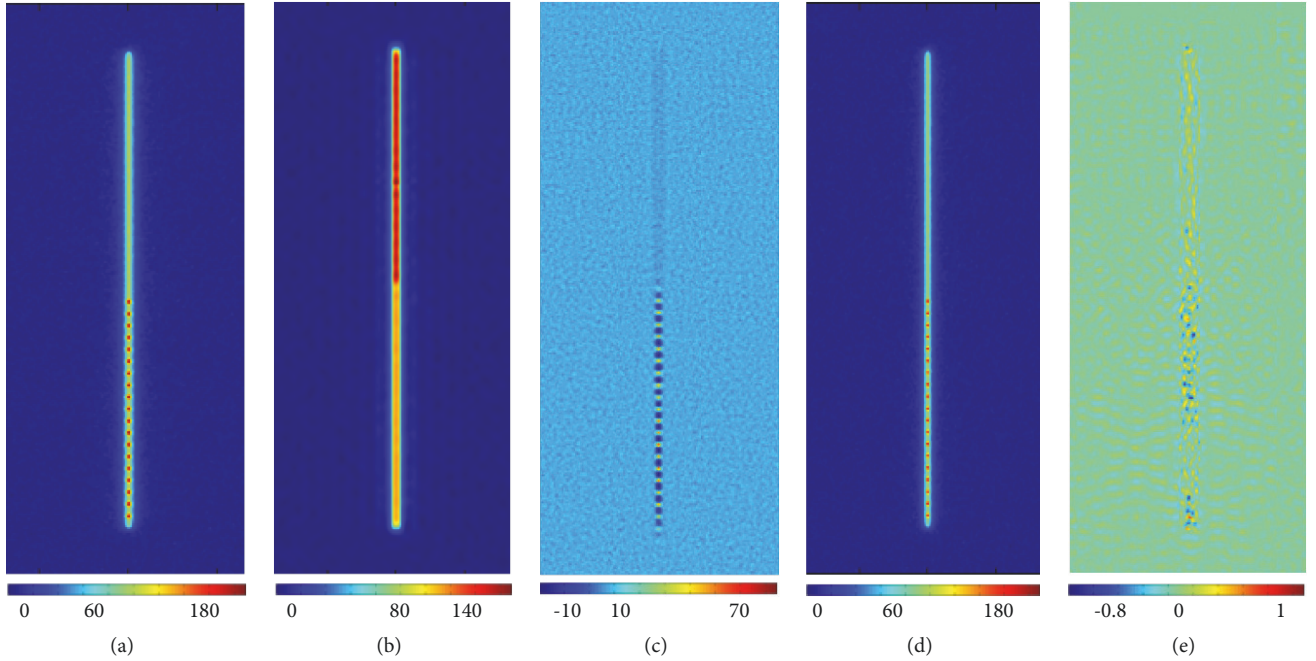


FIGURE 2: Results of the simulated images obtained using MCA without convolution and denoising. (a) Original simulated image. (b) Cartoon layer. (c) Texture layer. (d) Reconstructed filaments without denoising. (e) Residuals.

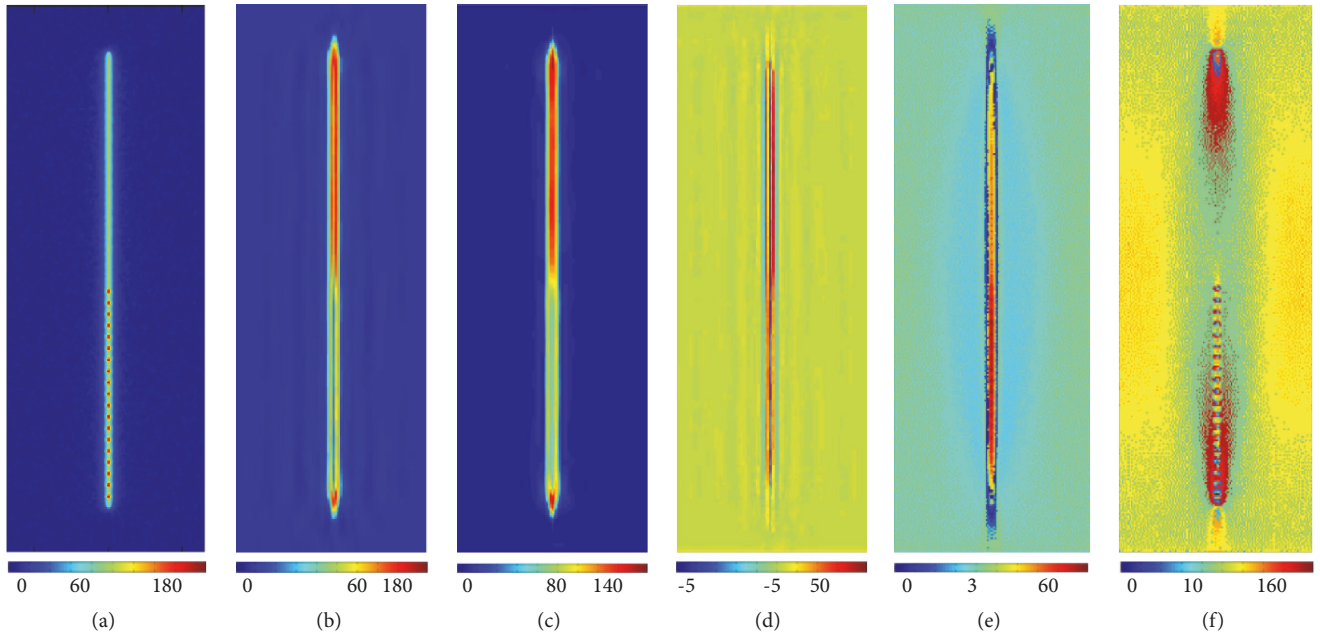


FIGURE 3: Extraction results for the simulated image obtained using our method. (a) Original simulated image. (b) The 40th subimage after convolution. (c) Cartoon layer of the 40th subimage after decomposition. (d) Texture layer of 40th subimage after decomposition and denoising. (e) Reconstructed filaments. (f) Residuals.

contrast between getfilaments and our method, we mark the image with different colors according to the intensity (Unit: MJy/sr) in the image.

First, we apply the MCA algorithm to process image layers before applying the iterative cleaning algorithm. As described in Section 2.2, we choose curvelet as the cartoon

layer dictionary and LDCT as the texture layer dictionary for MCA. We decompose each layered image to get the cartoon layer and texture layer (Figure 6). The cartoon layer contains smooth parts of the image and retains most of the low-frequency information of the filament in the layered image. The frequency of the texture layer is higher. The texture

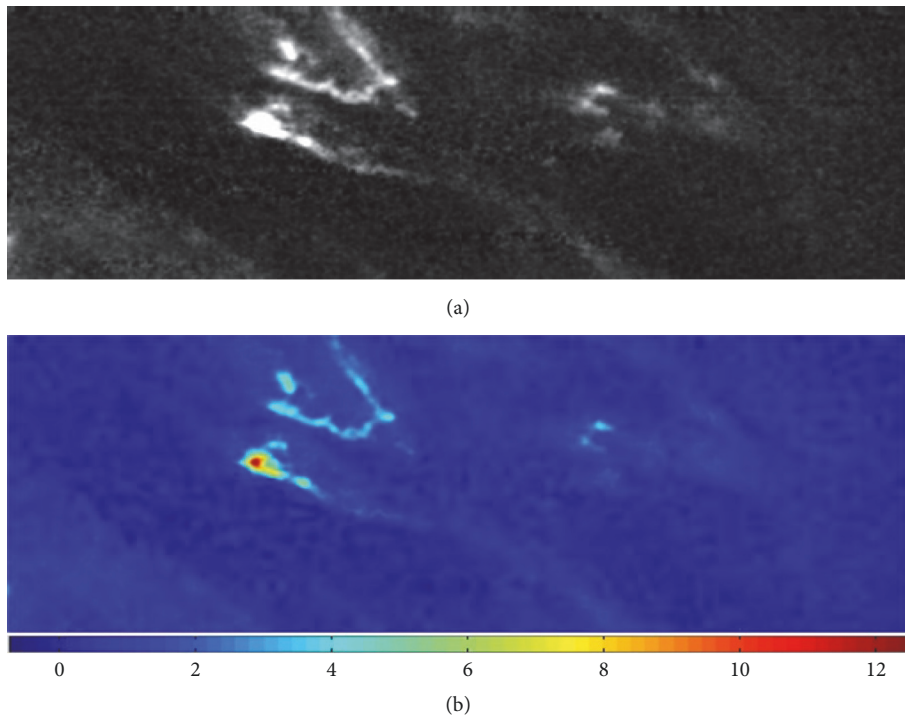


FIGURE 4: The 715th original image from GALFA-HI. (a) Original image for experiments. (b) Colored image for better visual contrast.

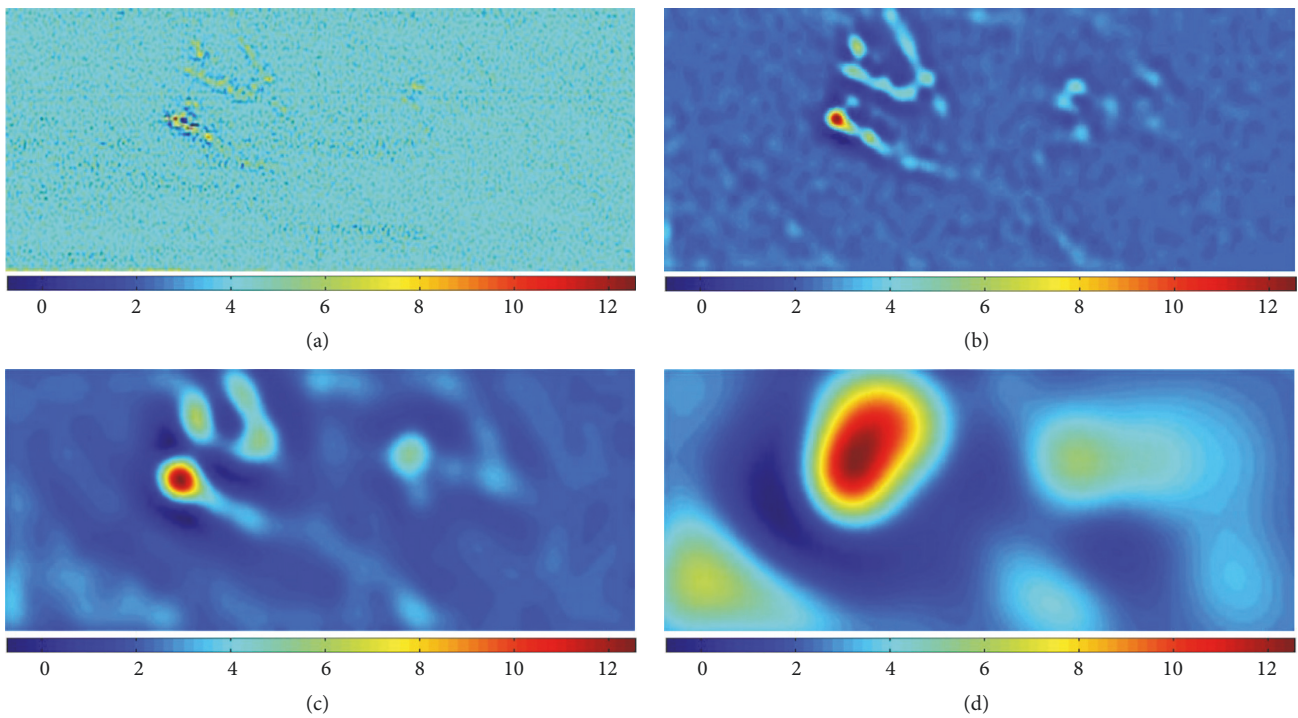


FIGURE 5: Images at different scales after convolution of the 715th image. Choose the 40th subimage as comparison example. (a) The 1st subimage. (b) The 40th subimage. (c) The 60th subimage. (d) The 80th subimage.

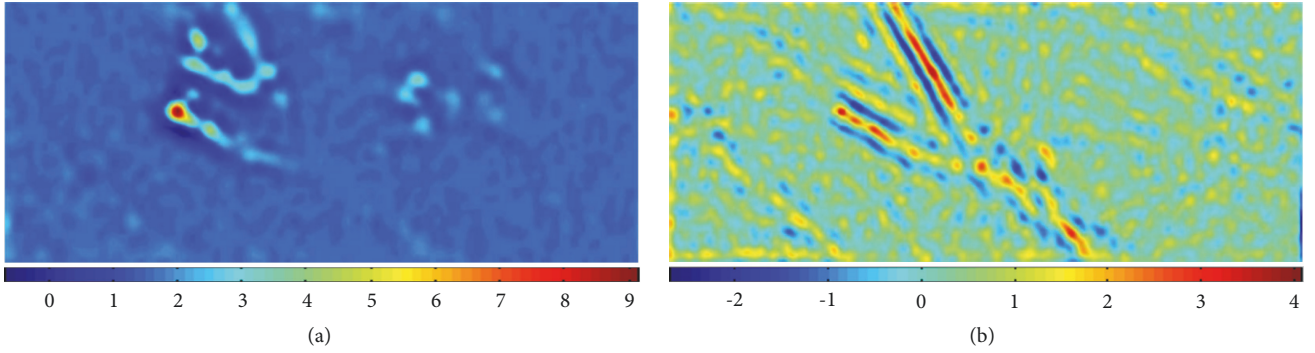


FIGURE 6: Decomposition results of the 40th subimage obtained using MCA. (a) The cartoon layer obtained using the MCA algorithm. (b) The texture layer obtained using the MCA algorithm.

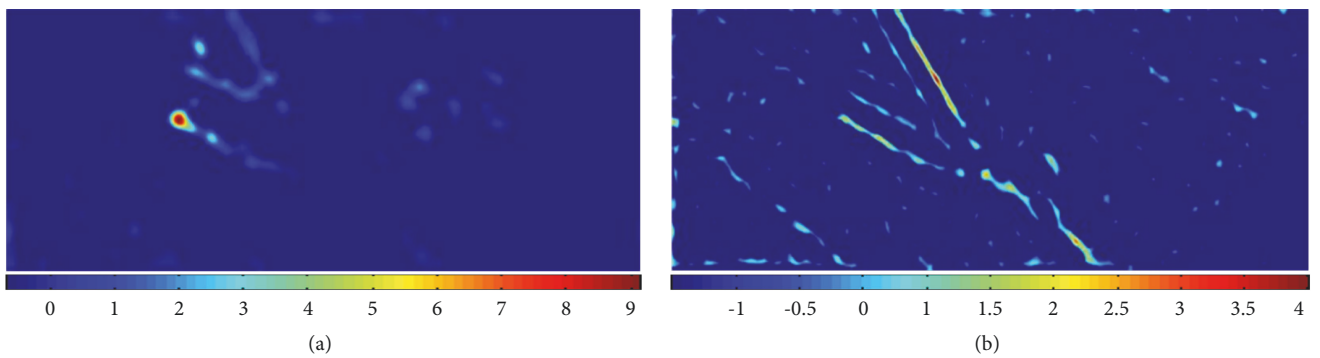


FIGURE 7: Denoising results for the cartoon layer and texture layer obtained using the iterative cleaning algorithm. (a) Denoising results for the cartoon layer of the 40th subimage. (b) Denoising results for the texture layer of the 40th subimage.

layer contains more edge information that is difficult to be distinguished visually in the layered image. The texture layer is also part of the filament. The texture layer also contains part of the filament. It shows some artefacts that might be caused by DCT in texture layer; these artefacts can be removed after denoising.

We then set different reasonable threshold for cartoon layer and texture layer, respectively, and apply the iterative cleaning algorithm to the cartoon layer and texture layer to remove noise (as shown in Figure 7). Compared to Figure 6(b), Figure 7(b) almost contains no artefacts.

Next, the cartoon layer and texture layer are fused according to the intensity ratio (e.g., the information of the texture layer is expanded by a factor of 5). Small structures can then be retained and interference information is removed. Processing results after fusion are shown in Figure 8(c). To allow comparison with our method, we also use the iterative cleaning algorithm directly to process the 40th subimage; the results are shown in Figure 8(b).

As seen in Figure 8(b), most of the noises in the 40th subimage are removed by cleaning. However, getfilaments method sets only one noise threshold, and the values less than the threshold are cleared. This might clear weak information of the filament. As shown in the red box of Figure 8(b), the weak part of the filament is directly removed. Figure 8(c) is the denoised image after MCA decomposition. Structures of

the filament are more complete than those in panel b. Setting different threshold of noise variance for the two parts can avoid the removal of useful information, especially in the area marked by red box. Our method not only removes noise from the cartoon layer and texture layer but also strengthens details in the image synthesis process. Applying MCA to extract filament can retain much structural information of the filament.

3.2.2. Extraction Results. Finally, the layered subimages are added together to produce the filament. Figure 9 shows the extraction results obtained using the getfilaments algorithm. Figure 9(a) is the extracted filament done by getfilaments. Compared to the input filament in Figure 3(a), most of noises are cleaned and structures of the filament are also removed (marked in red boxes in Figures 9(b) and 9(c)). The extraction results obtained using our method are shown in Figure 10. Comparing with the getfilaments method, our method obtains the filament more complete, excludes noise, and retains more structural information, especially in the three areas marked by red boxes (Figures 10(b) and 10(c)).

3.3. Peak Signal-to-Noise Ratio. We compare the peak signal-to-noise ratio (PSNR) of images processed by the getfilaments

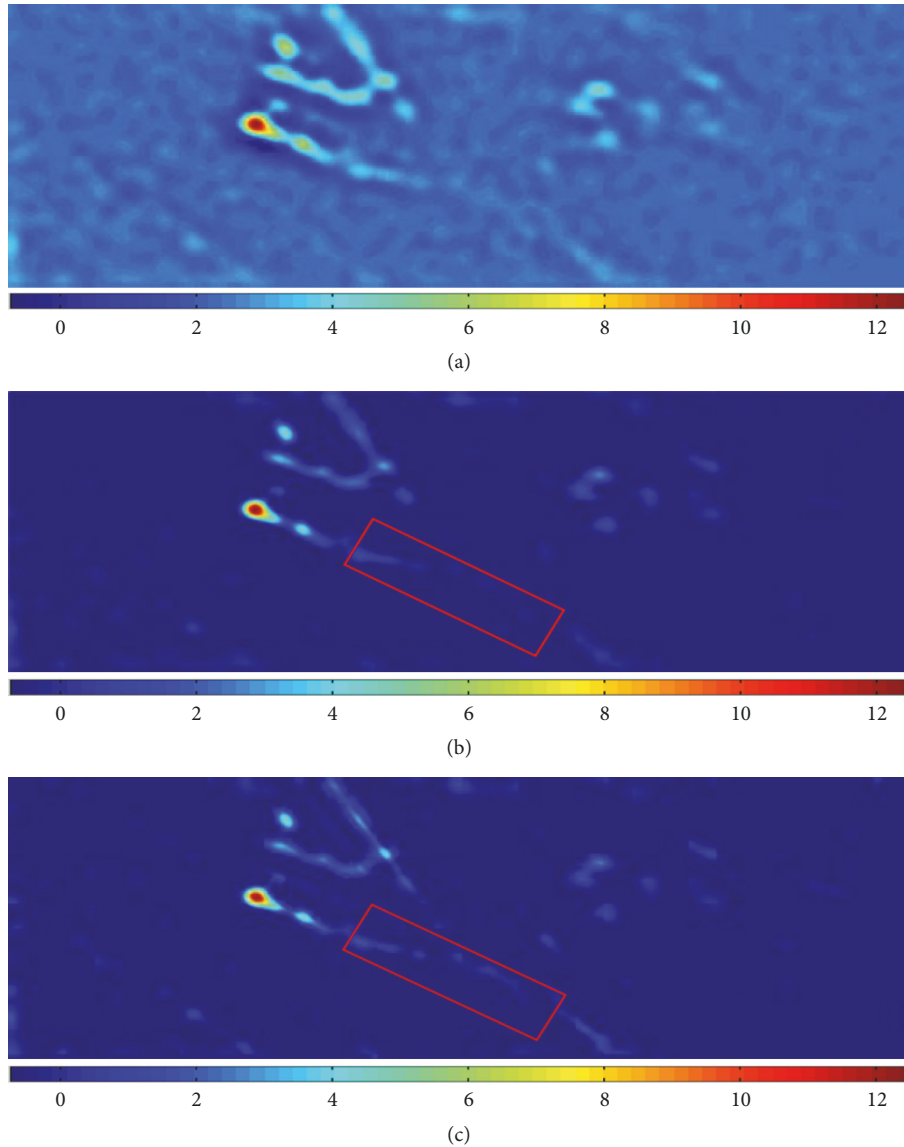


FIGURE 8: Denoising results of the 40th subimage. (a) The original 40th subimage. (b) The 40th subimage obtained using the iterative cleaning algorithm directly. (c) The denoised 40th subimage after MCA decomposition and fusion.

algorithm with that of our method. The PSNR is the objective criterion most widely used to evaluate image quality. An image has less noise when the PSNR is higher. The PSNR is defined as follows:

$$PSNR = 10 * \log_{10} \frac{(2^n - 1)^2}{MSE}. \quad (7)$$

where MSE is the mean square error (i.e., difference) between the original image and the image after noise is superimposed and n is 8 since pixels are represented using 8 bits per sample.

For different intensities of salt-and-pepper noise, we analyze the PSNRs of images processed by different methods. Table 1 shows that the PSNR of the images processed using

TABLE 1: PSNR comparison of images processed using different methods.

| Noise intensity | Images processed by different methods | | |
|-----------------|---------------------------------------|--------------|------------|
| | Original | Getfilaments | Our method |
| 0.1 | 18.2545 | 18.3443 | 18.5836 |
| 0.15 | 15.4864 | 16.5203 | 16.8379 |
| 0.2 | 14.9552 | 15.3325 | 15.7445 |
| 0.3 | 13.2314 | 13.7914 | 13.8200 |
| 0.5 | 11.2925 | 11.3277 | 11.8972 |

our improved method is always higher than that of getfilaments method, which means that the images processed by MCA have less noise.

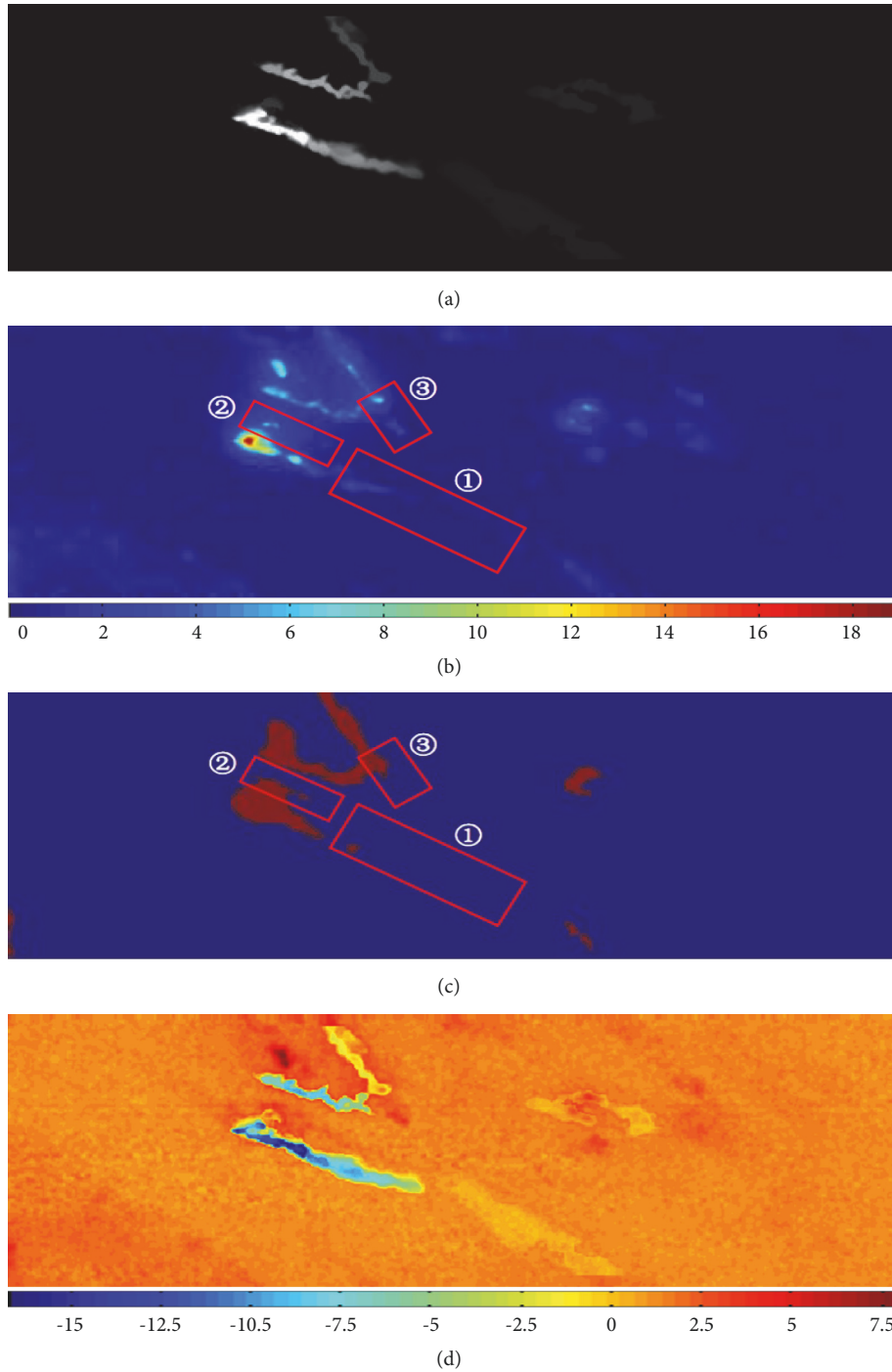


FIGURE 9: Extraction results using the `getfilaments` algorithm. (a) Extracted filament. Part of information is removed, including noises and structures. (b) Extracted filament with colors for contrast. Structures of the filament are cleaned in three marked places. (c) Contour extraction of the filament. (d) Residuals after the subtraction of the extracted filament from the original input image.

Data Availability

The data suffixed with fits used to support the findings of this study were supplied by National Astronomical Observatory of China under license and so cannot be made freely available.

Disclosure

F. Q. Duan present address is College of Information Science and Technology, Beijing Normal University, Beijing, China. The authors presented this work in 2016 Astronomical Data Analysis Systems and Software Conference.

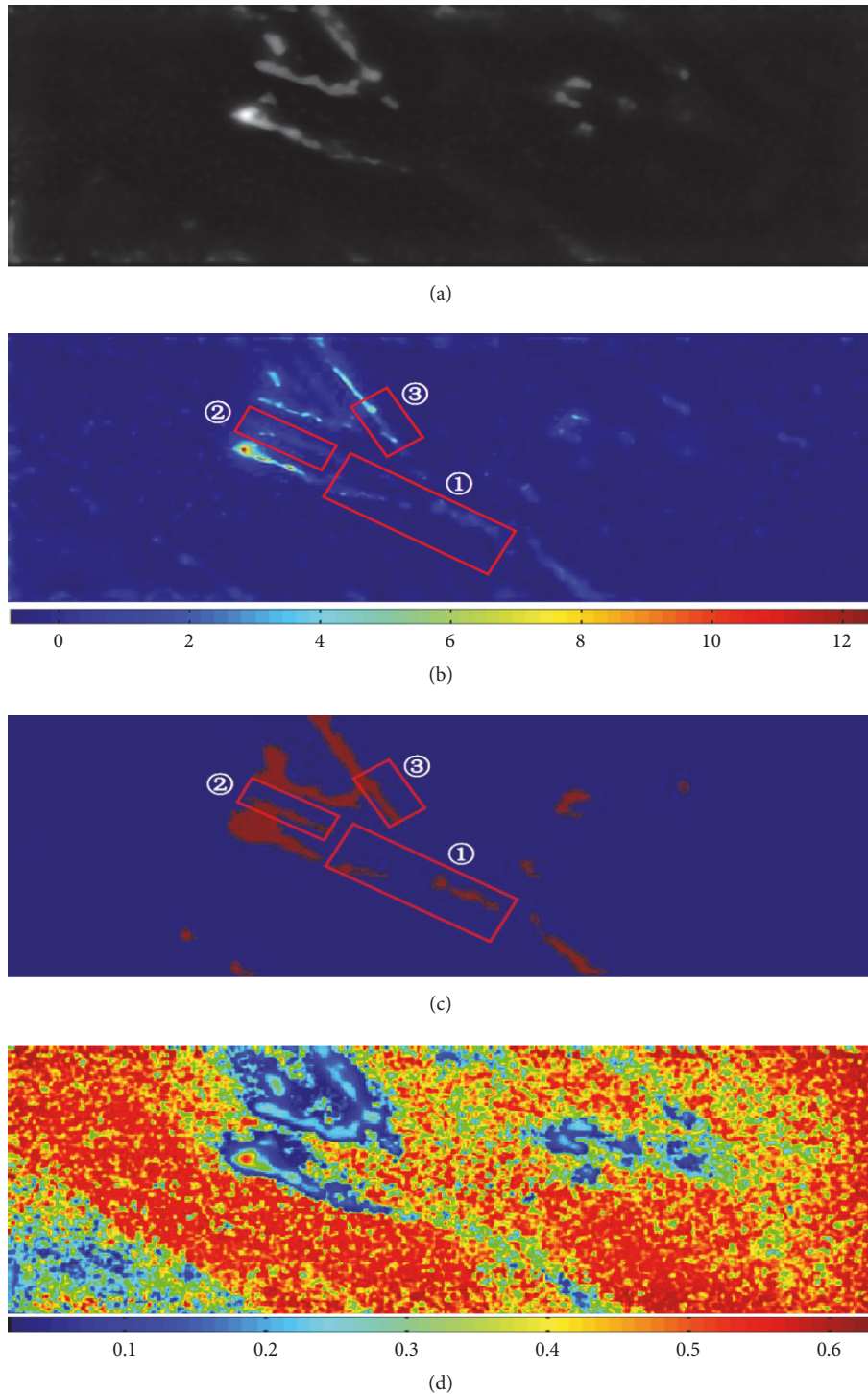


FIGURE 10: Extraction results obtained using our method. (a) Un-colored extracted filament. Noises are removed, and filament's structures are retained. (b) Colored extracted filament. Compared to the filament in Figure 9(b), filament's structures are more complete, especially in three marked places. (c) Contour extraction. (d) Residuals after the subtraction of the extracted filament from the original image.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

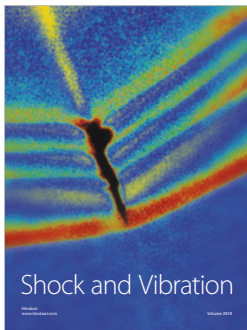
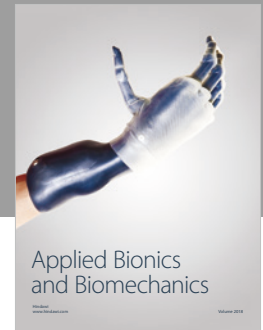
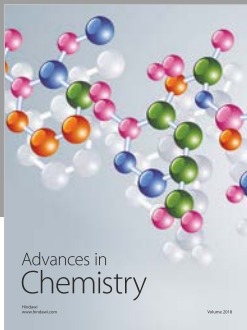
Acknowledgments

The study is supported by the National Natural Science Foundation of China (no. 41272359), Ministry of Land and

Resources for the Public Welfare Industry Research Projects (201511079-02), and Ph.D. Programs Foundation of Ministry of Education of China (20120003110032).

References

- [1] A. Men'shchikov, P. André, P. Didelon et al., "Filamentary structures and compact objects in the aquila and polaris clouds observed by herschel," *Astronomy and Astrophysics*, vol. 518, article L103, 7 pages, 2010.
- [2] G. C. Gómez and E. Vazquez-Semadeni, "Filaments in simulations of molecular cloud formation," *The Astrophysical Journal*, vol. 791, no. 2, article 6298, 2014.
- [3] P. Padoan and Å. Nordlund, "The stellar initial mass function from turbulent fragmentation," *The Astrophysical Journal*, vol. 576, no. 2, pp. 870–879, 2002.
- [4] P. André, A. Men'shchikov, S. Bontemps et al., "From filamentary clouds to prestellar cores to the stellar IMF: initial highlights from the herschel Gould belt survey," *Astronomy & Astrophysics*, vol. 518, article L102, 7 pages, 2010.
- [5] S. Molinari, B. Swinyard, J. Bally et al., "Clouds, filaments, and protostars: the herschel Hi-GAL milky way," *Astronomy and Astrophysics*, vol. 518, Article ID L100, 5 pages, 2010.
- [6] D. Kereš, N. Katz, D. H. Weinberg, and R. Davé, "How do galaxies get their gas?" *Monthly Notices of the Royal Astronomical Society*, vol. 363, no. 1, pp. 2–28, 2005.
- [7] D. Kereš, N. Katz, M. Fardal, R. Davé, and D. H. Weinberg, "Galaxies in a simulated Λ CDM universe - I. Cold mode and hot cores," *Monthly Notices of the Royal Astronomical Society*, vol. 395, no. 1, pp. 160–179, 2009.
- [8] S. F. Shandarin and Y. B. Zeldovich, "The large-scale structure of the universe: turbulence, intermittency, structures in a self-gravitating medium," *Reviews of Modern Physics*, vol. 61, no. 2, pp. 185–220, 1989.
- [9] C. F. McKee and L. L. Cowie, "The interaction between the blast wave of a supernova remnant and interstellar clouds," *The Astrophysical Journal*, vol. 195, pp. 715–725, 1975.
- [10] S. Casassus, G. van der Plas, M. Sebastian Perez et al., "Flows of gas through a protoplanetary gap," *Nature*, vol. 493, pp. 191–194, 2013.
- [11] N. Schneider, T. Csengeri, M. Hennemann et al., "Clusterformation in the Rosette molecular cloud at the junctions of filaments (Corrigendum)," *Astronomy and Astrophysics*, vol. 551, article C1, 1 page, 2013.
- [12] P. Hennebelle, R. Banerjee, E. Vázquez-Semadeni, R. S. Klessen, and E. Audit, "From the warm magnetized atomic medium to molecular clouds," *Astronomy & Astrophysics*, vol. 486, no. 3, pp. L43–L46, 2008.
- [13] A. V. Tugay, "Extragalactic filament detection with a layer smoothing method," *Physics*, vol. 1, 2014, <https://arxiv.org/abs/1410.2971>.
- [14] A. Men'shchikov, "A multi-scale filament extraction method: getfilaments," *Astronomy and Astrophysics*, vol. 560, article A63, 15 pages, 2013.
- [15] N. L. J. Cox, D. Arzoumanian, Ph. André et al., "Filamentary structure and magnetic field orientation in Musca," *Astronomy & Astrophysics*, vol. 590, Article A110, 8 pages, 2016.
- [16] J. Bobin, J.-L. Starck, J. Fadili, and Y. Moudden, "Sparsity and morphological diversity in blind source separation," *IEEE Transactions on Image Processing*, vol. 16, no. 11, pp. 2662–2674, 2007.
- [17] J. L. Starck, M. Elad, and D. L. Donoho, "Image decomposition via the combination of sparse representations and a variational approach," *IEEE Transactions on Image Processing*, vol. 14, no. 10, pp. 1570–1582, 2005.
- [18] S. Velasco-Forero and J. Angulo, "Classification of hyperspectral images by tensor modeling and additive morphological decomposition," *Pattern Recognition*, vol. 46, no. 2, pp. 566–577, 2013.
- [19] L. Yan, X. Xiaohua, and L. Jian-Huang, "Face hallucination based on morphological component analysis," *Signal Processing*, vol. 93, no. 2, pp. 445–458, 2013.
- [20] Z. Xue, J. Li, L. Cheng, and P. Du, "Spectral-spatial classification of hyperspectral data via morphological component analysis-based image separation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 1, pp. 70–84, 2015.
- [21] D. Szolgay and T. Sziranyi, "Adaptive image decomposition into cartoon and texture parts optimized by the orthogonality criterion," *IEEE Transactions on Image Processing*, pp. 3405–3415, 2012.
- [22] A. Buades, T. Le, J.-M. Morel, and L. Vese, "Fast Cartoon + Texture Image Filters," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 19, no. 8, 2010.
- [23] C. Yu, Q. Qiu, Y. Zhao, and X. Chen, "Satellite image classification using morphological component analysis of texture and cartoon layers," *IEEE Geoscience and Remote Sensing Letters*, vol. 10, no. 5, pp. 1109–1113, 2013.
- [24] X. Xiang, L. Jun, H. Xin, M. Dalla Mura, and P. Antonio, "Multiple morphological component analysis based decomposition for remote sensing image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 5, pp. 3083–3102, 2016.
- [25] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D Nonlinear Phenomena*, vol. 60, no. 1-4, pp. 259–268, 1992.
- [26] L. L. Jiang, H. Q. Yin, and X. C. Feng, "Adaptive variational models for image decomposition combining staircase reduction and texture extraction," *Journal of Systems Engineering and Electronics*, vol. 20, no. 2, pp. 254–259, 2009.
- [27] T. F. Chan, S. Esedoglu, and F. E. Park, "Image decomposition combining staircase reduction and texture extraction," *Journal of Visual Communication and Image Representation*, vol. 18, no. 6, pp. 464–486, 2007.
- [28] E. J. Cand and D. L. Donoho, "Curvelets, multiresolution representation, and scaling laws," 1, 2000.
- [29] J.-L. Starck, E. J. Candes, and D. L. Donoho, "The curvelet transform for image denoising," *IEEE Transactions on Image Processing*, vol. 11, no. 6, pp. 670–684, 2002.
- [30] L. Vincent and P. Soille, "Morphological segmentation of binary patterns," *Pattern Recognition Letters*, 1991.
- [31] N. Laurent and S. Michel, "Geodesic saliency of watershed contours and hierarchical segmentation," vol 18, 1163, 1996.
- [32] F. Meyer, "Watersheds on weighted graphs," *Pattern Recognition Letters*, vol. 47, pp. 72–79, 2014.
- [33] F. Malmberg and C. L. L. Hendriks, "An efficient algorithm for exact evaluation of stochastic watersheds," *Pattern Recognition Letters*, vol. 47, pp. 80–84, 2014.
- [34] A. Men'shchikov, P. André, P. Didelon et al., "A multi-scale, multi-wavelength source extraction method: getsources," *Astronomy and Astrophysics*, vol. 542, article A81, 31 pages, 2012.



Hindawi

Submit your manuscripts at
www.hindawi.com

