

# Text Line Segmentation in Handwritten Documents Using Mumford-Shah Model

Xiaojun Du

Department of Computer  
Science and Software  
Engineering, Concordia  
University  
xiaoj\_du@cse.concordia.ca

Wumo Pan

Department of Computer  
Science and Software  
Engineering, Concordia  
University  
wumo\_pan@cse.concordia.ca

Tien. D. Bui

Department of Computer  
Science and Software  
Engineering, Concordia  
University  
bui@cse.concordia.ca

## Abstract

*Text line segmentation in handwritten documents is an important step in document processing. We present a new text line segmentation method based on the Mumford-Shah model. The algorithm is script independent. In addition, we use morphing to remove overlaps between neighboring text lines and connect broken ones. Experimental results show the validity of our method.*

**Keywords:** Text line detection, image segmentation, Mumford-Shah model.

## 1. Introduction

Text line segmentation in handwritten documents is an important step in document processing. Although the current techniques of text line detection are quite successful in machine printed documents, processing of handwritten documents has remained a difficult problem. Most current text line segmentation approaches are based on two assumptions: 1) the gap between two neighboring lines is significant; and 2) the lines are reasonably straight. However, these assumptions are not always valid for handwritten documents. Most approaches are sensitive to the topological changes in handwritten documents. The authors of [4] proposed to use the level-set method to segment the text lines. In [4], the binary text image is first converted to a gray scale image, and the text line segmentation is achieved by using a general image segmentation approach: the level set method [2]. Because the method in [4] does not depend on the above assumptions, it obtains very good results. However, the approach in [4] also faces some difficulties: Because the boundary based level set method is used, the segmentation result depends on the number of boundary evolution steps. In addition, the failure cases shown in [4] indicate that the

approach is sensitive to the overlaps between neighboring text lines. In this paper, we present a new algorithm based on the Mumford-Shah (MS) model [1]. Because the text image only consists of two uniform regions: text region and background region, the piecewise constant approximation [3] of the MS model is very appropriate for the segmentation of text lines. Different from the approach in [4], the MS model is a region based approach. Segmentation is achieved by minimizing the MS energy functional. Therefore, the segmentation result does not depend on the number of evolution steps. We also use morphing to remove overlaps between neighboring text lines and connect the broken lines. Compared with the text line localization approach in [4], our approach takes advantages of the level set representation to morph the segmentation regions. Our method is straightforward and easy to implement. The approach also can overcome the problem of overlaps between neighboring text lines. In sections 2 and 3, the new algorithm and experimental results are described. Discussion of the algorithm is presented in section 4. Finally the conclusion is reached in section 5.

## 2. The method

### 2.1. Mumford-Shah model and the level set approach for segmentation

The segmentation of an image can be defined as follows: For an observed image  $u_0$ , we want to find an optimal piecewise smooth approximation  $u$  of  $u_0$  for each specific region. The regions are denoted by  $\Omega_i, i=1,2,\dots,n$ . The function  $u$  varies smoothly within each  $\Omega_i$  and rapidly or discontinuously across the boundaries of  $\Omega_i$ . The boundaries of all  $\Omega_i$  are denoted as  $C$ . The whole image can be expressed as:

$$\Omega = \cup \Omega_i \cup C$$

The process of finding the boundaries of  $\Omega_i$  is called segmentation. Mumford and Shah proposed that the segmentation of an image can be obtained through the minimization of an energy functional [1]:

$$E(u, C) = \int_{\Omega} |u - u_0|^2 dx dy + \mu \int_{\Omega \setminus C} |\nabla u|^2 dx dy + \nu \cdot |C| \quad (1)$$

In equation (1),  $u_0$  is the original image;  $u$  is the smooth approximation of the image;  $C$  is the union of the segmentation curves;  $|C|$  represents the total length of the curves;  $\Omega$  is the image domain;  $\Omega \setminus C$  represents the image domain excluding the segmentation curves. If the MS energy functional is minimized, the image will be segmented into regions so that: (1)  $u$  is a good approximation of  $u_0$ ; (2)  $u$  is smooth in each region; and (3) the boundary of each region is as short as possible. The parameters  $\mu$  and  $\nu$  are used to balance the effects of different terms.

Minimization of the MS energy functional is not a trivial task. There are some alternative solutions to this problem, such as the elliptic approximation to the weak formulation of the MS function [5], the active contours without edges [3], the curve evolution based approach [6], and the hierarchical segmentation [8]. Many approaches are based on the level set method [2] [7] because it provides an efficient way for the numerical solution.

Chan and Vese [3] proposed a piecewise constant approximation for the Mumford-Shah model. If the image intensities inside different regions are uniform, the image intensities inside different regions can be approximated by constants. In this case, the MS energy functional can be simplified to equation (2).

$$E(c_k, C) = \sum_k \int_{\Omega_k} (c_k - u_0)^2 dx dy + \nu |C| \quad (2)$$

where  $\Omega_k$  represents the area inside each region. The gradient term in the MS energy functional disappears in equation (2) because of the piecewise constant approximation.

Using the level set function, the MS energy functional of the two phase segmentation can be written as:

$$E(c_1, c_2, \phi) = \int (c_1 - u_0)^2 H(\phi) dx dy + \int (c_2 - u_0)^2 (1 - H(\phi)) dx dy + \nu \int \delta(\phi) |\nabla \phi| dx dy \quad (3)$$

where  $H(x)$  is the Heaviside function.

$$H(x) = \begin{cases} 1 & \text{if } x > 0 \\ 0 & \text{if } x < 0 \end{cases} \quad (4)$$

Minimizing the energy functional with respect of  $c_1$ ,  $c_2$ , and  $\phi$ , we obtain following equations:

$$c_1(\phi) = \frac{\int u_0 H(\phi) dx dy}{\int H(\phi) dx dy} \quad (5)$$

$$c_2(\phi) = \frac{\int u_0 (1 - H(\phi)) dx dy}{\int (1 - H(\phi)) dx dy} \quad (6)$$

$$\frac{\partial \phi}{\partial t} = \delta(\phi) [\nu \nabla \cdot \left( \frac{\nabla \phi}{|\nabla \phi|} \right) - (u_0 - c_1)^2 + (u_0 - c_2)^2] \quad (7)$$

After solving these equations, we can obtain the information about  $c_1$ ,  $c_2$ , and  $C$ , and image  $u_0$  is segmented into two regions  $\{u = c_1\}$  and  $\{u = c_2\}$ .

In piecewise constant approximation, only one PDE needs to be solved, therefore the approach is relative efficient. Because the text image can be represented by two constants: white background and black text, the piecewise constant approximation is an ideal approach for text line segmentation.

## 2.2. Text line segmentation algorithm

The level set approach is first used to solve the text line segmentation problem by the authors in [4]. They proposed a three steps algorithm for text line segmentation: (1) blurring the text image to enhance text lines; (2) segmentation of text lines by the level set method; (3) text line localization. We adopt the first step of their algorithm, but we use different approaches in the other two steps. Our algorithm consists of the following steps: (1) blurring the text image to enhance text lines; (2) segmentation of text lines by the Mumford-Shah model; (3) text line detection by morphing approach.

### 1) Blurring text lines

This step is the same as the first step in [4]. To enhance the text lines, the text image is blurred by a Gaussian filter. The filter window is a rectangle whose width is larger than its height. Assuming the text lines are horizontal, a wide window blurs words and fills the horizontal gaps between the words. As the result, isolated words are connected into horizontal lines. On the other hand, a too narrow window may not be able to close the gaps between the words on the same line. In the blurred image, we cannot distinguish detailed structure of characters. The only information is the blurred lines across the image. Different from the original binary text image, the blurred image is a gray scale image. The intensity of each pixel represents its possibility to be on the text lines: the darker the pixel, the more likely the pixel is on the text lines. We can segment the text lines according to the image intensity.

### 2) Text line segmentation

We use piecewise constant approximation of the Mumford-Shah model to segment text lines. Because the text image can be represented by two phases: text line region and background region, we use one level set function to represent the segmentation regions. We use two constants to represent the two regions. In the segmentation process, an initial curve is set to segment the image into two regions: inside and outside regions. Then the segmentation curve evolves to the boundaries of the text lines according to the MS energy functional. In each evolution step, the constant of each region is calculated by the average of the intensity of the region; and the difference between the constant and the image intensity of the region at each pixel is calculated. Finally, the image is segmented into two regions, and the uniformity of each region is maximized. At the same time, the minimized segmentation boundary length can avoid the effect of noise. Different from the level set approach in [4], our approach does not depend on the number of evolution steps, and the segmentation can automatically stop after the MS energy is minimized.

### 3) Text line detection

After the image segmentation step, the image is segmented into two regions: text line region and background region. Because of the overlap among different text lines, neighboring text lines may be connected. Another problem is the broken text lines due to large horizontal gaps between different words. We use morphing approach to overcome these problems.

First, we shrink (erode) the text line region along the horizontal direction. If the overlap between two neighboring text lines is not large, the connection of the two text lines is thin along the horizontal direction. In this case, the shrinkage of the text line region will remove the overlaps.

Second, we prolong (dilate) the text line region horizontally. The prolongation length is longer than the shrinkage length. If the gaps between neighboring broken text lines is small, the horizontally broken text line will be connected and merged into a long text line across the image.

Third, we shrink the text line regions back to the original horizontal length.

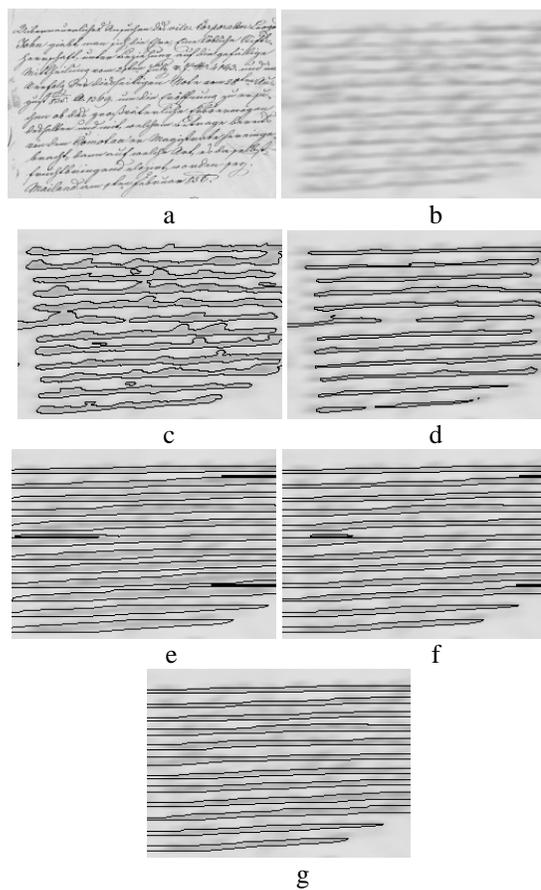
Finally, we check the length and height of every segmentation region. If the length or height is too small, this segmentation region is considered as noise and is removed.

The level set method is a good approach to obtain the segmentation regions. It is also easy to implement the morphing approach based on the level set method.

## 3. Experiments

We test the algorithm on different text images.

In Figure 1, the original image first is blurred; then it is segmented. In the segmentation stage, we need not fix the number of iteration steps, and we can obtain good segmentation results automatically. After this step, there are many overlaps between neighboring text lines and broken text lines. After morphing, the overlaps are removed, and the broken text lines are connected. The final step removes the noise.



**Figure 1.** The process of text line segmentation algorithm. a: original image; b: blurred image; c: segmented image; d: shrunk text lines; e: prolonged text lines; f: shrunk back to original text line length; g: final segmentation result after removing noise.

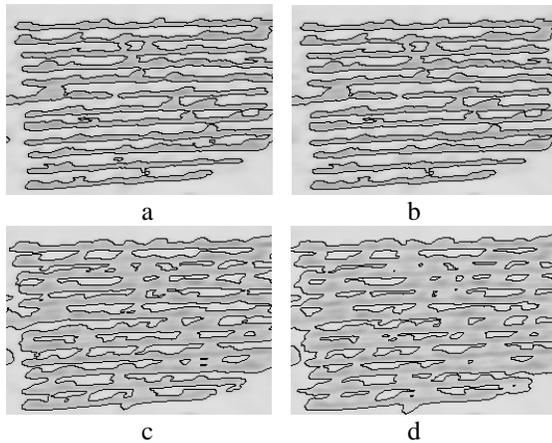
The algorithm in [4] consists of three main steps:

- 1) Pre-processing (blurring).
- 2) Application of the level set method to segment the blurred text lines.
- 3) Post-processing (merging broken text lines resulting from step 2).

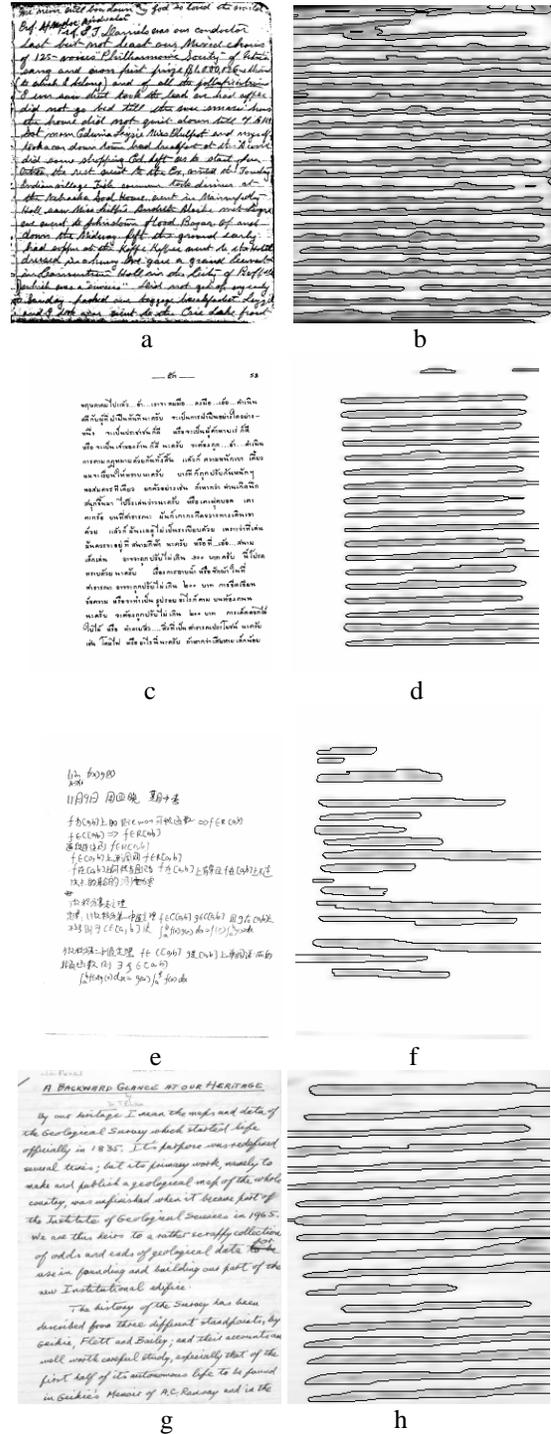
Our algorithm is different from [4] in steps 2 and 3. In step 2, we use the energy minimization model of MS to segment the blurred text lines. In step 3, we use morphing to merge the broken text lines and remove overlaps. This approach is different from the post-processing step in [4]. It is not possible to compare our algorithm with [4] directly because we do not have the same datasets, nor do we have the ground truth used in [4]. In the following, we show the different results based only on step 2 of our algorithm as compared to step 2 of [4]. These results show that the MS model is much better than the case of the level set method for segmentation of the blurred text lines. We must emphasize that we did not compare our method with the full algorithm in [4].

In Figure 2, we use the level set method to segment the text image in Figure 1a. Figure 2a is the initial segmentation curve; Figs. 2b, 2c, and 2d are the segmentation results with different iteration steps. As shown in Figure 2, the segmentation results with different iteration steps are quite different. In the algorithm of [4], we need to fix the number of iteration steps to get the best result. The final results will depend strongly on the number of iteration steps of the level set method.

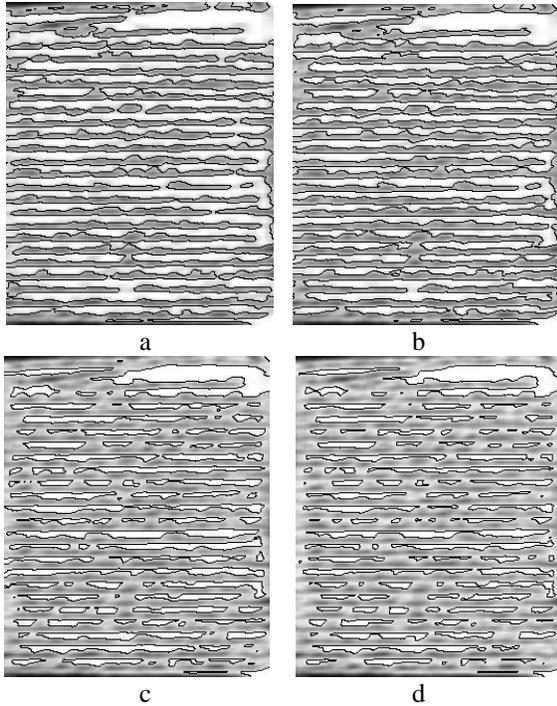
Figure 3 shows the experimental results on different language text images. Especially, Figure 3e is a mix of Chinese language and mathematical equations. The results indicate that the algorithm is script independent.



**Figure 2.** The segmentation results of Figure 1a using the level set method. a: initial segmentation curve; b: segmentation result of 8 iteration steps; c: segmentation result of 10 iteration steps; d: segmentation result of 11 iteration steps.

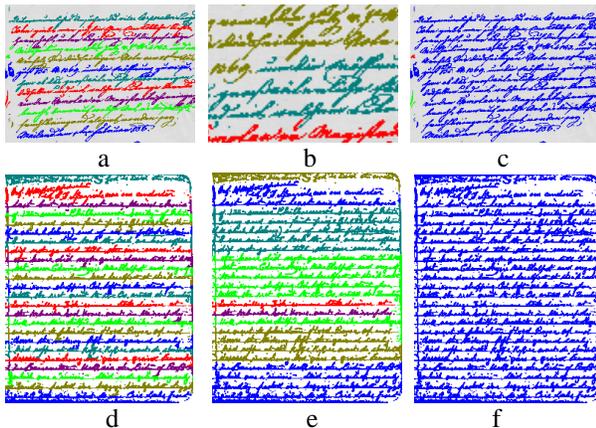


**Figure 3.** Experiments on different languages. left column: text images; right column: segmentation results.



**Figure 4.** The segmentation results of Figure 3a using the level set method. a: initial segmentation curve; b: segmentation result of 10 iteration steps; c: segmentation result of 20 iteration steps; d: segmentation result of 30 iteration steps.

Figure 4 is the segmentation results of Fig. 3a using the level set method. Fig. 4a is the initial segmentation curve; Figs. 4b, 4c, and 4d are the segmentation results with different iteration steps. In Figs. 4c and 4d, more iteration steps create more overlaps among text lines, and many text lines can not be separated. This indicates that the choice of the iteration steps is critical to the algorithm in [4].



**Figure 5.** Text line detection results. a: result of figure 1a; b: result of Figure 2b; c: result of Figure 2c and 2d;

d: result of Figure 3b; e: result of Figure 4b; f: result of Figure 4c and 4d.

Figure 5 shows the text line detection results. In this figure, different colors represent different text lines. Figures 5a and 5d are the text line detection results using the MS model. Our algorithm can detect all text lines. Figures 5b, 5c, 5e, and 5f are the text line detection results using the level set method. Figures 5b and 5e are the results of Figures 2b and 4b. Because of the overlaps among text lines, some text lines can not be separated. Figures 5c and 5f are the results of Figures 2c, 2d, 4c and 4d. Because more iteration steps cause more overlaps among text lines, all text lines can not be separated.

## 4. Discussion

There are some factors that need to be considered in our algorithm.

### 1) The size of blurring window

The width of the blurring window depends on the horizontal gaps between words in one text line. The height of the blurring window depends on the vertical gaps between text lines. Too small width will produce broken text lines; too large height will add overlaps between neighboring text lines. However, these two factors are not critical because the overlaps and broken text lines can be corrected by morphing. So these two factors can be roughly defined.

### 2) The initial condition for segmentation

Because the MS energy functional is not convex, minimization result depends on the initial condition. However, because the content of text images is simple (only text and background), this problem is not an issue. We use different initial conditions and obtain the same results.

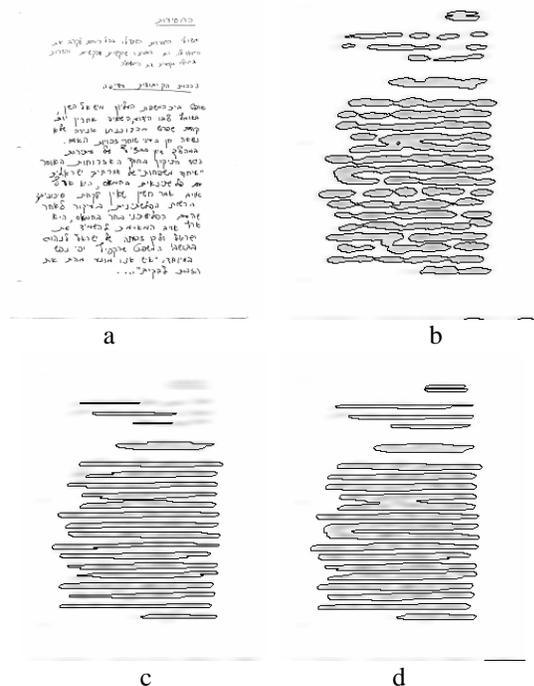
### 3) The shrinkage length of text regions

The shrinkage length depends on the thinness of the overlaps between neighboring text lines. In the case of broken text lines, it also depends on the minimum length of the broken text segments. If the shrinkage length is too small, the overlaps will not be removed. On the other hand, too large shrinkage length will remove some short broken text segments (as shown in Fig 6). In the case that the overlaps are big and the broken segments are short, it is tricky to select the shrinkage length.

### 4) The prolongation length of text regions

The prolongation length depends on the horizontal gaps between neighboring words in one text line. Too small prolongation length will produce many broken text lines.

Consider all these factors, we select some parameters for all experiments (burring windows width: 20 pixels, height: 5 pixel; shrinkage length: 50 pixels; prolongation length: 200).



**Figure 6.** The segmentation results with different parameters. a: Text image; b: segmented image before morphing; c: final segmentation result with shrinkage length of 50; d: final segmentation result with shrinkage length of 20.

Figure 6 is a failure example of the algorithm. Figure 6a is the text image. Figure 6b is the segmented image before morphing. In this figure, there are some broken text lines and overlaps between neighboring text lines. Figure 6c is the final segmentation result with shrinkage length of 50. In this case, some texts are missed in the upper right corner. If we use smaller shrinkage length, say 20 pixels, these texts can be kept, but some overlaps between neighboring text lines cannot be removed as shown in Figure 6d.

## 5. Conclusion

In this paper, we present a text line segmentation approach based on the Mumford-Shah model. The algorithm is script independent. We use piecewise constant approximation of the MS model to segment handwritten text images. The segmentation result does not depend on the number of evolution steps. In addition, we use the

morphing to remove overlaps between neighboring text lines and connect broken text lines. The experimental results indicate the validation of the approach.

## Acknowledgement

The authors would like to thank David Doermann and Stefan Jaeger of the University of Maryland for sending us the database of handwriting documents.

## References

- [1] D. Mumford, and J. Shah, "Optimal approximation by piecewise smooth functionals and associated variational problems," *Comm. Pure Appl. Math.*, ol. 42, pp.577-685, 1989.
- [2] S. Osher and J. Sethian, "Fronts propagating with curvature-dependent speed: Algorithm based on the Hamilton-Jacobi formulation," *Journal of Computational Physics*, 79 pp12-49, 1988.
- [3] T. F. Chan and L. A. Vese, "Active contours without edges," *IEEE Trans on Image Processing*, vol. 10, no. 2, pp.266-277, 2001.
- [4] Yi Li, Yefeng Zheng, David Doermann, and Stefan Jaeger, "A New Algorithm for Detecting Text Line in Handwritten Documents," *Proceedings of the Tenth International Workshop on Frontiers in Handwriting Recognition*, Oct. 2006, La Baule.
- [5] G. Aubert and P. Kornprobst, "*Mathematical Problem in Image Processing: Partial Differential Equations and the Calculus of Variations*," New York: Springer, vol. 147, 2002.
- [6] A. Tsai, A. Yezzi, and Alan S. Willsky, "Curve evolution implementation of the Mumford-Shah functional for image segmentation, denoising, interpolation, and magnification," *IEEE Trans. on Image Processing*, vol. 10, no. 8, pp.1169-1186, 2001.
- [7] J. A. Sethian, *Level Set methods: evolving interfaces in geometry, fluid mechanics*, Cambridge Univ. Press, 1996
- [8] S. Gao and T. D. Bui, "Image segmentation and selective smoothing by using Mumford-Shah model," *IEEE Transactions on Image Processing*, vol. 14, no. 10, pp.1537-1549, 2005.