Cloud Detection of RGB Color Aerial Photographs by Progressive Refinement Scheme

Qing Zhang, Chunxia Xiao

Abstract—In this paper, we propose an automatic and effective cloud detection algorithm for color aerial photographs. Based on the properties derived from observations and statistical results on a large number of color aerial photographs with cloud layers, we present a novel progressive refinement scheme for detecting clouds in the color aerial photographs. We first construct a significance map which highlights the difference between cloud regions and non-cloud regions. Based on the significance map and the proposed optimal threshold setting, we obtain a coarse cloud detection result which classifies the input aerial photograph into the candidate cloud regions and non-cloud regions. In order to accurately detect the cloud regions from the candidate cloud regions, we then construct a robust detail map derived from a multiscale bilateral decomposition to guide us remove non-cloud regions from the candidate cloud regions. Finally, we further perform a guided feathering to achieve our final cloud detection result, which detects semitransparent cloud pixels around the boundaries of cloud regions. The proposed method is evaluated in terms of both visual and quantitative comparisons, and the evaluation results show that our proposed method works well for cloud detection of color aerial photographs.

Index Terms—Progressive refinement scheme, cloud detection, optimal thresholding, image segmentation, color aerial photograph, significance map, detail map, guided feathering.

I. INTRODUCTION

WITH the rapid development of aerial photograph acquisition technology, obtaining high resolution color aerial photographs is an easy task now. Huge amount of aerial photographs are created every day, and these aerial photographs have been widely used in agriculture engineering, environmental protection, resource exploration, geographical survey and military reconnaissance. As cloud covers more than 50% surface of the earth, many aerial photographs will contain cloud regions. As a result, clouds will lead to inaccurate analysis and interpretation for color aerial photographs. For example, it will be easier to identify objects from color aerial photographs if cloud regions have been specified. Hence, cloud detection of color aerial photographs is an important preprocessing for many follow-up manipulation, such as object recognition, image retrieval [1], [2] and image classification [3], [4]. Besides, detecting cloud regions automatically and effectively will benefit for many automation applications about color aerial photographs, such as disaster forecast, pollution monitoring and oil exploration etc. Thus, cloud detection for color aerial photographs is a meaningful work.

The authors are with the School of Computer, Wuhan University, Wuhan 430072, China. E-mail: zhangqing.whu.cs@gmail.com, cxxiao@whu.edu.cn.
Corresponding to Chunxia Xiao: cxxiao@whu.edu.cn.

Cloud detection is a very challenging work, since it is essentially an image segmentation problem. To detect cloud regions in color aerial photographs, we need to first recognize the cloud layers, and then segment the aerial photograph into two classes (cloud regions and non-cloud regions). However, for complex aerial photographs that have both bright non-cloud regions and semitransparent cloud pixels, current techniques usually fail to provide a satisfactory cloud detection result. Even resorting to the state-of-art image segmentation methods, we still cannot get a good cloud detection result without user interaction. Besides the detection accuracy, high efficiency and automaticity (no user interactions) should be taken into consideration as well to process high resolution photographs, while the later two requirements are also difficult problems.

A number of cloud detection methods [5]–[7] have been proposed. However, most of them are designed for moderate spatial resolution sensors such as Advanced Very High Resolution Radiometer (AVHRR) and Moderate Resolution Imaging Spectroradiometer (MODIS). These sensors are usually equipped with more than one thermal band, or with water vapor absorption bands. To the best of our knowledge, there is little work focusing on cloud detection of color aerial photographs with just RGB bands information. The other relevant work may be image segmentation [8]–[13], shadow detection [14]–[16], and image dehazing [17]–[19]. Different from multispectral satellite images, the RGB color aerial photograph is optical data which is consisted of only three RGB channels. Thus, cloud detection for color aerial photographs is more difficult than multispectral satellite images since we have no auxiliary channels information. Automatic image segmentation methods [9]–[11] can be used for cloud detection, however, they usually fail to handle complex aerial photographs. Interactive image segmentation methods, such as graph-cut [8], Grabcut [12], and alpha matting [13], [20], [21] can be used to detect clouds, but these methods need large amount of user interactions which prevent them from being put into practical applications in cloud detection for high resolution photographs.

In this paper, we proposed a novel progressive refinement scheme for cloud detection of color aerial photographs. Our approach is based on some properties derived from observations and statistical results on a large number of color aerial photographs with cloud layers. We find that cloud pixels in color aerial photographs usually have higher intensities and lower hues. Based on this observation, we first construct a significance map for the input aerial photograph, which successfully highlights the difference between cloud pixels and non-cloud pixels. Then, we present an optimal threshold setting scheme to obtain a coarse cloud detection result.

In order to handle complex cloudy aerial photograph, such as the photographs with both bright non-cloud regions and semitransparent cloud pixels, we incorporate a detail map into the cloud detection to obtain a finer cloud detection result. Finally, we further perform a guided feathering to achieve the final cloud detection result, which detects semitransparent cloud pixels around the boundaries of cloud regions.

In summary, our work has the following three main contributions:

- Based on the constructed significance map, we propose an optimal threshold setting scheme to obtain a good coarse cloud detection.
- We develop a detail-aware cloud detection method to process complex cloudy aerial photographs, which can refine the coarse cloud detection results.
- We introduce guided image filtering to detect the semitransparent cloud pixels around the boundaries of the cloud regions.

Our method has the following advantages that would benefit practical applications:

- **Automation:** Our method can automatically detect clouds in color aerial photograph without user interactions.
- **Accuracy:** Our method has high cloud detection accuracy even processing complex cloudy aerial photographs.
- **Efficiency:** Our method can provide us fast feedback ($<$2s) for input color aerial photographs with moderate size (typically for aerial photograph with size of 1024$\times$1024).

The remainder of this paper is organized as follows. In section II, we introduce the related work. In section III, we present the technique details for the progressive refinement scheme used in the cloud detection system. Section IV gives the experimental results and comparisons to demonstrate the performance of our method, and the limitations of our method is also given. Section V concludes the paper and gives the future work.

II. RELATED WORK

In this section, we review the most related work to our work, namely, cloud detection, image segmentation, shadow detection and image dehazing. Our cloud detection method for color aerial photographs is partially inspired by these works.

Cloud detection. In the last few years many researchers have explored automatic cloud detection methods for multi-spectral satellite images. Most of the current cloud detection methods are highly dependent on the available spectral bands besides RGB. Some of these methods work on pixel by pixel basis [22], some use neighborhood information, such as local standard deviation [5]. The Moderate Resolution Imaging Spectroradiometer (MODIS) onboard the NASA *Terra* and *Aqua* satellites measures radiances at 36 wavelengths, including infrared and solar bands, with a spatial resolution between 250m to 1km. Based on the MODIS data, many cloud detection methods [6], [7] have been proposed to improve the detection performance. Although there are many effective cloud detection methods for multi-spectral data have been proposed, we can not directly apply these methods to detect clouds for color aerial photographs. To the best of our knowledge, there is little work about cloud detection for color aerial photographs in literature. Cloud detection of color aerial photographs is more difficult because we have no auxiliary bands information except common RGB. Le et al. [23] proposed an automatic detection algorithm for cloud/shadow on high resolution optical images by using a Markov Random Field (MRF) framework. However, this method may fail to distinguish cloud regions from bright non-cloud regions, and it is also time-consuming for energy optimization.

Image segmentation. Cloud detection of color aerial photographs is essentially an image segmentation problem, we can apply some automatic image segmentation methods [9]–[11] to detect clouds for color aerial photograph. However, automatic image segmentation methods usually cannot achieve good results for complex images. To increase the detection accuracy, we can resort to some interactive image segmentation methods. Many popular interactive image segmentation methods [8], [12] have been proposed. Boykov et al. [8] proposed a method via graph cuts to achieve segmentation results by combining colour information and edge information. Grabcut [12] system employed graph cuts optimization to achieve more coherent and higher quality foreground segmentation. However, these methods require lots of user interactions to process high resolution aerial photographs. Thus, these methods are inapplicable in practical applications. Many soft segmentation methods based on image matting [13], [20], [21] have been designed to deal with boundaries of fuzzy foreground objects. Although we can get a good cloud detection result based on the these matting methods, we should provide some complex user specified scribbles beforehand, furthermore, these methods usually requires long processing time.

Shadow detection. Similar to the automatic cloud detection of color aerial photographs, automatic shadow detection is also a challenging problem, and there are lots of similarities between these two topics. Researchers have designed many effective automatic shadow detection methods [14]–[16], [24]. Huang et al. [15] noticed that shadow pixels usually have larger hues, lower blue channel values, and relatively small difference between the values of green and blue channels. Based on these facts, they computed three thresholds over the histogram of the input image to separate the shadow regions from non-shadow regions. Inspired by observation of Huang et al. [15], Tsai [16] proposed a shadow detection method for color aerial photographs which used a ratio map of the hue to the intensity in conjunction with a thresholding setting. Instead of using the ratio map proposed in Tsai's method, Chung et al. [14] proposed a modified ratio map to further stretch the disparity between shadow regions and non-shadow regions, and a successive thresholding scheme is then applied to detect shadow regions from color aerial photographs.

Image Dehazing. As cloud is sometimes similar to the haze in appearance, cloud detection is relevant to image dehazing in some aspects. Many works have been done on image dehazing in recent years. Tan [19] found that clear images had higher contrast compared with foggy images, thus he maximized the local contrast of the restored image for enhancing image visibility. Based on the assumption that the propagation of

light and the shading parts of the target surface were locally uncorrelated, Fattal [17] first estimated the scene radiance and then derived the transmission image. He et al. [18] proposed a single image haze removal approach based on Dark Channel Prior (DCP) and produced impressive results. More recently, inspired by the Dark Channel Prior [18], Xiao et al. [25] proposed a fast image dehazing method by using joint bilateral filtering.

III. CLOUD DETECTION ALGORITHM

The HSI color model [26], which follows the human visual perception closely, separates the color components in terms of intensity, hue and saturation. For an input RGB color aerial photograph, we can transform it into HSI color model as follows:

$$\begin{pmatrix} I \\ V_1 \\ V_2 \end{pmatrix} = \begin{pmatrix} 1/3 & 1/3 & 1/3 \\ -\sqrt{6}/6 & -\sqrt{6}/6 & \sqrt{6}/3 \\ 1/\sqrt{6} & -2/\sqrt{6} & 0 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix} \quad (1)$$

$$S = \sqrt{V_1^2 + V_2^2} \quad (2)$$

$$H = \begin{cases} (\tan^{-1}(\frac{V_2}{V_1}) + \pi) \times \frac{255}{2\pi}, & if\ V_1 \neq 1 \\ H\ is\ undefined, & otherwise \end{cases} \quad (3)$$

In the HSI color model, H and I corresponds to the intensity-equivalent and hue-equivalent components, respectively.

It seems that it is an easy task for human beings to identify clouds in color aerial photographs. However, identifying clouds in color aerial photographs by computer is a difficult problem, especially to detect the cloud regions automatically and effectively. Fortunately, after a large amount of observations and statistical experiment, we have observed that cloud regions in the color aerial photographs usually share the following common properties:

- *Property 1*: Cloud regions generally have higher intensity since the reflectivity of cloud regions is usually larger than that of the non-cloud regions.
- *Property 2*: Most cloud regions often have lower saturations and hues.
- *Property 3*: In color aerial photographs, the ground covered by cloud veil usually has little details as the ground object features are attenuated by cloud veil.
- *Property 4*: Cloud regions in color aerial photographs always appear in terms of clustering, not in sparkle cloud pixels.
- *Property 5*: Semitransparent cloud pixels are often present around the boundaries of cloud regions.

Based on above observations, we propose our progressive refinement scheme for cloud detection. In Fig.1, we give the block diagram of the proposed cloud detection system. Our system has the following main steps. Firstly, we construct a significance map for the input color aerial photograph. Secondly, based on the significance map, we develop an optimal threshold setting to obtain a coarse cloud detection result. Then, we incorporate a detail map into the cloud detection to remove redundant non-cloud regions in the coarse result. Finally, a guided feathering technology is used to refine the cloud detection result.
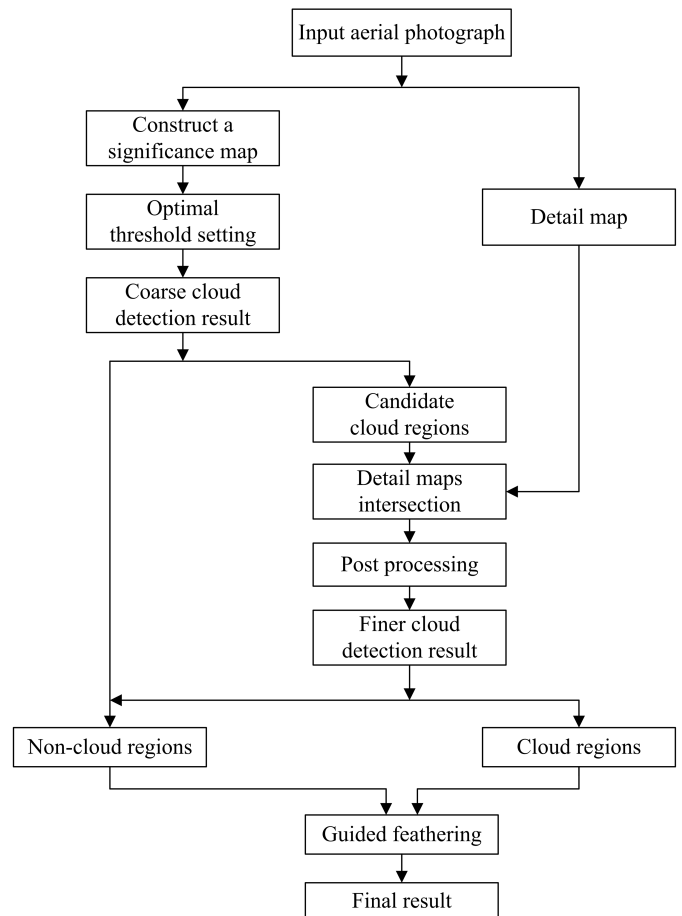

Fig. 1. Block diagram of the proposed cloud detection system.

A. Significance map

We have learnt from above properties that cloud regions in color aerial photographs usually have higher intensity and lower hue. For an input color aerial photograph, we first transform it from RGB color model into HSI color model, and then we construct a significance map to highlight the difference between cloud regions and non-cloud regions as follows

$$W = \frac{I_{intensity} + \varepsilon}{I_{hue} + \varepsilon} \quad (4)$$

where $I_{intensity}$ and I_{hue} refer to intensity and hue of the pixel in the input aerial photograph I, respectively. We bound the intensity and hue to $[0, 1]$ to compute the significance map, which is proved to be a better significance map. ε is an amplification factor, in our paper, we typically set $\varepsilon = 1.0$. To alleviate the noise disturbing without blurring the boundaries, we then apply the bilateral filter [27] to W. To obtain an intuitive visual description of the significance map, we scale the value of filtered W to the range of $[0, 255]$. The significance map W efficiently stretch the disparity between cloud regions and non-cloud regions, which will be used as an input of our optimal threshold setting scheme.

In Fig.2, we visualize the significance map. We can notice that significance map in Fig.2(b) successfully stretch the difference between cloud regions and non-cloud regions, and

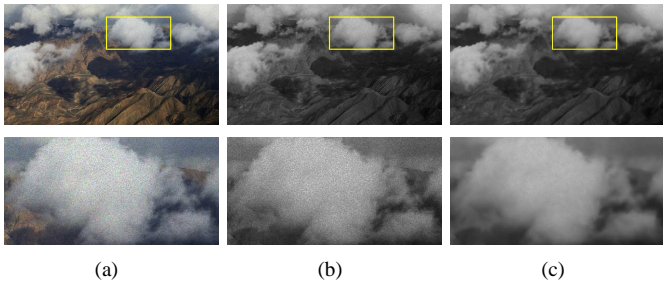the filtered significance map in Fig.2(c) reduces the noise in Fig.2(b).

Fig. 2. The significance map. (a) Input color aerial photograph. (b) Original significance map. (c) Filtered significance map.

B. Optimal threshold setting

Based on the significance map, we can identify cloud regions from input color aerial photograph by using global thresholding method, such as the Otsu's method [28]. We use the Otsu's method to segment the input photograph into candidate cloud regions and non-cloud regions, which can receive a coarse cloud detection result. The Otsu's method assumes that the photograph to be thresholded contains two classes of pixels or bi-modal histogram (for example, foreground and background), then calculates the optimal threshold separating those two classes so that their inter-class variance is maximum. Specifically, the optimal threshold T can be determined by maximizing the following formulation

$$T = argmax\{w_0(T)(u_0(T)-\bar{u})^2+w_1(T)(u_1(T)-\bar{u})^2\} \quad (5)$$

where $w_0(T) = \sum_{i=0}^{T} p_i$, $w_1(T) = \sum_{i=T+1}^{255} p_i$, $\bar{u} = \sum_{i=0}^{255} i \cdot p_i$, $u_0(T) = \sum_{i=0}^{T} i \cdot p_i$, $u_1(T) = \sum_{i=T+1}^{255} i \cdot p_i$ and p_i is the probability of the gray level i . The variable T is within the range of $[0, 255]$. To get a coarse cloud detection result, we can first apply Otsu's method over the histogram of the significance map W to select the global threshold T^{Global}. Then, we can get a coarse cloud detection result R^{Coarse} based on the significance map W , and it is defined by

$$R^{Coarse} = \begin{cases} 1, & W(x) \geq T^{Global} \\ 0, & otherwise \end{cases} \quad (6)$$

here, 1 refers to those pixels lying in candidate cloud regions, and 0 refers to those pixels lying in non-cloud regions. However, due to the threshold shifting problem of the Otsu's method, the coarse cloud detection result may suffer from excessive error detection. For example, when the histogram of the significance map has single peak instead of two isolated peaks, the threshold shifting problem will occur. Specifically, a prohibitively high threshold will be selected if most pixels have comparatively high intensity, a coarse cloud detection based on the prohibitively high threshold will miss some true cloud pixels which actually have relatively low intensity. On the contrary, a prohibitively low threshold will be selected if more pixels have comparatively low intensity, a coarse cloud detection based on the low threshold will mistake excessive non-cloud pixels for cloud pixels.

In Fig.3, we illustrates the threshold shifting problem of the Otsu's method. According to the histogram in Fig.3(d), we can see that the significance map in Fig.3(b) has a small number of pixels of relatively high intensity (larger than 100). Thus, a prohibitively low threshold 61 will be obtained by Otsu' method due to the threshold shifting problem. As a result, many non-cloud regions are mistaken for cloud regions in Fig.3(c). In Fig.3(e), we further provide a histogram to indicate the result in Fig.3(c), in which blue region refers to detected non-cloud pixels, the yellow region for detected cloud pixels. To ensure the detection accuracy, we should make the coarse cloud detection result R^{Coarse} containing the whole true cloud pixels and as few non-cloud pixels as possible. To address the threshold shifting problem, we propose an optimal threshold setting.
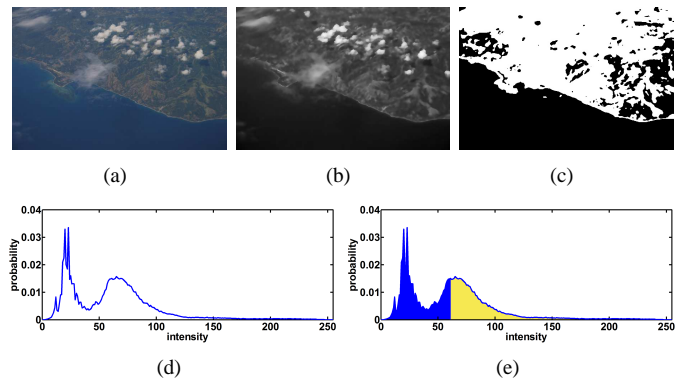

Fig. 3. Threshold shifting problem. (a) Input color aerial photograph. (b) Significance map. (c) Cloud detection result based on the threshold determined by the Otsu's method [28]. (d) Histogram of the significance map in (b). (e) Histogram indicates the detection result in (c).

According to above observations (property 1 and property 2), we recognize that true cloud pixels usually have comparatively high intensities and comparatively low hues. To identify the distribution law of intensity and hue of true cloud pixels, we collect 500 cloudy color aerial photographs from Flickr.com and several popular search engines, and we manually extract all cloud regions of these photographs. Then, we plot the histogram of intensity and hue based on pixels in all extracted cloud regions. Fig.4(a) is the intensity histogram over all cloud regions from 500 cloudy color aerial photographs. Fig.4(b) is the corresponding cumulative histogram of Fig.4(a). We can see that over 95% of the true cloud pixels have intensity not less than 100. Fig.4(c) is the hue histogram. Fig.4(d) is the corresponding cumulative histogram of Fig.4(c). Obviously, almost all of the hues are between 50 and 80. Notice that, to make the histograms more intuitive, we have bounded intensity and hue to the range [0, 255].

The intensity histogram in Fig.4 illustrates that the true cloud pixels usually have an intensity value not less than 100. In addition, to stretch the disparity between true cloud pixels and non-cloud pixels, we have constructed a significance map beforehand. Therefore, we believe that a proper global threshold for coarse cloud detection should not less than 100. The lower bound (100) can ensure the coarse cloud detection will not bring in excessive non-cloud pixels which have lower intensities. As a prohibitively high threshold will cause the
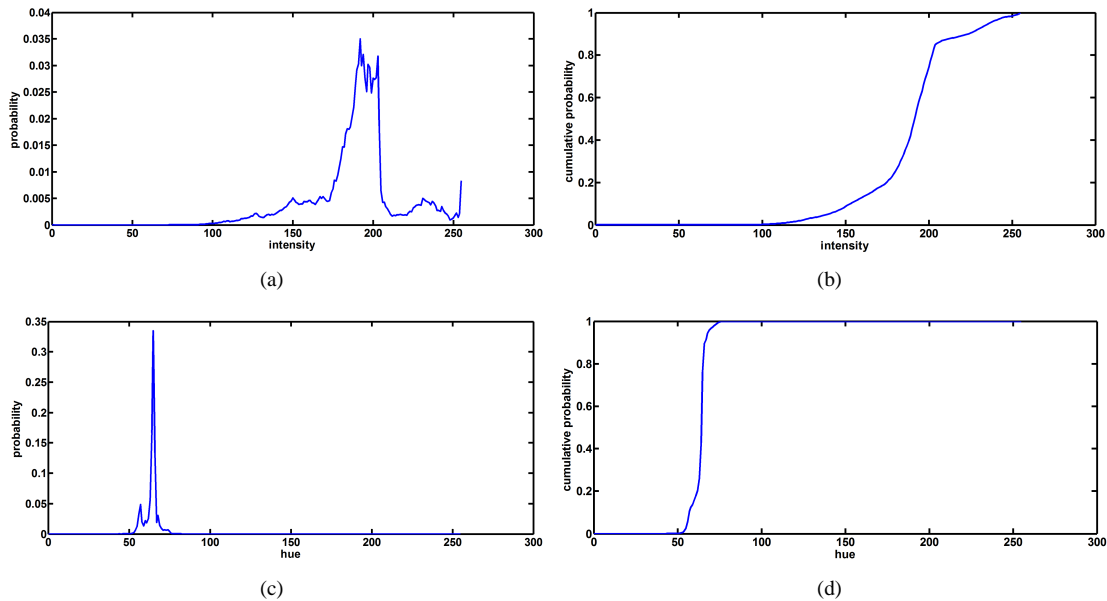
Fig. 4. Statistics of intensity and hue of cloudy photographs. (a) Histogram of the intensity of the cloud pixels in the 500 cloudy color aerial photographs. (b) Corresponding cumulative distribution of (a). (c) Histogram of the hue of the cloud pixels in the 500 cloudy color aerial photographs. (d) Corresponding cumulative distribution of (c).

coarse cloud detection result missing some true cloud pixels, thus an upper bound is required as well. Our coarse cloud detection aims at extracting all true cloud regions, and we can tolerate a certain amount of non-cloud pixels in the coarse cloud detection result, since our follow-up operations will further remove them. However, we cannot tolerate missing too many true cloud pixels in this step. Thus, we typically set the upper bound to 150 which turns out to be suitable for protecting the true cloud pixels. Our optimal threshold setting can be defined by

$$T^{Optmal} = \begin{cases} 100, & T^{Global} < 100 \\ T^{Global}, & 100 \leq T^{Global} \leq 150 \\ 150, & T^{Global} > 150 \end{cases} \quad (7)$$

where T^{Global} is the global threshold determined by the Otsu's method over the histogram of the significance map. Based on the $T^{Optimal}$, we can compute a more accurate coarse cloud detection result.

We have noticed that hue has a small value range $[50, 80]$ compared with the wide value range $[100, 255]$ of the intensity of true cloud pixels. To further improve the detection accuracy, we compute one more cloud detection result based on the value range of hue, we regard pixels having a hue value between 50 and 80 as cloud pixels, the rest as non-cloud pixels. Then, we implement an intersection between the two cloud detection results to get the final coarse cloud detection result. In Fig.5, we compare our final coarse cloud detection result with result derived from Otsu's method, we can see that there is a large amount of error detection in Fig.5(b) while our final coarse cloud detection result in Fig.5(c) successfully avoids the serious error detection by using optimal threshold setting.

For some cloudy color aerial photographs, our final coarse cloud detection result will be very close to the ideal cloud detection result. However, for some complex input color aerial
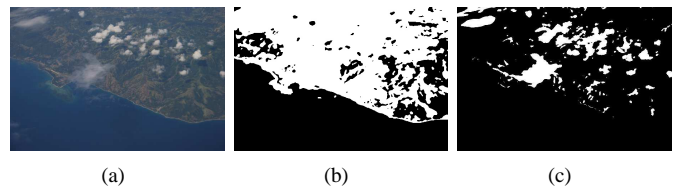

Fig. 5. Coarse cloud detection result. (a) Input color aerial photograph. (b) Coarse cloud detection result by performing Otsu's method on our significance map. (c) Coarse cloud detection result produced by the proposed optimal threshold setting.

photographs which have lots of ground objects of comparatively high intensity, our optimal threshold setting may fail to distinguish these bright ground objects from cloud regions. Thus, to remove possible non-cloud regions from our coarse cloud detection result, we further incorporate a robust detail map into the cloud detection system.

C. Cloud detection incorporating detail map

According to our above analysis, some non-cloud regions may be included in the coarse cloud detection result, thus we should remove these non-cloud regions to achieve better results. Fortunately, the property 3 have illustrated that cloud regions in color aerial photographs usually have much less details compared with complex ground. This property is an important clue for us to remove redundant non-cloud regions from our coarse cloud detection result. Based on the multiscale bilateral decomposition [29], we construct a detail map to capture the edge features of the input aerial photographs, which will be used to guide the following cloud detection.

The bilateral filtering [27] is a non-linear, edge-preserving and noise-reducing filter. The basic idea of the bilateral filtering is to replace the intensity value of each pixel by a

weighted average of intensity values from neighboring pixels. Specifically, the weights depend on both Euclidean distance and color intensity difference of pixels, and are based on Gaussian function. For an input intensity image I, the bilateral filter is defined as

$$I'(p) = \frac{1}{k} \sum_{q \in \Omega} g_{\delta_s}(\|p - q\|) \cdot g_{\delta_r}(\|I(q) - I(p)\|) \cdot I(q) \quad (8)$$

where the normalization term

$$k = \sum_{q \in \Omega} g_{\delta_s}(\|p - q\|) \cdot g_{\delta_r}(\|I(q) - I(p)\|) \quad (9)$$

where I' denotes the filtered image, p denotes the coordinate of the current pixel to be filtered. Ω denotes a square window centered in p, and q denotes coordinates of pixels within Ω. $I(p)$ and $I(q)$ are intensity values of pixels located at p and q, respectively. $g_\delta(x) = e^{(-x^2/\delta^2)}$, δ_s and δ_r are the standard deviation of the Gaussian function.

Using bilateral filtering, we can decompose an input image I into two layers: a base layer and a detail layer. For an input image, the filtered image I' refers to the base layer that maintains strongest edges while smoothing out small changes in intensity. The detail layer D then can be produced by subtraction of a based layer from an input image: $D = I - I'$. The basic idea of the multiscale bilateral decomposition is to construct M detail layers on the input image by using bilateral filtering. For an input image I, we iteratively implement the bilateral filtering to build a series of filtered image $\{I'_j\}_{j=0}^{M}$. At the finest scale, we have $I'_0 = I$. Based on equation 8, we can iteratively construct $\{I'_j\}_{j=0}^{M}$ by

$$I'_{j+1} = \frac{1}{k} \sum_{q \in \Omega} g_{\delta_{s,j}}(\|p - q\|) \cdot g_{\delta_{r,j}}(\|I'_j(q) - I'_j(p)\|) \cdot I'_j(q) \quad (10)$$

where $\delta_{s,j}$ and $\delta_{r,j}$ are the standard deviation of the j-th bilateral filtering, respectively. In our implementation, we typically set the size of Ω to 7. To increase the spatial smoothing at each scale j, we set $\delta_{s,1} = \sqrt{3}\delta_{s,0}$ and $\delta_{s,j} = 2^{j-1}\delta_{s,j-1}$, and $\delta_{s,0}$ is set to 2. Since cloud regions may have some weak edges, we set $\delta_{r,0}$ to $R/10$ and $\delta_{r,j} = \delta_{r,j-1}$ to iteratively weaken these edges, where R is the maximum intensity of the image.

We compute M subsequent image detail levels $\{D_j\}_{j=1}^{M}$, where $D_j = I'_j - I'_{j-1}$. Here, I'_j attenuates the strong edges in the input color aerial photograph and the detail layers D_j contain small changes in intensity. A simple detail map can be constructed just dependent on $D_1 = I'_0 - I'_1$. when the input color aerial photograph contains very little noise, the simplest detail map will not be a bad choice. However, when the image suffers from heavy noise, or the cloud regions have too many weak edges, the simple detail map will not work well. To construct an effective detail map which has little noise and edge information in cloud regions, we compute the detail map based on the $M(M > 1)$ subsequent image detail levels $\{D_j\}_{j=1}^{M}$. We can accelerate the multiscale decomposition by using [30].

Our goal in building a detail map I^{Detail} is to reduce the possible noise of the input image and restrain some weak edges within cloud regions. Thus, we compute I^{Detail} as a weighted sum of the M detail levels $\{D_j\}_{j=1}^{M}$,

$$I^{Detail} = \frac{1}{\lambda} \sum_{j=2}^{M} \omega_j \cdot D_j \quad (11)$$

$$\lambda = \sum_{j=2}^{M} \omega_j \quad (12)$$

where the weight ω_j is computed as

$$\omega_j = g_{\delta_d} * e^{(|D_j|)} \quad (13)$$

Here, we typically set $\delta_d = 7$ to locally smooth the weight and reduce noise. Since the input color aerial photograph may suffer from heavy noise, we do not take the first detail level D_1 into the weighted sum. In all our implementation, we set $M = 4$ to construct 4 detail levels to compute a detail map. In Fig.6 we display the detail map computed by our method, and we can see that our multiscale detail map attenuate some weaken edges and noise in cloud regions compared with the single level detail map.

In order to refer more detail information, we iteratively apply a morphology dilation operator with 7×7 structural elements on the detail map I^{Detail} to obtain a more consistent detail map. In all our implementation, we typically implement the dilation operator with 2 times.
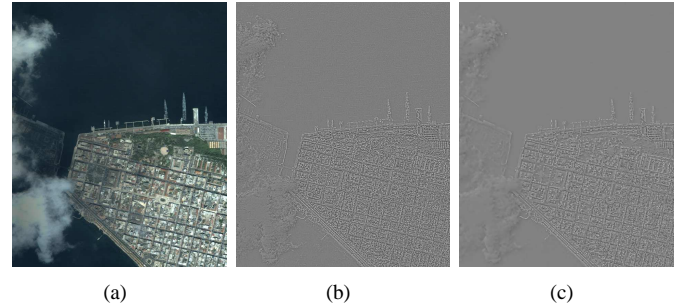

Fig. 6. Comparison of different detail map. (a) Input color aerial photograph. (b) The single level detail map derived from once decomposition. (c) Our multiscale detail map.

As cloud regions are clusters of similar cloud pixels in most cases, cloud regions usually have little details. On the other hand, our multiscale bilateral decomposition can further attenuate weak details within cloud regions. Thus, cloud regions and non-cloud regions are usually visually different in the detail map. To make the detail map display the disparity between cloud regions and non-cloud regions more intuitive, we apply the Otsu's method to compute a threshold T^{Detail} on the detail map I^{Detail}, then a binary detail map R^{Detail} can be obtained by

$$R^{Detail} = \begin{cases} 1, & I^{Detail} < T^{Detail} \\ 0, & \text{otherwise} \end{cases} \quad (14)$$

Then the binary detail map will be consist of regions with details and other regions without details. According to our analysis, we believe that all regions without details in the binary detail map are highly likely to be cloud regions. Then,
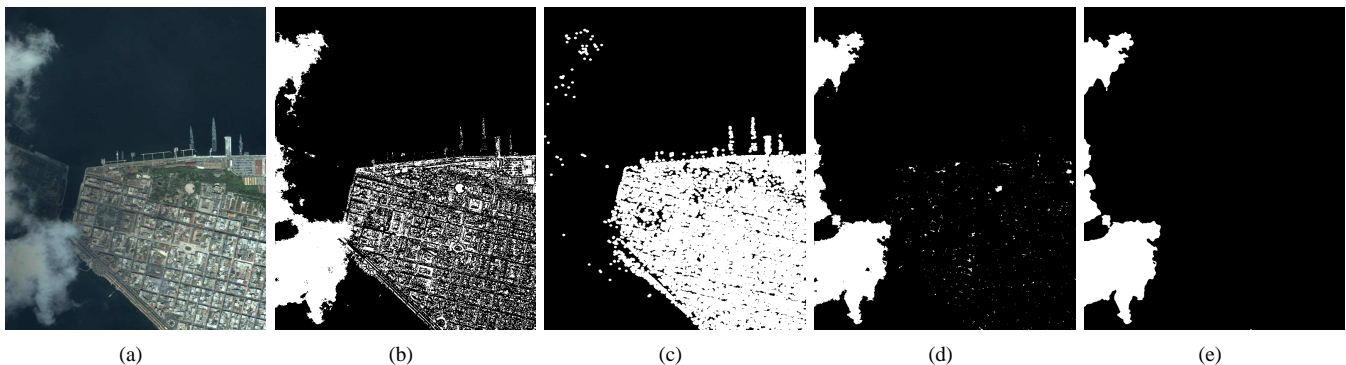
Fig. 7. Procedure of our detail-aware cloud detection. (a) Input color aerial photograph. (b) Coarse cloud detection result. (c) Binary detail map. (d) Intersection between (b) and (c). (e) Final result with hole filling and tiny clustering removing.

by combining our coarse cloud detection result R^{Coarse}, a finer cloud detection result R^{Finer} can be computed by

$$R^{Finer} = R^{Coarse} \bigcap R^{Detail} \qquad (15)$$

where R^{Coarse} denotes the coarse cloud detection result, and R^{Detail} denotes the binary detail map derived from I^{Detail}.

As some cloud regions (for example cirrus) may contain strong edges, the cloud regions around the strong edges will be mistaken as non-cloud regions. Thus,we resort to the Matlab function *imfill()* to fill holes inside cloud regions of R^{Finer}. In fact, before filling these regions, we should exclude the tiny clusters in R^{Finer} to provide a cleaner result since the cloud regions are usually not less than a certain size. To remove these tiny clusters (including regions just have single pixel), we first use a median filter to remove the salt and pepper noise, and then we extract all connected components from R^{Finer}. We calcualte the number of pixels for all extracted connected components, if the pixel number of a connected component is less than a certain threshold T^{Size}, the corresponding cloud region will be regarded as non-cloud regions. In our paper, the threshold T^{Size} is set to 120, and a larger T^{Size} can be selected if the aerial photographs having high resolution.
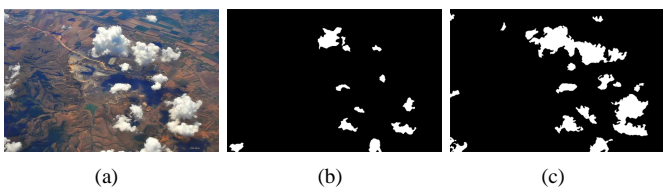

Fig. 8. Comparison between the results derived from different detail maps. (a) Input color aerial photograph. (b) R^{Finer} based on the single level detail map D_1. (c) R^{Finer} based on our multiscale detail map I^{Detail}.

In Fig.7, we give the procedure of the cloud detection incorporated with detail map. In Fig.7(b), we give the coarse cloud detection result of the input aerial photograph, and Fig.7(c) is a binary detail map, Fig.7(d) is the intersection between Fig.7(b) and Fig.7(c). Based on Fig.7(d), we further remove the tiny clusters and repair holes inside cloud regions to achieve the final result R^{Finer} (Fig.7(e)).

In Fig.8 we compare our detection results with result derived from the single level detail map. As the input aerial photograph in the first row of Fig.8(a) has some strong edges in cloud
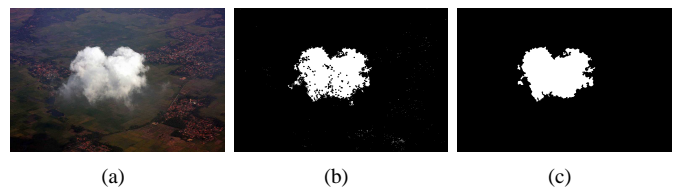

Fig. 9. Refinement by removing tiny clusters and filling holes. (a) Input color aerial photograph. (b) Result without using tiny clusters removing and holes filling. (c) Result by combining tiny clusters removing and holes filling.

regions , the result derived from the the single detail level D_1 in Fig.8(b) has mistaken more cloud pixels for non-cloud pixels compared with the result derived from multiscale detail map in Fig.8(c).

As illustrated in Fig.9, the cloud regions (cirrus cloud) of the aerial photograph in Fig.9(a) have some strong edges, which lead to a detection result with some holes inside the cloud regions in Fig.(b). By removing tiny clusters and filling holes, we receive a more accurate result.

D. Refinement by guided image filtering

Although we can get a good cloud detection result through above processes, we may miss some semitransparent cloud pixels around boundaries of the cloud regions (according to property 5) since above cloud detection processes belong to hard segmentation. To receive a more accurate cloud boundary detection results, we apply the guided filter [31] to refine the cloud boundary detection.

The guided filter is a new type of explicit image filter which involves a guidance image I, an input image P, and an output image f. It can be used as a guided feathering, in which a binary mask will be refined to appear an alpha matte near the object boundaries. The key assumption of the guided filter is a local linear mode between the guidance I and the filter output f, and f is supposed to a linear transform of I in a square window ω_k centered at the pixel k:

$$f_i = \alpha_k I_i + \beta_k, \forall i \in \omega_k, \qquad (16)$$

where α_k and β_k are constant linear coefficients in ω_k, i denotes a pixel coordinate in the square window ω_k. The local linear model presented in equation 16 ensures that f
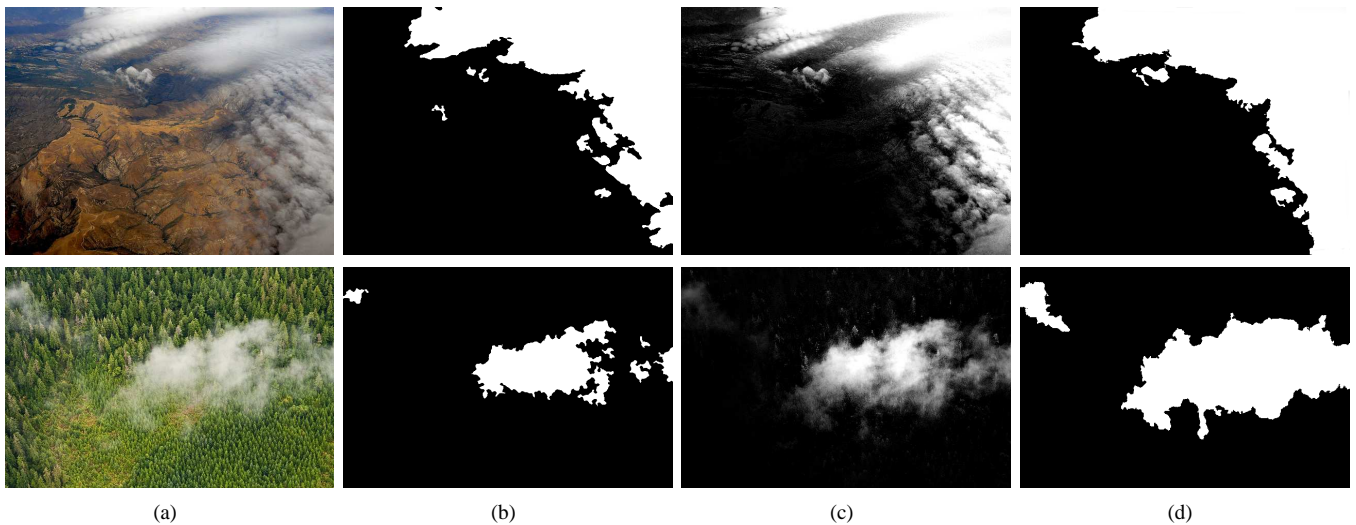
Fig. 10. Result comparison between R^{Finer} and R^{Final}. (a) Input color aerial photograph. (b) Result of R^{Finer}. (c) Output of the guided feathering over (b). (d) Final binary result R^{Final}.

has an edge only if g has an edge, since $\nabla f = \alpha \nabla I$. The two coefficients α_k and β_k are defined by

$$\alpha_k = \frac{\frac{1}{|\omega|} \sum_{i \in \omega_k} I_i P_i - \mu_k \bar{P}_k}{\delta_k^2 + \varepsilon} \quad (17)$$

$$\beta_k = \bar{P}_k - \alpha_k \mu_k \quad (18)$$

where μ_k and δ_k^2 denote the mean and variance of I in ω_k, respectively. $|\omega|$ denotes the number of pixels in ω_k, and $\bar{P}_k$ is the mean of P in ω_k. As a pixel is involved by all windows that contain it, so the output value of one pixel should combine all of them, and the final output of the pixel is defined by

$$f_i = \bar{\alpha}_i I_i + \bar{\beta}_i \quad (19)$$

where $\bar{\alpha}_i$ and $\bar{\beta}_i$ denote mean of all α_k and β_k of windows that contain the pixel i.

To implement the guided feathering, the former cloud detection result R^{Finer} is used as the input P, and the guidance image is the original color aerial photograph I, then the output is the final cloud detection result R^{Final} can be obtained by applying a guided image filtering over R^{Finer} and I. In our implementation, we typically set the window radius to 60, and $\varepsilon = 10^{-6}$ for the guided filter. We notice that the output of the guided feathering is not a binary result. To obtain a binary result, we can select an threshold T^{Binary} to produce a binary result. However, besides the true semitransparent cloud pixels, the guided feathering may also bring in some non-cloud pixels around the boundaries of cloud regions. Since semitransparent cloud pixels are usually not as bright as the common cloud pixels, we trend to select a small threshold to obtain these semitransparent cloud pixels. However, a too small threshold may bring in some non-cloud pixels. Our experience is that if the user tends to detects more semitransparent cloud pixels, a smaller threshold is recommended, and if the user tends to detects more accurate semitransparent cloud pixels, a larger threshold is recommended. To make our method detecting

clouds automatically, T^{Binary} is fixed to 60 to make a trade-off in our implementation.

In Fig.10 we compare our final cloud detection result R^{Final} with R^{Finer}, we notice that R^{Final} has repaired some hiatuses around the boundary of cloud regions compared with R^{Finer}, and what's more, R^{Final} contains most semitransparent cloud pixels.

IV. RESULTS AND DISCUSSION

In this section, we further demonstrate the effectiveness of our cloud detection algorithm by both visual comparisons and quantitative evaluation. We implemented our cloud detection algorithm in C++ combined with Matlab on a PC with Pentium Dual-Core CPU E6500 @2.93GHz and 4GB RAM. To evaluate the efficiency of our cloud detection algorithm, we compare our method with the cloud detection method [23], some popular automatic image segmentation methods [9]–[11] and interactive image segmentation methods [8], [13]. Furthermore, we present a quantitative evaluation of the detection accuracy and runtime to prove the efficiency of our method.
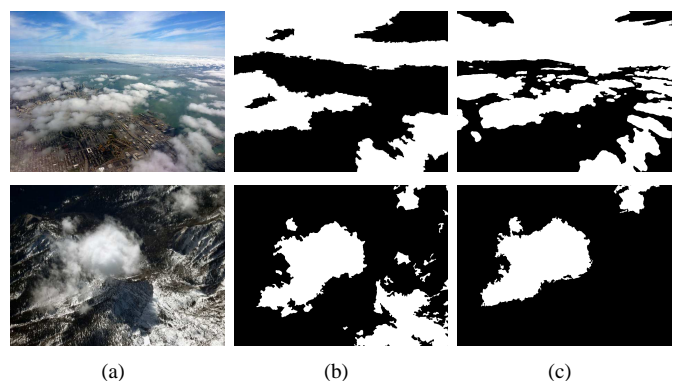

Fig. 11. Comparisons with other competing cloud detection method [23]. (a) Input color aerial photograph. (b) Result of [23]. (c) Our result.
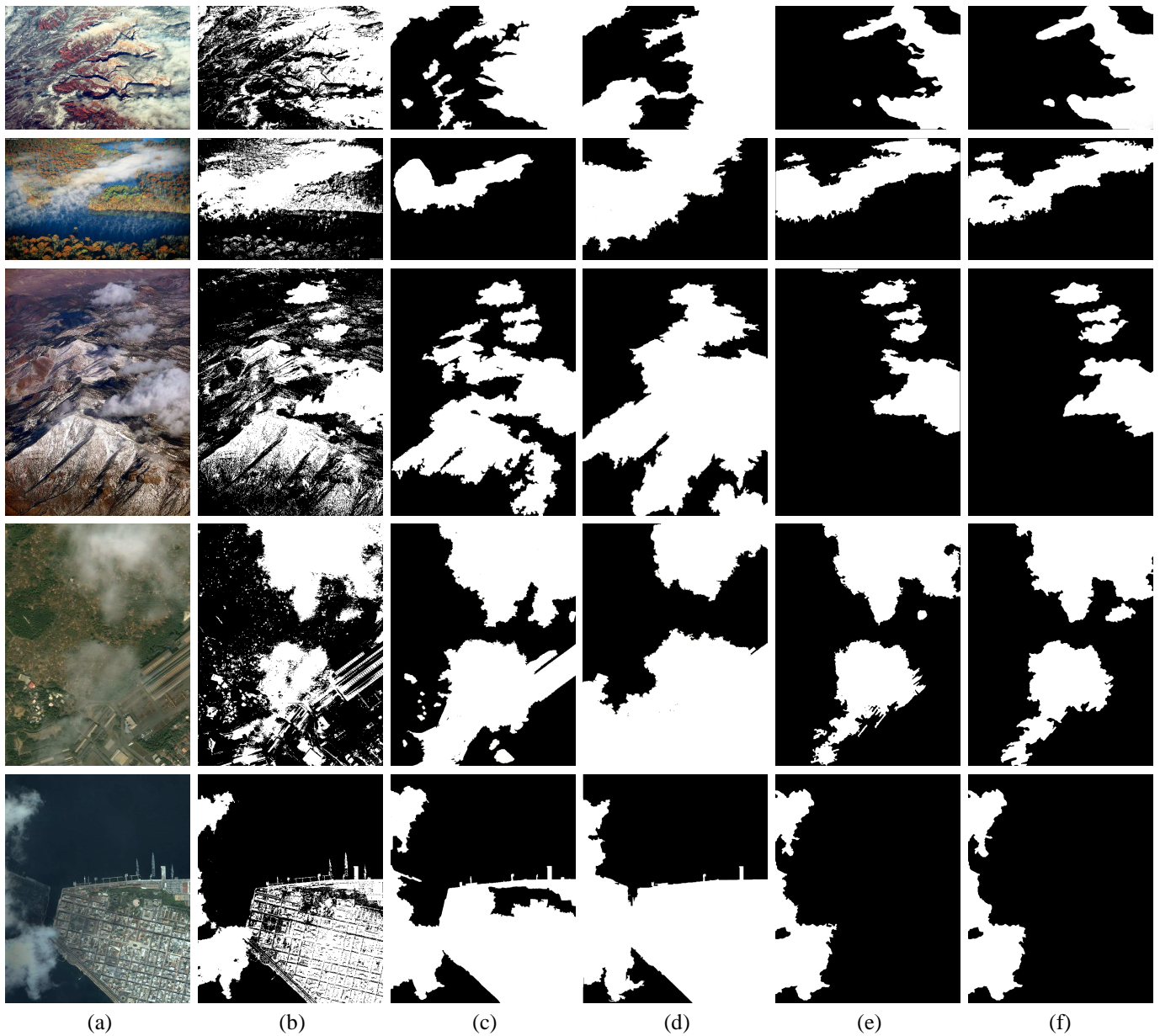
Fig. 12. Detection comparisons with automatic image segmentation methods. (a) Input color aerial photographs. (b) Result of k-means [11]. (c) Result of mean-shift [9]. (d) Result of graph-based segmentation [10]. (e) Result of our method. (f) The ground truth.

A. Visual comparisons

In Fig.11, we compare our method with [23], which automatically detect cloud/shadow based on a Markov Random Field (MRF) framework. We observe that the method [23] cannot produce satisfactory results for complex aerial photographs while our method can achieve better results. Besides, as the method [23] involves a complex optimization process, our method is more efficient than [23]. For example, for an input photograph with size of 1600×1200, [23] takes more than 10s to detect the clouds, while our method takes only 2s.

In Fig.12, we compare our cloud detection algorithm with the automatic image segmentation methods [9]–[11], and we present five groups of cloud detection results. In these results, the results of k-means are obtained by computing two clustering center for the photograph based on intensity. The results of both mean-shift and graph-based segmentation are computed in two steps. We first segment the input photographs using mean-shift or graph-based segmentation, and then we detect cloud regions according to the average intensity of the region, regions having an average intensity more than 100 are regarded as cloud regions. In Fig.12, the cloud layers are diverse, including thick clouds (first and fifth row), snows (third row), and semitransparent clouds (second and fourth row). For photographs with bright non-cloud regions, for example, the third photograph in Fig.12 with snows, current cloud detection or image segmentation methods usually fail to produce satisfactory results. However, though these non-cloud regions are visually similar to cloud regions, they are quite different in our multiscale detail map. Thus, by incorporating the detail map into our cloud detection system, we can remove
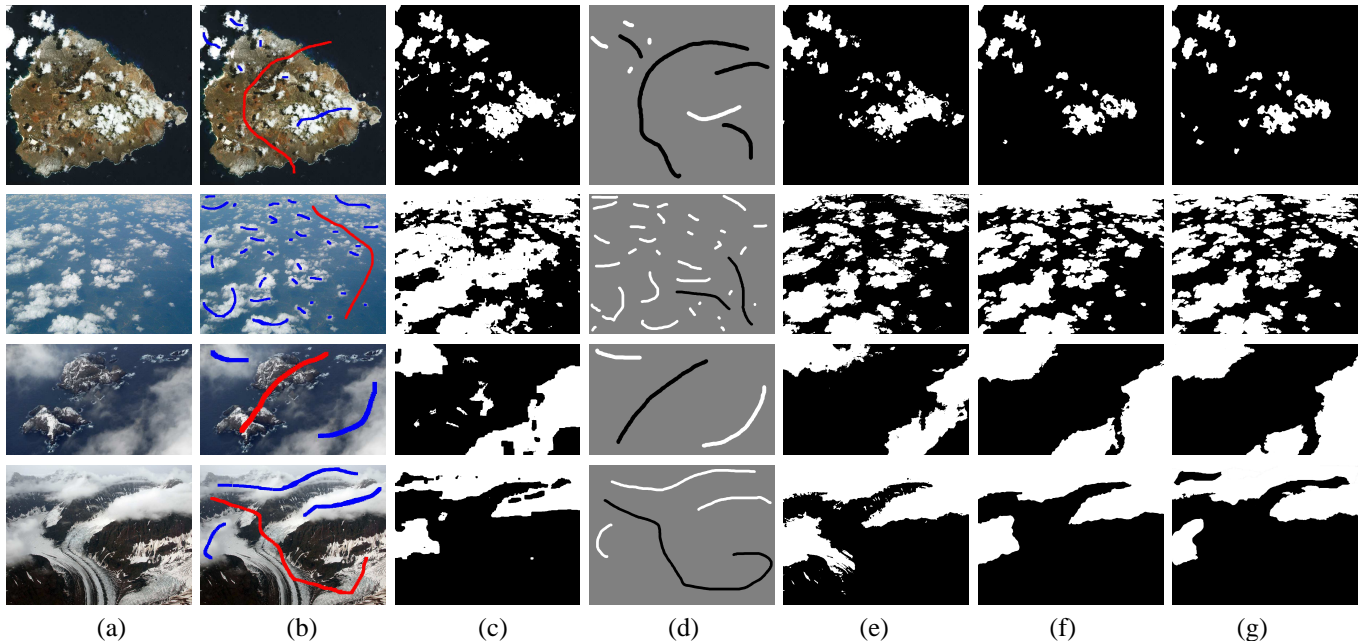
Fig. 13. Detection comparisons with interactive image segmentation methods. (a) Input color aerial photograph. (b) User interaction for graph-cut [8], where blue scribbles indicate cloud regions (foreground), and red scribbles indicate non-cloud regions (background). (c) Result of graph-cut based on mask (b). (d) Trimap for robust-matting [13], where white scribbles indicate cloud regions, black scribbles indicate non-cloud regions, and the other gray regions denote unknown regions. (e) Result of robust-matting based on trimap (d). (f) Our cloud detection result. (g) The ground truth.

them from the candidate cloud regions. As for photographs with semitransparent cloud pixels, our guided feathering will help us involve them in the final cloud detection result. As illustrated in Fig.12, compared with these image segmentation methods, our method produce much better results.

In Fig.13, we further compare our approach with some interactive image segmentation methods [8], [13]. Since the first and the second aerial photograph in Fig.13 have some isolated small cloud regions, we have to take pains to produce lots of interactions for graph-cut [8] and robust-matting [13]. However, our method achieves better results though we requires no interaction. As for complex cloudy aerial photographs in the third and fourth row in Fig.13, they have both white non-cloud regions and semitransparent cloud pixels. Though we have provided enough interactions for these complex photographs to perform graph-cut and robust-matting, they still mistake more non-cloud pixels for cloud pixels than our method.

B. Quantitative evaluation

To quantitatively evaluate the efficiency of our cloud detection method, we use the error rate to evaluate the accuracy of our method, automatic image segmentation methods [9]–[11], and interactive image segmentation methods [8], [13]. The error rate (ER) is defined by

$$ER = \frac{CN + NC}{TN} \qquad (20)$$

where CN denotes the number of cloud pixels identified as non-cloud pixels, NC denotes the number of non-cloud pixels identified as cloud pixels, and TN denotes the number of pixels in the input photograph. We also give the quantity of the user interaction for interactive image segmentation methods, which is defined as

$$QI = \frac{SP}{TN} \qquad (21)$$

where SP denotes the number of user scribbled pixels.

In Table I, we give the aforementioned error rate ER for the results in Fig.12, and the table shows that our method has lower error rate compared with k-means, mean-shift and graph-based segmentation. Since these methods may mistake lots of non-cloud pixels (usually have high intensities) for cloud pixels, and they are also weak in detecting the semitransparent cloud pixels, so they usually have higher error rate than our method. In Table II, we show error rate and quantity of the user interaction for results in Fig.13. Although our method needs no interaction, our method still outperforms graph-cut and robust-matting for these color aerial photographs.

C. Time complexity

The main computation of our cloud detection method is the detail-aware cloud detection and guided feathering refinement. For the detail-aware cloud detection, the computation mainly lies in the multiple bilateral filtering whose time complexity is O(Nr^2), where r is the kernel radius. Fortunately, the acceleration methods [30], [32] with O(N) have been developed based on histograms recently. In this paper, we resort to [30] for our task. As illustrated in [31], the guided filter has an O(N) complexity which can provide fast feedback (<1s) for a 6-mega-pixel photograph. In Table III and IV, we give the runtime for all the aerial photographs in Fig.12 and Fig.13, where the stage 1 refers to runtime for the detail-aware cloud detection, and the stage 2 refers to runtime for the guided feathering refinement. The results show that our

TABLE I
QUANTITATIVE EVALUATION OF ACCURACY FOR RESULTS IN FIG.12

Method	ER of 1th photograph	ER of 2th photograph	ER of 3th photograph	ER of 4th photograph	ER of 5th photograph
K-means	29.71%	23.03%	26.64%	13.76%	33.13%
Mean-shift	23.92%	14.48%	24.63%	12.55%	27.07%
Graph-based	34.23%	26.31%	38.33%	39.67%	41.56%
Our method	1.73%	4.73%	1.85%	4.03%	1.41%

TABLE II
QUANTITATIVE EVALUATION OF ACCURACY FOR RESULTS IN FIG.13

Method	$QI\&ER$ of 1th photograph		$QI\&ER$ of 2th photograph		$QI\&ER$ of 3th photograph		$QI\&ER$ of 4th photograph	
	QI	ER	QI	ER	QI	ER	QI	ER
Graph-cut	2.65%	3.41%	5.91%	17.69%	6.13%	14.91%	7.03%	6.87%
Robust-matting	5.75%	2.02%	6.75%	9.52%	5.13%	13.01%	5.67%	7.33%
Our method	0	1.34%	0	5.18%	0	2.14%	0	5.71%

method provides fast feedback (<2s) for 1-mega-pixel aerial photographs. Since the core steps of our method can be implemented in O(N), we can put our method into practical applications.

TABLE III
QUANTITATIVE EVALUATION OF RUNTIME FOR RESULTS IN FIG.12

Input photograph	Size	Stage 1	Stage 2	Total time
The first	700 × 465	0.796s	0.212s	1.008s
The second	930 × 614	1.141s	0.311s	1.452s
The third	480 × 640	0.734s	0.219s	0.953s
The fourth	579 × 758	0.883s	0.231s	1.124s
The fifth	629 × 777	0.949s	0.267s	1.216s

TABLE IV
QUANTITATIVE EVALUATION OF RUNTIME FOR RESULTS IN FIG.13

Input photograph	Size	Stage 1	Stage 2	Total time
The first	690 × 668	0.734s	0.228s	0.962s
The second	1024 × 768	1.402s	0.391s	1.793s
The third	700 × 419	0.704s	0.218s	0.922s
The fourth	1024 × 768	1.314s	0.353s	1.667s

D. Limitations

To detect clouds from the aerial photograph with bright non-cloud regions that have little details is very challenging for our cloud detection system. These non-cloud regions basically meet the properties we have summarized from common cloud regions, such as higher intensity, smoother, lower chromatic, and appearing in terms of clusters. Thus, our system may fail to distinguish cloud regions from these non-cloud regions. In the first aerial photograph of Fig.14, cloud regions are surrounded by some non-cloud regions that are extremely similar to cloud regions. As a result, our system have mistaken these non-cloud regions for cloud regions. When clouds in aerial photographs are extremely thin, we will fail to get an accurate detail map, and we will miss some cloud regions, as illustrated in the second row of Fig.14.

V. CONCLUSION AND FUTURE WORK

We have presented a novel progressive refinement scheme to automatically detect the cloud regions in the color aerial
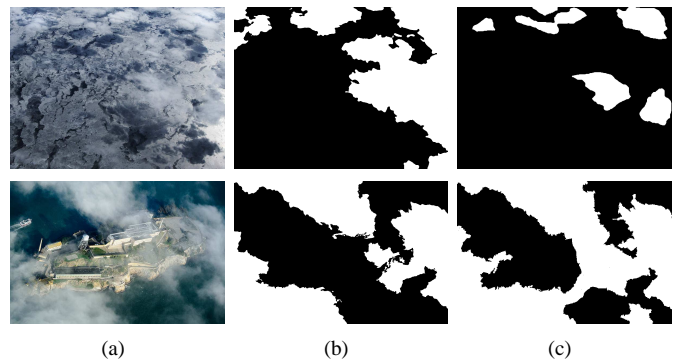

Fig. 14. Failed cases. (a) Input color aerial photograph. (b) Our result. (c) The ground truth.

photographs. We first compute a significance map to highlight the difference between cloud regions and non-cloud regions. Then we proposed an optimal threshold setting to obtain a coarse cloud detection result based on the significance map, and the result consists of candidate cloud regions and non-cloud regions. To remove the redundant non-cloud regions from the candidate cloud regions, we incorporate a multiscale detail map into the cloud detection system to achieve a finer result. Finally, we perform a guided feathering to receive a more accurate boundaries for the cloud regions. To demonstrate the effectiveness and efficiency of our method, we evaluate our method in visual comparisons and quantitative evaluation. Experiment results have proved that our method can produce satisfactory results. In the future, we will take more semantic information into consideration to further improve the accuracy of our cloud detection method. For example, we will add the image depth information to help us achieve a higher accuracy. In addition, we will adapt our work to multispectral satellite images in the future.

REFERENCES

[1] M. Ferecatu and N. Boujemaa, "Interactive remote-sensing image retrieval using active relevance feedback," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 45, no. 4, pp. 818–826, 2007.

[2] L. Tao, L. Yuan, and J. Sun, "Skyfinder: attribute-based sky image search," in *ACM Transactions on Graphics (TOG)*, vol. 28, no. 3. ACM, 2009, p. 68.

[3] F. Melgani and L. Bruzzone, "Classification of hyperspectral remote sensing images with support vector machines," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 42, no. 8, pp. 1778–1790, 2004.

[4] G. Camps-Valls, T. Bandos Marsheva, and D. Zhou, "Semi-supervised graph-based hyperspectral image classification," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 45, no. 10, pp. 3044–3054, 2007.

[5] R. Saunders and K. Kriebel, "An improved method for detecting clear sky and cloudy radiances from avhrr data," *International Journal of Remote Sensing*, vol. 9, no. 1, pp. 123–150, 1988.

[6] R. A. Frey, S. A. Ackerman, Y. Liu, K. I. Strabala, H. Zhang, J. R. Key, and X. Wang, "Cloud detection with modis. part i: Improvements in the modis cloud mask for collection 5," *Journal of Atmospheric and Oceanic Technology*, vol. 25, no. 7, pp. 1057–1072, 2008.

[7] S. Ackerman, R. Holz, R. Frey, E. Eloranta, B. Maddux, and M. McGill, "Cloud detection with modis. part ii: validation," *Journal of Atmospheric and Oceanic Technology*, vol. 25, no. 7, pp. 1073–1086, 2008.

[8] Y. Y. Boykov and M.-P. Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in nd images," in *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, vol. 1. IEEE, 2001, pp. 105–112.

[9] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 5, pp. 603–619, 2002.

[10] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *International Journal of Computer Vision*, vol. 59, no. 2, pp. 167–181, 2004.

[11] W.-T. Lin, C.-H. Lin, T.-H. Wu, and Y.-K. Chan, "Image segmentation using the k-means algorithm for texture features," *World Academy of Science, Engineering and Technology*, vol. 65, 2010.

[12] C. Rother, V. Kolmogorov, and A. Blake, "Grabcut: Interactive foreground extraction using iterated graph cuts," in *ACM Transactions on Graphics (TOG)*, vol. 23, no. 3. ACM, 2004, pp. 309–314.

[13] J. Wang and M. F. Cohen, "Optimized color sampling for robust matting," in *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*. IEEE, 2007, pp. 1–8.

[14] K.-L. Chung, Y.-R. Lin, and Y.-H. Huang, "Efficient shadow detection of color aerial images based on successive thresholding scheme," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 47, no. 2, pp. 671–682, 2009.

[15] J. Huang, W. Xie, and L. Tang, "Detection of and compensation for shadows in colored urban aerial images," in *Intelligent Control and Automation, 2004. WCICA 2004. Fifth World Congress on*, vol. 4. IEEE, 2004, pp. 3098–3100.

[16] V. J. Tsai, "A comparative study on shadow compensation of color aerial images in invariant color models," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 44, no. 6, pp. 1661–1671, 2006.

[17] R. Fattal, "Single image dehazing," in *ACM Transactions on Graphics (TOG)*, vol. 27, no. 3. ACM, 2008, p. 72.

[18] K. He, J. Sun, and X. Tang, "Single image haze removal using dark channel prior," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 1956–1963.

[19] R. T. Tan, "Visibility in bad weather from a single image," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–8.

[20] A. Levin, D. Lischinski, and Y. Weiss, "A closed-form solution to natural image matting," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 30, no. 2, pp. 228–242, 2008.

[21] C. Xiao, M. Liu, D. Xiao, Z. Dong, and K. Ma, "Fast closed-form matting using hierarchical data structure," *IEEE Trans. Circuits Syst. Video Techn.*, vol. 24, no. 1, pp. 49–62, 2014.

[22] F.-M. Bréon and S. Colzy, "Cloud detection from the spaceborne polder instrument and validation against surface synoptic observations," *Journal of Applied Meteorology*, vol. 38, no. 6, pp. 777–785, 1999.

[23] S. Le Hégarat-Mascle and C. André, "Use of markov random fields for automatic cloud/shadow detection on high resolution optical images," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 64, no. 4, pp. 351–366, 2009.

[24] C. Xiao, R. She, D. Xiao, and K.-L. Ma, "Fast shadow removal using adaptive multi-scale illumination transfer," in *Computer Graphics Forum*, vol. 32, no. 8. Wiley Online Library, 2013, pp. 207–218.

[25] C. Xiao and J. Gan, "Fast image dehazing using guided joint bilateral filter," *The Visual Computer*, vol. 28, no. 6-8, pp. 713–721, 2012.

[26] W. Pratt, "Digital image processing john wiley," *New York*, pp. 307–446, 1991.

[27] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Computer Vision, 1998. Sixth International Conference on*. IEEE, 1998, pp. 839–846.

[28] N. Otsu, "A threshold selection method from gray-level histograms," *Automatica*, vol. 11, no. 285-296, pp. 23–27, 1975.

[29] R. Fattal, M. Agrawala, and S. Rusinkiewicz, "Multiscale shape and detail enhancement from multi-light image collections." *ACM Trans. Graph.*, vol. 26, no. 3, p. 51, 2007.

[30] Q. Yang, K.-H. Tan, and N. Ahuja, "Real-time o (1) bilateral filtering," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 557–564.

[31] K. He, J. Sun, and X. Tang, "Guided image filtering," in *Computer Vision–ECCV 2010*. Springer, 2010, pp. 1–14.

[32] F. Porikli, "Constant time o (1) bilateral filtering," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–8.

Qing Zhang received the BSc and MSc degrees in computer science and technology from Wuhan University, Wuhan, China, in 2011 and 2013, respectively. Currently, he is working toward the PhD degree at School of Computer, Wuhan University. His research interests include image and video processing and computational photography.

Chunxia Xiao received the BSc and MSc degrees from the Mathematics Department of Hunan Normal University in 1999 and 2002, respectively, and the PhD degree from the State Key Lab of CAD & CG of Zhejiang University in 2006. Currently, he is a professor at the School of Computer, Wuhan University, China. From October 2006 to April 2007, he worked as a postdoc at the Department of Computer Science and Engineering, Hong Kong University of Science and Technology, and during February 2012 to February 2013, he visited University of California-Davis for one year. His research areas include computer graphics, computer vision, image and video processing, and he has published more than 50 papers in journals and conferences.