Segmentation of Hand Posture against Complex Backgrounds Based on Saliency and Skin Colour Detection

Qingrui Zhang, Mingqiang Yang, Kidiyo Kpalma, Qinghe Zheng, Xinxin Zhang

Abstract—**Hand posture segmentation is an important step in hand posture recognition process, which aims to eliminate the interference from other elements in an image, and get the hand posture region precisely. In this paper, we propose a hand segmentation method based on saliency and skin colour detection. This method mainly consists of a pixel-level hand detection method, a region-level hand detection method, and a multiple saliency maps fusion framework, thereby achieving the deep integration of the bottom-up saliency information and top-down skin colour information. Experimental results show that the proposed method has excellent performance, and is reliable against cluttered backgrounds. Moreover, evaluations based on comparisons with other approaches of the literature have demonstrated its effectiveness.**

Index Terms—**hand posture segmentation, image saliency, skin colour detection**

I. Introduction

As one of the most frequently used part of human body, hands provide a very effective approach for human-computer interaction. In order to achieve gesture recognition, there are many methods proposed [1, 2]. As the initial step in hand posture recognition, segmentation directly determines the accuracy of hand recognition. Paper in [1] shows that a good segmentation result used in the machine learning (convolutional neural network) can improve the recognition accuracy of hand posture. Many mainstream hand segmentation methods, such as skin-colour model, texture detection, and template matching [3-7], cannot work well as soon as the background is complex and cluttered. The existing skin-based posture segmentation techniques are based on a presumption that the colour of the skin can be modeled correctly and used for posture segmentation. However, when the hand regions and the non-hand regions have apparent overlap in colour space, these algorithms that merely use a skin colour model to detect the hand posture and ignore the spatial information in the image, usually resulting in high false positive rate. Correspondingly, when the external illumination in the image is not uniform, sometimes there will be a higher false negative rate.

Recently, saliency detection becomes a desirable way for computer vision based models to find the most noticeable objects in a scene. Since Itti [8] first deriving the visual saliency of a single image, numerous works have been proposed to extract the saliency information of images for compression, segmentation, or classification [9-11]. Given the advantage of saliency detection, some authors take the saliency detection as a supplement for hand posture segmentation [12, 14]. However, they just combine the results of saliency detection and traditional methods linearly. As a result, the salient objects with non-skin colour and non-salient objects with skin colour may be incorrectly segmented.

This paper aims to develop a robust and accurate hand segmentation method against complex backgrounds, which profoundly integrates the bottom-up saliency information and top-down skin colour information. The main contributions of this study are as follows:

1) A pixel-level hand detection method, which integrates dispersion measurement based saliency detection with skin colour detection.
2) A region-level hand detection method, which utilizes a saliency detection method based on a skin probability distance measurement.
3) We adopt a Bayesian framework to fuse the obtained saliency maps, and get the final confidence map for segmentation.

The rest of the paper is organized as follows: Section II introduces the framework of our hand posture segmentation method. Section III discusses the experimental results on two datasets to verify the effectiveness of proposed method. The paper is concluded with comments in Section IV.

II. The Proposed Hand Posture Segmentation Method

As described above, using skin colour model for hand posture segmentation is an effective approach. However, the skin-like backgrounds and uneven illumination in the image bring a lot of uncertainty to the segmentation results. Saliency

Manuscript received May 3, 2017; revised May 28, 2018. This work was supported by Shandong Provincial Natural Science Foundation, China (ZR2014FM030, ZR2014FM010).

Qingrui Zhang is with the School of Information Science and Engineering, Shandong University, Qingdao 266237, China (e-mail: zhangqingrui1993@126.com).

Mingqiang Yang is with the School of Information Science and Engineering, Shandong University, Qingdao 266237, China (corresponding author, e-mail: imageinstitute@outlook.com).

Kidiyo Kpalma is with the EII / IETR-INSA - UMR CNRS 6164, France (e-mail: kidiyo.kpalma@insa-rennes.fr).

Qinghe Zheng is with the School of Information Science and Engineering, Shandong University, Qingdao 266237, China (e-mail: 15005414319@163.com).

Xinxin Zhang is with the School of Information Science and Engineering, Shandong University, Qingdao 266237, China (e-mail: 201511898@mail.sdu.edu.cn).

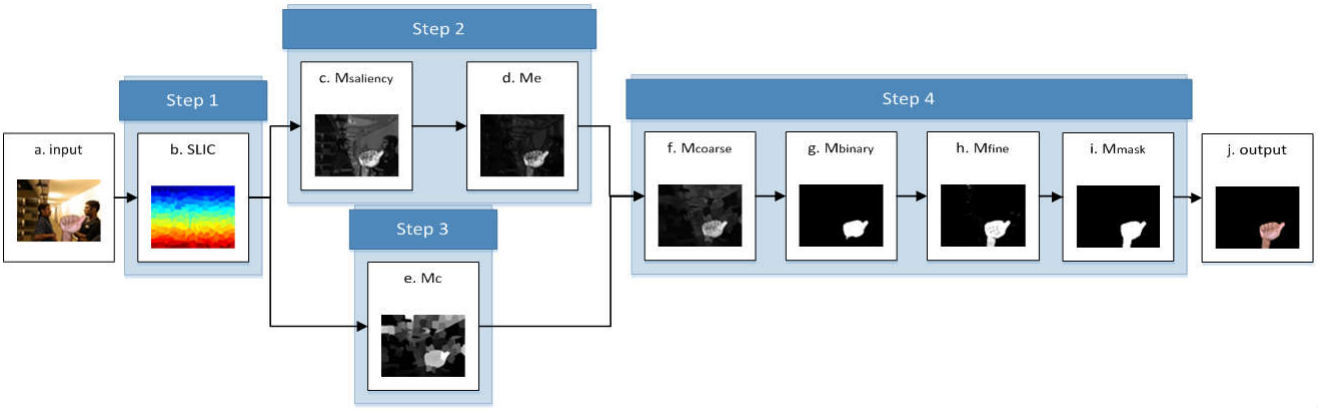


Fig. 1. The framework of the proposed method. Step 1: the pre-segmentation. Step 2: a pixel-level hand detection method. Step 3: a region-level hand detection method. Step 4: a Bayesian framework for fusing saliency maps.

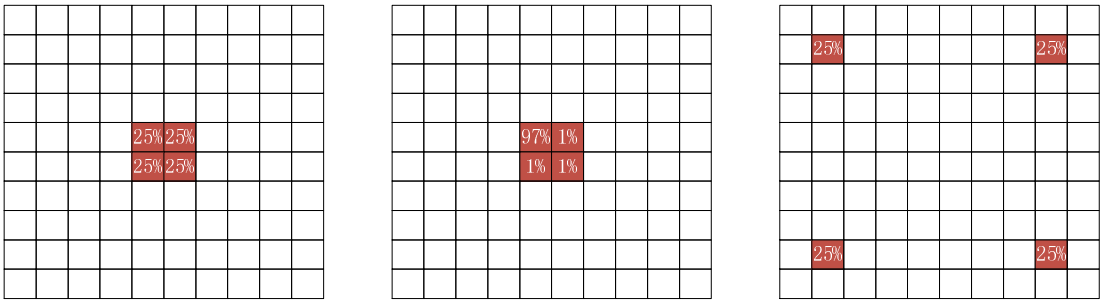


Fig. 2. Three images with different distributions of colour *i*. These images are all evenly divided into 10×10 regions, and the number in each region betrays the ratio between pixels with colour *i* in this region and pixels with colour *i* in the whole image.

detection approach provides a useful breakthrough for solving this problem. In this section, we first present a pixel-level hand detection method, which is then integrated with a region-level hand detection method. After the integration, the saliency maps are processed by a Bayesian framework, then we get the final confidence map for segmentation. The framework of the proposed segmentation method is shown in Fig. 1.

The method in this paper mainly comprises four steps. The image with a cluttered background in Fig. 1(a) is chosen as an input image. To reduce the complexity of the algorithm, we quantize each channel in CIELab space to have 12 different values beforehand. This parameter 12 is determined based on the balance of efficiency and accuracy. The larger the parameter, the larger the calculation and accuracy of the algorithm.

A. The First Step: The Pre-segmentation

There are many clustering-based image segmentation methods that can be used for pre-segmentation [13, 15]. In the first step, the input image is segmented into N regions (R_1,R_2⋯R_N) by using simple linear iterative clustering (SLIC), a method for generating superpixels based on k-means clustering [15].

The SLIC algorithm is based on the similarity between the pixel colour and the proximity of spatial position so that the image is clustered to produce superpixels. This method adapts a k-means clustering approach to generate superpixels efficiently. The number of superpixels can be pre-set, the superpixels obtained is relatively compact and have a similar size. In view of the SLIC method has been shown to outperform existing superpixel methods in many respects, such as speed and quality of the results, we segment the input image into regions using this approach. The result of pre-segmentation is shown in Fig. 1(b).

B. The Second Step: A Pixel-level Hand Detection Method

First, we introduce a dispersion measurement based saliency detection method. In information theory [24], the entropy of a random variable is defined as:

$$\mathrm{H}(X) = -\sum_{i=1}^{n} p(x_i)\log_2 p(x_i) \qquad (1)$$

where X is a discrete random variable with possible values $\{x_1, x_2 \cdots x_n\}$, and $p(x_i)$ is the probability mass function of X, which betrays the probability of $X = x_n$. Intuitively, the essence of entropy is "inner degree of confusion" of a system. The higher the degree of confusion, the greater the entropy.

Based on the idea that 'background colours have a broader spatial distribution in the image and a more balanced distribution among all the N regions, while foreground colours have a more concentrated distribution', we define (2) to measure a colour's dispersion degree in an image and thereby compute the saliency of each colour *i* in (3):

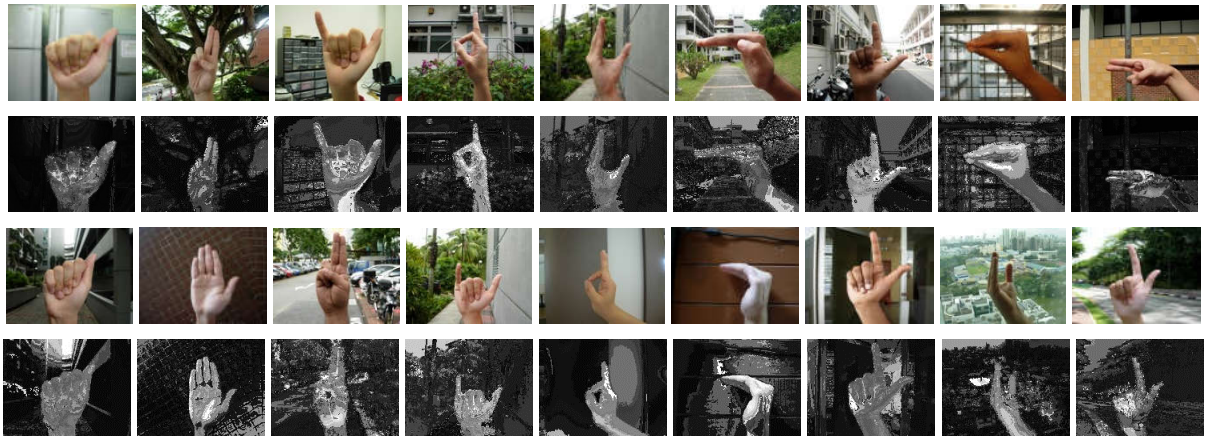
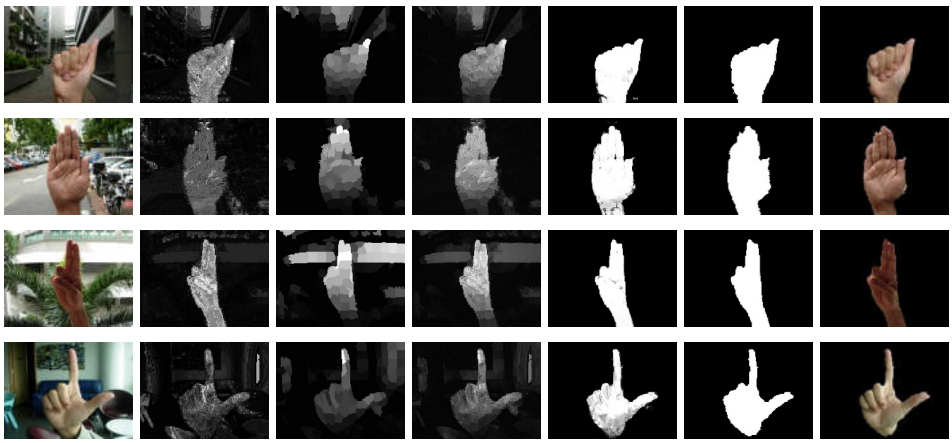


Fig. 3. Examples of image saliency detection for salient objects. The first and third row: the input images. The second and fourth row: the corresponding saliency maps $M_{saliency}$.



(a) (b) (c) (d) (e) (f) (g)

Fig. 4. Results of each step in the proposed method. (a) The input image. (b) The saliency map M_e obtained from step 2. (c) The saliency map M_c obtained from step 3. (d) The coarse confidence M_{coarse} obtained from step 4. (e) The final confidence map M_{fine} . (f) The final binary mask (using Otsu's segmentation method). (g) The segmentation result.

$$E(i) = \sum_{R_j, j \in 1,2 \cdots N} [-p_{ij} \log_2 p_{ij} \sum_{R_k \neq R_j} D_s(R_k, R_j) p_{ik}] \quad (2)$$

$$S(i) = \exp[-E(i) / \sigma_e] \quad (3)$$

In (2), $E(i)$ is a dispersion measure function of the colour *i*. A background colour is expected to have a higher dispersion degree $E(i)$. Inspired by the entropy theory, meanwhile considering the spatial distribution information, we define an entropy term to measure the breadth and uniformity of this colour's distribution. $D_s(R_k, R_j)$ is the spatial distance between region R_k and R_j , *i.e.*, the Euclidean distance between their centroids. p_{ij} is the ratio between pixels with colour *i* in region R_j and pixels with colour *i* in the whole image.

As shown in Fig. 2, we assume that these three images are all evenly divided into 10×10 regions, and the number in each region betrays the ratio between pixels with colour *i* in this region and pixels with colour *i* in the whole image. The term $p_{ij} \log_2 p_{ij}$ in (2) is used for measuring the uniformity of colour *i*'s distribution, which makes $E(i)$ in Fig. 2(a) higher than it in Fig. 2(b). However, this term ignores the spatial distribution of colour *i* in the image, resulting in that $p_{ij} \log_2 p_{ij}$ in Fig. 2(a) is the same as it in Fig. 2(c). As a solution, in (2), by introducing a spatial weighting term $\sum_{R_k \neq R_j} D_s(R_k, R_j) p_{ik}$, the colour dispersion measure function $E(i)$ can measure the breadth and uniformity of a colour's distribution at the same time. Thereby, $E(i)$ in Fig. 2(a) is smaller than $E(i)$ in Fig. 2(c), so the colour *i* in Fig.2(a) is more likely to belong to the foreground.

In (3), $S(i)$ betrays the saliency of colour *i*, and the parameter σ_e controls the strength of $E(i)$. By combining (2) with (3), we obtain the saliency value of each pixel and the corresponding saliency map $M_{saliency}$. Thus, we can effectively locate the salient objects in the image. Fig. 1(c) and Fig. 3 show examples of image saliency detection for salient objects on several typical images. It can be seen from the figure that the saliency detection method based on dispersion measure has a good ability to locate salient objects (hand posture regions) in the image. It is worth noting that the edge of the gesture region is still very clear.

After detecting saliency of each pixel, we try to integrate saliency detection and skin detection. As shown in Fig. 3, the saliency detection method which based on the proposed

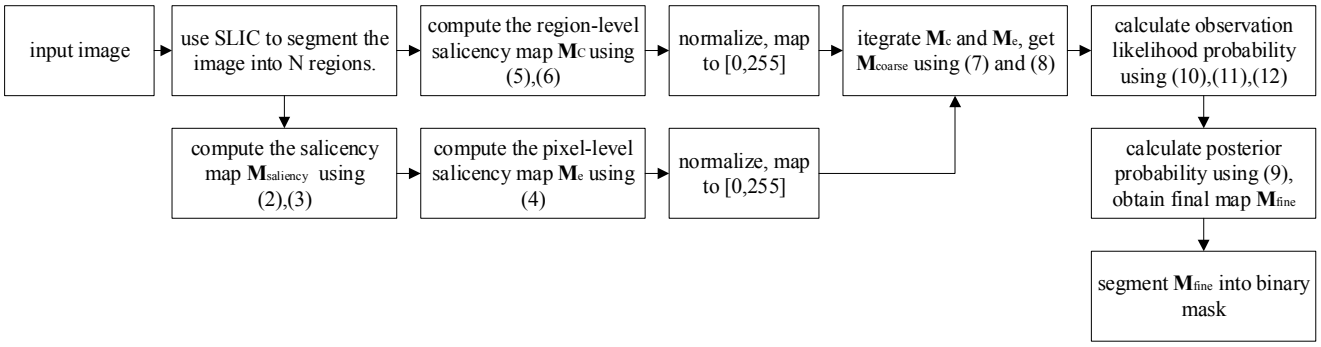


Fig. 5. The flow diagram of the proposed segmentation method.

dispersion measure can accurately detect the salient objects. However, when the background of an image is complex and cluttered, there will be a lot of interference in the saliency map. Thus, the attempt to segment hand postures only using the saliency detection method is too difficult to proceed. To obtain more stable results, the saliency detection is then integrated with a skin colour detection, thereby computing saliency value of each pixel. For pixels in the input image with colour *i*, they have same saliency value $S_e(i)$:

$$S_e(i) = P_{skin}(i) \cdot \sqrt{S(i)} \tag{4}$$

In (4), $P_{skin}(i)$ is the skin probability of colour *i*. In this paper, we use the Conaire's skin model to compute the probability of a colour to be a skin colour [16]. The purpose of the square calculation is to appropriately reduce the influence of $S(i)$ on $S_e(i)$. Intuitively, if colour *i* has a broader spatial distribution and more balanced distribution among all the regions (larger $E(i)$) and smaller skin probability $P_{skin}(i)$, it will have a smaller saliency value $S_e(i)$ and thus is more likely to belong to the background. It is worth noting that the saliency value $S_e(i)$ of a pixel only depends on its colour, which means for pixels with same colour *i*, they have the equal saliency value. By computing every pixel's saliency value and normalizing the saliency values to [0,255] latter, the obtained map is referred to as saliency map M_e , as shown in Fig. 1(d).

C. The Third Step: A Region-level Hand Detection Method

Studies have shown that people will pay more attention to areas of high contrast with surrounding objects. In the computer vision, contrast is determined by the difference in the colour or brightness of the object and other regions within the same field of view.

At the third step, we utilize a region-level saliency detection method to compute saliency of each region by measuring the difference in skin probability between it and the other regions, which means the contrast in this method is determined by the difference in the skin probability of different regions. We refer to this kind of contrast as skin probability contrast. The pixel-level hand detection method in section II.B focuses on the distribution of different colours in the image while ignoring the spatial relationship between the image regions. This step aims to enhance the saliency of the most aggregate and skin-like regions, meanwhile weakening that of the others. For region $R_k, k \in 1,2\cdots N$, its corresponding saliency value $S_r(R_k)$ is defined in (5):

$$S_r(R_k) = \sum\nolimits_{R_k \neq R_i} [N_{R_i} D_c(R_k, R_j) \exp(D_s(R_k, R_j)/\sigma_s)] \tag{5}$$

$$D_c(R_k, R_j) = \sum\nolimits_{x=1}^{N_{R_k}} P_{skin}(c_{kx})/N_{R_k} - \sum\nolimits_{y=1}^{N_{R_j}} P_{skin}(c_{jy})/N_{R_j} \tag{6}$$

In (5), N_{R_i} is the number of pixels in region R_i , $D_c(R_k, R_j)$ is the skin probability distance between region R_i and R_k , which is computed in (6). This term enhances saliency of regions with more skin-like pixels. And c_{kx} in (6) is the colour of *x*-th pixel in R_k . Then we incorporate spatial information by introducing a spatial weighting term $\exp(D_s(R_k, R_j)/\sigma_s)$ in (5), where $D_s(R_k, R_j)$ is the spatial distance between region R_j and R_k . σ_s controls the strength of spatial distance weighting. By introducing this spatial weighting term, skin-like regions with a higher degree of aggregation will have a higher saliency value. In contrast, the regions with lesser skin-like pixels or isolated skin-like region will have smaller saliency values. By normalizing the saliency values to [0,255] later, the obtained saliency map is referred to as saliency map M_c , which is shown in Fig. 1(e).

D. The Fourth Step: A Bayesian Framework for Fusing Saliency Maps

Rathu *et al.* [17] proposed a statistical saliency model based on Bayesian inference. By using the sliding window on the image, the image is divided into two parts: inside the window and outside the window. The saliency value of each pixel is calculated by comparing features inside the window and features outside the window. However, when this method uses the Bayesian inference to calculate the final saliency, the prior probability is set to a fixed value. In fact, this strategy makes the method rougher, and makes the contribution of Bayesian inference less obvious. In response to this problem, Xie *et al.* proposed using corner detection to construct convex hulls firstly, thereby obtaining the approximate location of salient areas. And then it used Bayesian inference to calculate saliency value of each pixel [31]. However, the effectiveness of this method depends on the accuracy of the convex hull

detection. Only when the background is simple, can a good result be obtained. Once the background of image is too complex, too many corner points are detected, which affects the accuracy of the algorithm.

At the fourth step, saliency map M_e and M_c are fused to construct a confidence map for segmentation. Inspired by "coarse-to-fine" model, a coarse confidence map M_{coarse} is first generated from M_e and M_c, and based on this coarse confidence map, a Bayesian framework is then utilized to obtain the final fine confidence map M_{fine}.

For pixel v with coordinate (x,y) in M_{coarse}, its corresponding coarse confidence value $M_{coarse}(x,y)$ is computed according to (7):

$$M_{coarse}(x,y) = \alpha \cdot \sqrt{M_e(x,y) \cdot M_c(x,y)} \tag{7}$$

Regions closer to the center of image have a higher probability of belonging to foreground. In our research, we use a center-bias method in [25] to enhance the course confidence map M_{coarse}. The principle of the center-bias algorithm is that regions near the center of image than those far from the center are more worthy of attention. So the weighted item α in (7) is defined in (8):

$$\alpha = \exp\left(-\frac{D(p(x,y), p_{center})}{[0.5 \cdot \min(W,H)]^2} \right) \tag{8}$$

where $D(p(x,y), p_{center})$ indicates the Euclidean distance between the pixel v with coordinate (x, y) and the center of the image. The parameter W and H represent the width and height of the input image. By using (8), the saliency of regions near the center of image will be enhanced. Fig. 1(f). shows an example of M_{coarse}.

To refine the coarse map M_{coarse}, it undergoes the Otsu's segmentation method in [18] to generate a binary image. As shown in Fig. 1(g), the image is roughly divided into two parts: hand region R_{hand} and background region R_{BG}. Next, we adopt a Bayesian framework to get the fine confidence value of pixel $v(x,y)$:

$$p(hand \mid v) = \frac{p(hand)p(v \mid hand)}{p(hand)p(v \mid hand) + (1 - p(hand))p(v \mid BG)} \tag{9}$$

$$p(v \mid hand) = \prod_{f \in \{L,a,b\}} \frac{H_{hand}(f(v))}{N_{hand}} \tag{10}$$

$$p(v \mid BG) = \prod_{f \in \{L,a,b\}} \frac{H_{BG}(f(v))}{N_{BG}} \tag{11}$$

$$p(hand) = M_{coarse}(x,y) \tag{12}$$

According to (12), we regard the coarse confidence value $M_{coarse}(x,y)$ as prior probability $p(hand)$ of pixel $v(x,y)$ to belong to hand region. By computing (9), we obtain every pixel's posterior probability $p(hand \mid v)$, which is regarded as the final confidence value of pixel $v(x,y)$. $p(v \mid hand)$ and $p(v \mid BG)$ refer to the observation likelihood probability, which are computed in (10) and (11), where $H_{hand}(x)$ refers to colour histogram of R_{hand} in the CIELab colour space, and $H_{BG}(x)$ refers to that of R_{BG}. N_{hand} and N_{BG} represent the number of pixels in R_{hand} and R_{BG}.By combining (10) and (11) with (9) to get every pixel's confidence value $M_{fine}(x,y)$, we obtain the final confidence map M_{fine}, in which each confidence value represents the probability of a pixel to belong to hand region, as shown in Fig. 1(h). Then we segment the confidence map M_{fine} with a threshold and get the final binary mask, in which the highlighted region represents the hand region, as shown in Fig. 1(i). The hand region is segmented using the binary mask leading to the result shown in Fig. 1(j).

E. Algorithm of the Proposed Method

To sum up, we use the approach in section II.A to make a pre-segmentation, and then two saliency maps are obtained from section II.B and section II.C respectively. In section II.D, the final fine confidence map is determined by using a Bayesian framework. The results of each step in the proposed method are shown in Fig. 4. The procedure of our algorithm is given as follows:

Input: A colour image.

1) By using SLIC approach, the input image is segmented into N regions.

2) Compute the pixel-level saliency map M_e using (2), (3) and (4).

3) Compute the region-level saliency map M_c using (5) and (6).

4) According to (7) and (8), the coarse confidence map M_{coarse} are computed by integrating saliency map M_e and M_c.

5) Calculate observation likelihood probability using (10) and (11).

6) Regard value in M_{coarse} as prior probability, compute posterior probability of each pixel to belong to hand region using (9), (12), and obtain the final confidence map M_{fine}.

7) Segment hand region using the final confidence map M_{fine}.

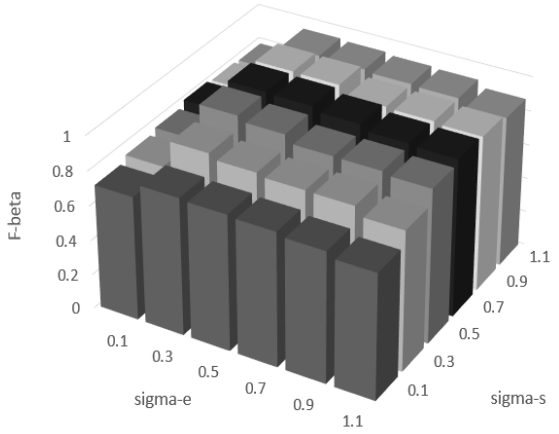
Fig. 5 is the flow diagram of the proposed segmentation method.

III. EXPERIMENTS

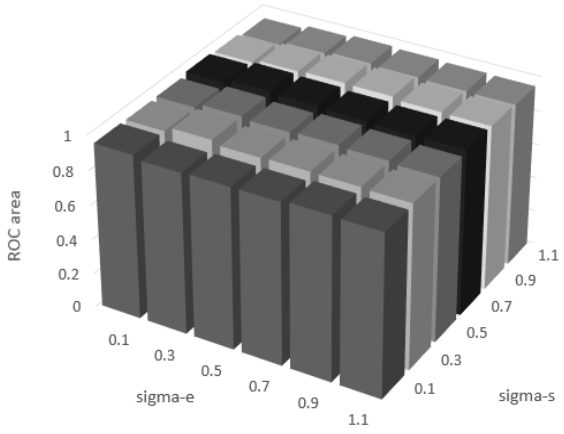
In this section, we make experiments on two datasets to verify the effectiveness of our hand posture segmentation method.

A. The Determination of Parameters and Methods

To make a comprehensive evaluation, we build a benchmark for hand segmentation. This evaluation is based on the Hand Gesture Recognition (HGR) dataset with uncontrolled backgrounds and lighting conditions [22]. The images from this dataset are all associated with ground-truth binary masks indicating hand posture regions. In this section, we use a series of evaluations to determine the parameters and threshold segmentation method in the proposed algorithm.



(a)



(b)

Fig. 6. Comparison for segmentation results using different parameters σ_e in (3) and σ_s in (5). (a) F_β bars. (b) ROC area bars.

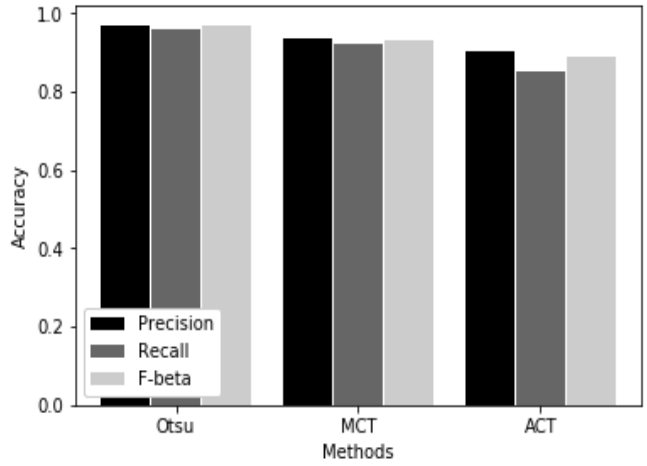
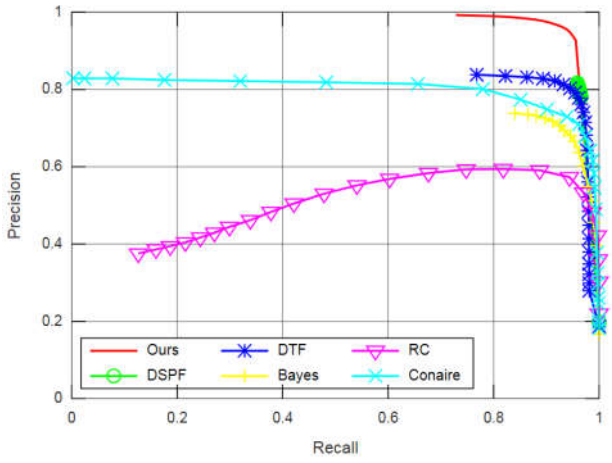
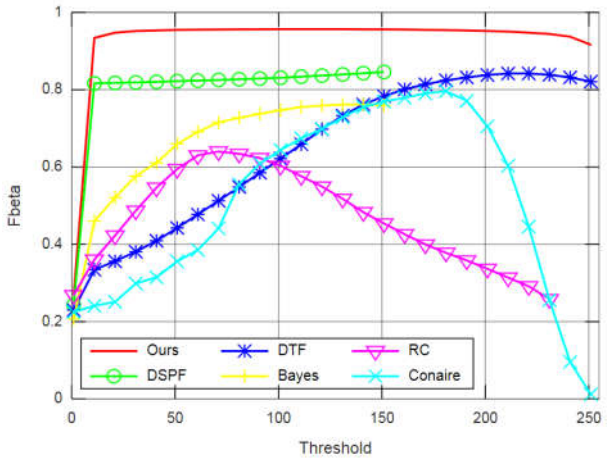


Fig. 7. Precision-Recall-F-beta bars for binary masks using different threshold-based segmentation methods.

Firstly, in order to determine the parameters and obtain a better segmentation result, we utilize images in the HGR dataset to test the effect of different parameter values on the experimental results. The parameter σ_e in (3) controls the strength of colour dispersion, and σ_s in (5) controls the strength of spatial distance weighting. Previous experiments showed that it's not necessary to set an overly big or small strength to control these values. Thus, the range of σ_e in (3) and σ_s in (5) are all limited to [0.1, 1.1]. With the stride of 0.2, these two parameters are used to build saliency maps and binary mask. The measurement utilized in [28] was employed



(a)



(b)

Fig. 8. Comparison for naive thresholding of saliency maps using images in HGR dataset. (a) Precision-recall curves. (b) Threshold- F_β curves.

to evaluate the performance. In this experiment, we use F_β and Receiver Operating Characteristic (ROC) area to evaluate the quality of the experiment results. The weighted harmonic mean of precision and recall, referred to as F_β, is computed in (13):

$$F_\beta = \frac{(1+\beta^2)Precision \times Recall}{\beta^2 \times Precision + Recall} \tag{13}$$

$$Precision = \frac{N_C}{N_R} \tag{14}$$

$$Recall = \frac{N_C}{N_G} \tag{15}$$

where the parameter β^2 is set to 0.3. The precision rate and recall rate are computed in (14) and (15), where N_C is the number of correct segmented pixels, N_R is the number of all segmented pixels, and N_G is the number of ground-truth pixels. As shown in Fig. 6, when σ_e is set to 0.3, and σ_s is set to 0.7, F_β and ROC area perform better. In fact, as long as these two parameters are not set to be particularly small, our method has similar performance. That is because, at the fourth step, the Bayesian framework makes a further refinement to the saliency map, thereby reducing the effect of these two

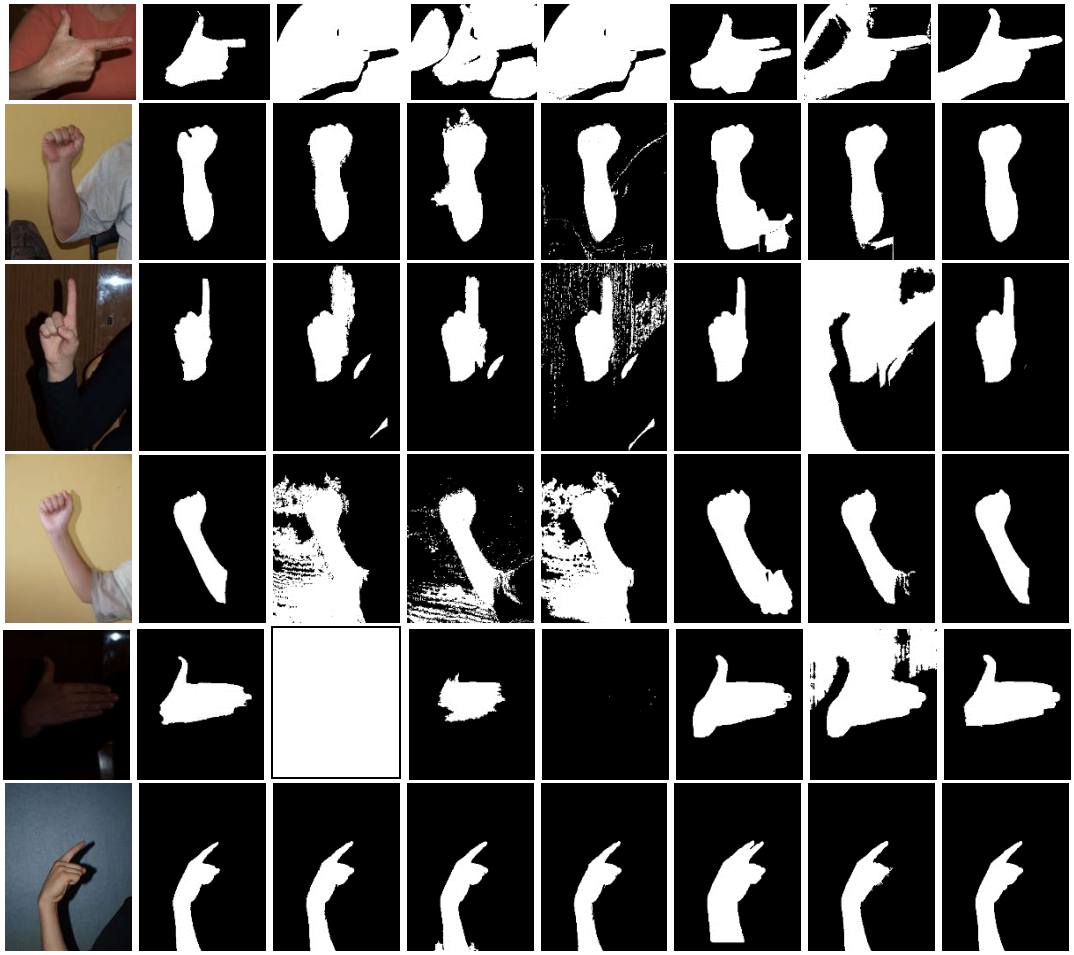
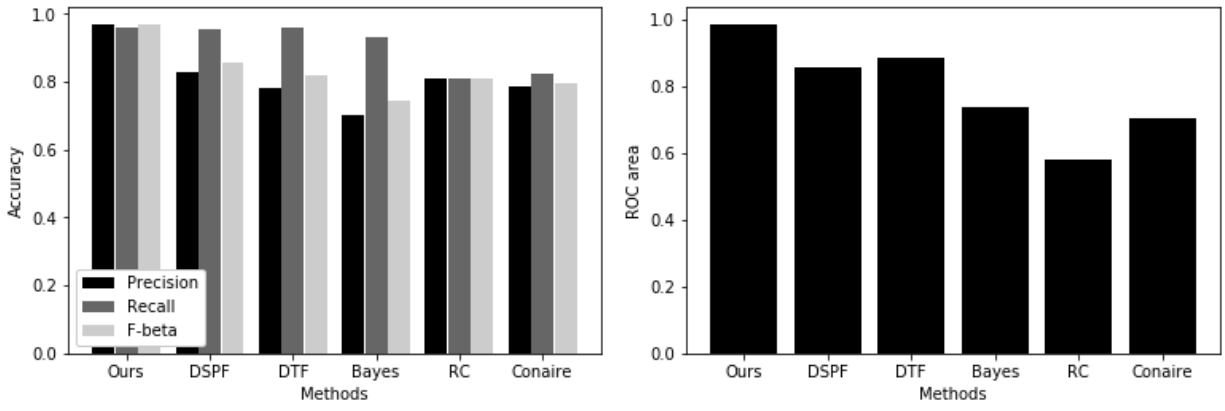


Fig. 9. Comparison for binary results using images in HGR dataset. (a) The input image. (b) Our method. (c) The DSPF method [22]. (d) The DTF method [7]. (e) The Bayes method [21]. (f) The RC method [10]. (g) The Conaire method [16]. (h) The ground truth.



(a) (b)

Fig. 10. Precision-Recall-F-beta bars and ROC area bars for binary masks after segmentation.

parameters on the experimental results. In the following experiments, the parameter σ_e and σ_s are always set to 0.3 and 0.7 respectively.

After the determination of parameters, we use a series of evaluations to determine the threshold segmentation method utilized in the fourth step. In the fourth step, after obtaining the final confidence map M_{fine}, in which each confidence value represents the probability of a pixel to belong to hand region, we need a threshold-based segmentation method to get the final binary mask. We utilize three typical methods for the evaluation, including the Otsu method [18], the minimum cross-entropy thresholding (MCT) method [26], and an adaptive threshold (AT) method [27]. As shown in Fig. 7, the Otsu method has a better result, so we utilize this method as the threshold segmentation method in the fourth step.

B. The Evaluation based on HGR Dataset

In this section, we compare the proposed method with five best performing and typical methods on this dataset, including methods based on discriminative textural features (DTF) [7], discriminative skin-presence features (DSPF) [22] and Bayesian skin model (Bayes) [21], the RC saliency detection with Saliency Cut method [10], and the Conaire's skin model method [16]. The performance of methods is assessed in two parts: the evaluation of confidence maps of different methods, and the evaluation of binary segmentation results of different

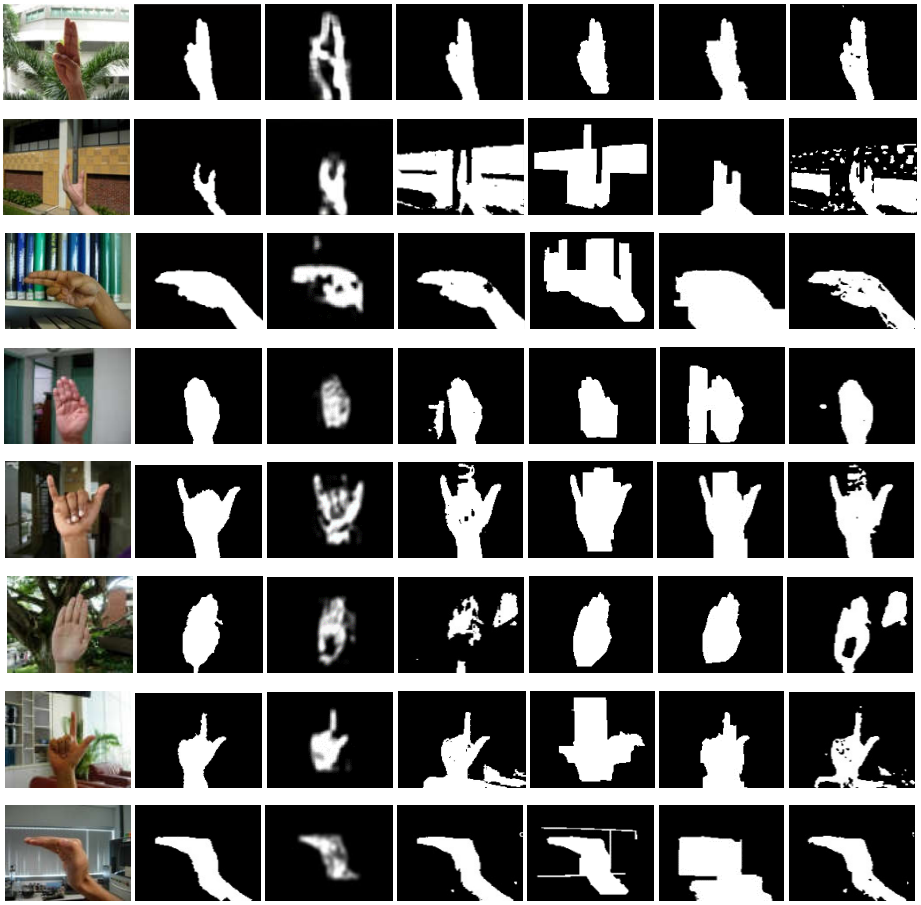


Fig. 11. Visual comparison of segmentation methods on NUS-II. (a) The input images. (b) Our method. (c) An attention based method in [19]. (d) An ellipse clustering colour model based method in [23]. (e) The Saliency Cut method in [10]. (f) The GrabCut method in [20]. (g) A Bayesian skin model based method in [21].

methods.

Firstly, we make an evaluation of confidence maps of different method. The easiest way to get the binary mask of the hand region is to set a fixed threshold and segment the confidence map with this threshold. To reliably compare how well various detection methods highlight hand posture regions in the confidence map, we adjust the segmentation threshold and compute recall, precision, and F_β at this condition.

The Precision-Recall curve in Fig. 8 shows that the proposed method is superior to the other five methods. With the growth of recall, the precision of our method drops more slowly. In this curve, the minimum recall value of the proposed method is less than that of DSPF, DTF, and Bayes method, and is bigger than that of RC and Conaire method, which means the smoothness and differentiation, and is conducive to the next segmentation.

Then we make an evaluation of binary segmentation results of different method. The comparison of binary results using images in HGR dataset is shown in Fig. 9. The DSPF, DTP, and Bayes method all use a consistent value of threshold to segment every image. Our method and Conaire method use the Otsu segmentation on the confidence maps. And the RC method uses an iterative fitting segmentation approach named Saliency Cut. After segmentation to get the binary results, the recall, precision, F_β and ROC area of each method are obtained. When computing F_β by (13), the parameter β^2 is also set to 0.3. The comparison of precision, recall and F_β values for binary masks after segmentation are shown in Fig. 9, Fig. 10 and Table I respectively. Among these six methods, the proposed method significantly outperforms other methods, especially in precision rate and F_β. Other colour-based methods and saliency-based methods are subject to poor performance due to uneven illumination and complex background interference, especially in the first and second image in Fig. 10.

In Fig. 11, DSPF, DTP, and Bayes method perform very well in terms of recall rate. However, these three methods require pre-training steps on a specific training set, which means that these methods will need more time in application. RC utilizes the global colour contrast information to detect saliency objects without using priori knowledge of colour, thereby performimg poor in terms of recall rate. Compared with the RC method, the Conaire method has higher recall rate and lower precision because of the usage of skin colour priori knowledge, which means that skin-like regions in the background might be misidentified as hand regions.

C. The Evaluation based on NUS-II Dataset

In this section, we make an evaluation on NUS Hand Posture dataset II (NUS-II) subset A [19]. All of the images in NUS-II are taken in complex environments, with a varied background, uneven lighting, and a wide variety of hand shapes. These hands in the dataset are from different age

TABLE I
PRECISION-RECALL-F-BETA AND ROC AREA VALUES FOR BINARY MASKS AFTER SEGMENTATION

Method	Precision	Recall	F-beta	ROC area
Ours	**0.9701**	**0.9613**	**0.9681**	**0.9852**
DSPF[22]	0.8297	0.9531	0.8552	0.8546
DTF[7]	0.7816	0.9596	0.8166	0.8823
Bayes[21]	0.6998	0.9297	0.7422	0.7341
RC[10]	0.8085	0.8097	0.8087	0.5798
Conaire[16]	0.7858	0.8228	0.7940	0.7034

TABLE II
THE RECOGNITION ACCURACY OF HAND POSTURE IN NUS-II

Method	Accuracy(%)
Ours	**96.53**
ECCM[23]	71.25
RC[10]	87.68
Grabcut[20]	85.31
Bayes[21]	79.46

TABLE III
THE AVERAGE TIME OF HAND SEGMENTATION IN HGR AND NUS-II

Method	Time In HGR(ms)	Time In NUS-II(ms)	Code
Ours	352	47	C++
DSPF[22]	361	51	C++
DTF[7]	328	40	C++
Bayes[21]	45	13	C++
RC[10]	106	19	C++
Conaire[16]	**17**	**6**	Matlab

groups, different races, and different sexes. In this experiment, we utilize typical images in NUS-II(indoor or outdoor, skin-like interference or not, different handshapes) for evaluating the performance of our method. As a comparison, we adopt five different methods for hand posture segmentation. They are an attention-based method [19] with pre-training on NUS-II, a method based on ellipse clustering colour model (ECCM) [23], Saliency Cut, a salient region (RC) segmentation method based on global contrast [10], Grabcut, a foreground extraction method using iterated Graph-Cuts [20], and a method based on Bayesian skin model [21]. It should be noted that the method in [19] uses the saliency maps for hand-position detection rather than segmentation, so we don't make a re-processing to the saliency map, but directly show the maps obtained.

In order to obtain the optimal experimental results, the parameter σ_e in (3) and σ_s in (5) are determined by the experiment in section III.B, in which σ_e is determined to be 0.3, and σ_s is determined to be 0.7. The threshold of segmentation to final confidence map in our method is determined by the Otsu method [18].

The segmentation results of these methods are shown in Fig. 11. In column (c), the method in [19] can detect the hand position approximately, but its edges aren't accurate enough, which is an important information for classification. In column (d), the method in [23] performs well only when the backgrounds don't contain skin-like regions and the illumination is even. Column (e) shows that only when the image colour contrast is strong, the effect of Saliency Cut method [10] will be more ideal. In column (f), when the true hand posture regions and background regions distribute overlap partially in colour space, the segmentation result of the GrabCut method [20] will be worse. In column (g), the Bayesian skin model [21] is also disturbed by the skin-like backgrounds. And column (b) shows that the proposed method has a better hand segmentation ability, irrespective of whether the image contains uneven illumination and skin-coloured background or not.

In view of the NUS-II dataset lacks ground-truth masks, but each of them has a classification label, therefore, we use the images after segmentation to make a classification and evaluate the performance of different segmentation algorithms. Image moments are useful to describe objects after segmentation, especially the Hu moment, invariants with respect to translation, scale, and rotation [29]. Thus, we adopt it to extract features of hand contours, which are then fed into a linear SVM for classification [30]. In this case, the recognition accuracies are much more dependent on the segmentation steps. The performance for hand posture recognition was evaluated using 10-fold cross-validation. As shown in Table II, the proposed method achieves the hand posture recognition rate to 96.53%, meanwhile the others achieve the accuracy of 71.25% (ECCM), 87.68% (RC), 85.31% (Grabcut), and 79.46% (Bayes). The results further demonstrate the practical value of this algorithm

D. The Evaluation of Speed

Finally, we compare the speed of each method, the average time of different method in HGR and NUS-II dataset is obtained on a computer equipped with an Intel Core i5-4460 3.2GHz CPU and 8GB RAM. In view of that the second and third steps of our algorithm have many shared parameters and data, we optimize the calculation process and implement it based on C++. As shown in Table III, the method in [21] and [16] have good running speed. The proposed method, DTF and DSPF have more similar speed performance which is generally acceptable. However, in order to achieve real-time performance, the running speed of this algorithm still has some room for improvement.

IV. CONCLUSION

In this study, a new hand posture segmentation method which deeply integrates the bottom-up saliency information with top-down skin colour information is proposed. By utilizing a saliency-based pixel-level and a saliency-based region-level hand detection algorithm, and then fusing saliency maps they output by a Bayesian framework, we obtain the final confidence map for segmentation. Evaluations based on comparisons with other approaches show that this method can resist the external interferences such as variable illumination and cluttered background, thereby obtaining the hand region accurately. Especially in the classification experiment, the results further demonstrate the practical value of this algorithm. In the future, instead of segmentation with a hard threshold, we are considering using an iterative fitting method to obtain a smoother and more precise segmentation result from the final confidence map. In addition, improving

the speed of the program to make it more suitable for real-time applications is our another target.

REFERENCES

[1] H. I. Lin, M. H. Hsu, and W. K. Chen, "Human hand gesture recognition using a convolution neural network," IEEE International Conference on Automation Science and Engineering, August 2014, pp.1038-1043.

[2] Q. Zheng *et al*., "Improvement of generalization ability of deep CNN via implicit regularization in two-stage training process," IEEE Access, vol. 6, no. 1, pp. 15844-15869, 2018.

[3] Y. S. Huang and Y. J. Wang, "A hierarchical temporal memory based hand posture recognition method," IAENG International Journal of Computer Science, vol. 40, no. 2, pp.87-93, 2013.

[4] Z. Jiang, M. Yao, and W. Jiang, "Skin detection using color, texture and space information," International Conference on Fuzzy Systems and Knowledge Discovery, August 2007, pp.366-370.

[5] J. Brand and J. S. Mason, "A Comparative Assessment of Three Approaches to Pixel-Level Human Skin-Detection," International Conference on Pattern Recognition, September 2000, pp. 5056.

[6] S. E. Ghobadi, O. E. Loepprich, F. Ahmadov, J. Bernshausen, K. Hartmann, and O. Loffeld, "Real time hand based robot control using multimodal images," IAENG International Journal of Computer Science, vol. 35, no. 4, pp.500-505, 2008.

[7] M. Kawulok, J. Kawulok, and B. Smolka, "Discriminative textural features for image and video colorization," Ieice Transactions on Information & Systems, vol. 95, no. D, pp.1722-1730, 2012.

[8] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," IEEE Transactions on Pattern Analysis & Machine Intelligence, vol. 20, no. 11, pp.1254-1259, 1998.

[9] R. Achanta, F. Estrada, P. Wils, and S. Süsstrunk, "Salient region detection and segmentation," Lecture Notes in Computer Science, vol. 5008, pp.66-75, 2008.

[10] M. M. Cheng *et al*., "Global contrast based salient region detection," Computer Vision and Pattern Recognition, June 2011, pp.409-416.

[11] R. Valenti, N. Sebe, and T. Gevers, "Image saliency by isocentric curvedness and color," IEEE International Conference on Computer Vision, September 2009, pp.2185-2192.

[12] Y. L. Chuang, L. Chen, and G. Chen, "Saliency-guided improvement for hand posture detection and recognition," Neurocomputing, vol. 133, no. 8, pp.404-415, 2014.

[13] K. Deshmukh, G. Shinde, "Adaptive Color Image Segmentation Using Fuzzy Min-Max Clustering," Engineering Letters, vol. 13, no. 2, pp.57-64, 2006.

[14] W. J. Yang, L. Kong, and M. Wang, "Hand gesture recognition using saliency and histogram intersection kernel based sparse representation," Multimedia Tools and Applications, vol. 75, no. 10, pp.1-14, 2016.

[15] R. Achanta *et al.*, "Slic superpixels compared to state-of-the-art superpixel methods," IEEE Transactions on Pattern Analysis & Machine Intelligence, vol. 34, no. 11, pp. 2274-2282, 2012.

[16] C. O. Conaire, N. E. O'Connor, and A. F. Smeaton, "Detector adaptation by maximising agreement between independent data sources," Computer Vision and Pattern Recognition, June 2007, pp.1-6.

[17] E. Rahtu *et al*., "Segmenting Salient Objects from Images and Videos," European Conference on Computer Vision, September 2010, pp. 366-379.

[18] N. Ohtsu, "A threshold selection method from gray-level histograms," IEEE Transactions on Systems Man & Cybernetics, vol. 9, no. 1, pp. 62-66, 1979.

[19] P. K. Pisharady, P. Vadakkepat, and P. L. Ai, "Attention based detection and recognition of hand postures against complex backgrounds," International Journal of Computer Vision, vol. 101, no. 3, pp. 403-419, 2013.

[20] C. Rother, V. Kolmogorov, and A. Blake, "GrabCut: interactive foreground extraction using iterated graph cuts," ACM Transactions on Graphics, vol. 23, no. 3, pp. 309-314, 2004.

[21] M. J. Jones and J. M. Rehg, "Statistical color models with application to skin detection," International Journal of Computer Vision, vol. 46, no. 1, pp. 81–96, 2003.

[22] M. Kawulok, J. Kawulok, and J. Nalepa, "Spatial-based skin detection using discriminative skin-presence features," Pattern Recognition Letters, vol. 41, no. 1, pp. 3-13, 2014.

[23] H. Tang and Z. Feng, "Hand's skin detection based on ellipse clustering," International Symposium on Computer Science and Computational Technology, December 2008, pp. 758-761.

[24] C. E. Shannon, "A mathematical theory of communication," Bell System Technical Journal, vol. 5, no. 1, pp. 3-55, 1948.

[25] X. Shen and Y. Wu, "A unified approach to salient object detection via low rank matrix recovery," IEEE Conference on Computer Vision and Pattern Recognition, June 2012, pp. 853-860.

[26] C. H. Li and C. K. Lee, "Minimum cross entropy thresholding," Pattern Recognition, vol. 26, no. 4, pp. 617-625, 1993.

[27] M. He, J. Cheng, and W. Feng, "Real-time albedo contrast-based hand segmentation in projector-camera system," IEEE International Conference on Robotics and Biomimetics, December 2015, pp. 937-942.

[28] R. Achanta *et al*., "Frequency-tuned salient region detection," 2009 Computer Vision and Pattern Recognition, June 2009, pp. 1597-1604.

[29] M. Hu, "Visual pattern recognition by moment invariants," IRE Transactions on Information Theory, vol. 8, no. 2, pp. 179-187, 1962.

[30] R. Fan *et al*., "LIBLINEAR: A Library for Large Linear Classification," Journal of Machine Learning Research, vol. 9, no. 9, pp. 1871-1874, 2008.

[31] Y. Xie and H. Lu, "Visual saliency detection based on Bayesian model," IEEE International Conference on Image Processing IEEE, September 2011, pp. 645-648.

Qingrui Zhang was born in Jinan, Shandong, China in 1993. He received his B.S. degree from Ocean University of China in 2015 and begin work for a M.S. degree in Shandong University in 2015. His research direction is image processing and deep learning.