Pattern Recognition 49 (2016) 102–114

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

A novel finger and hand pose estimation technique for real-time hand gesture recognition

Yimin Zhou [a,*,1], Guolai Jiang [a,b,1], Yaorong Lin [b]

[a] Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, China
[b] School of Electronic and Information Engineering, South China University of Technology, China

ARTICLE INFO

Article history:
Received 17 March 2014
Received in revised form
8 August 2014
Accepted 29 July 2015
Available online 8 August 2015

Keywords:
Computer vision
Finger modelling
Salient hand edge
Convolution operator
Real-time hand gesture recognition

ABSTRACT

This paper presents a high-level hand feature extraction method for real-time gesture recognition. Firstly, the fingers are modelled as cylindrical objects due to their parallel edge feature. Then a novel algorithm is proposed to directly extract fingers from salient hand edges. Considering the hand geometrical characteristics, the hand posture is segmented and described based on the finger positions, palm center location and wrist position. A weighted radial projection algorithm with the origin at the wrist position is applied to localize each finger. The developed system can not only extract extensional fingers but also flexional fingers with high accuracy. Furthermore, hand rotation and finger angle variation have no effect on the algorithm performance. The orientation of the gesture can be calculated without the aid of arm direction and it would not be disturbed by the bare arm area. Experiments have been performed to demonstrate that the proposed method can directly extract high-level hand feature and estimate hand poses in real-time.

1. Introduction

Hand gesture recognition based on computer vision technology has been received great interests recently, due to its natural human-computer interaction characteristics. Hand gestures are generally composed of different hand postures and their motions. However, human hand is an articulated object with over 20 degrees of freedom (DOF) [12], and many self-occlusions would occur in its projection results. Moreover, hand motion is often too fast and complicated compared with current computer image processing speed. Therefore, real-time hand posture estimation is still a challenging research topic with multi-disciplinary work including pattern recognition, image processing, computer vision, artificial intelligence and machine learning.

In human–machine interaction history, keyboard input & character text output and mouse input & graphic window display are main traditional interaction forms. With the development of computer techniques, the human–machine interaction via hand posture plays an important role under three dimensional virtual environment. Many methods have been developed for hand pose recognition [3,4,10,18,24,29].

A general framework for visual based hand gesture recognition is illustrated in Fig. 1. Firstly, the hand is located and segmented from the input image, which can be achieved via skin-color based segmentation methods [27,31] or direct object recognition algorithms. The second step is to extract useful feature for static hand posture and motion identification. Then the gesture can be identified via feature matching. Finally, different human machine interaction can be applied based on the successful hand gesture recognition.

There are a lot of constraints and difficulties in accurate hand gesture recognition from images since human hand is an object with complex and versatile shapes [25]. Firstly, different from less remarkable metamorphosis objects such as human face, human hand possesses over 20 free degree plus variations in hand gesture location and rotation which make hand posture estimation extremely difficult. Evidence shows that at least 6-dimension information is required for basic hand gesture estimation. The occlusion also could increase the difficulty in pose recognition. Since the involved hand gesture images are usually two dimensioned images, it would result in occlusion of some key parts of the hand on the plane project due to various heights of the hand shapes.

Besides, the impact of the complex environment to the broadly applied visual-based hand gesture recognition techniques has to be considered. The lightness variation and complex background such factors make it more difficult for the hand gesture segmentation. Up to now, there is no united definition for dynamic hand

* Corresponding author.
E-mail addresses: ym.zhou@siat.ac.cn (Y. Zhou), gl.jiang@siat.ac.cn (G. Jiang).
[1] The first author and second author contribute equally in the paper.

http://dx.doi.org/10.1016/j.patcog.2015.07.014
0031-3203/

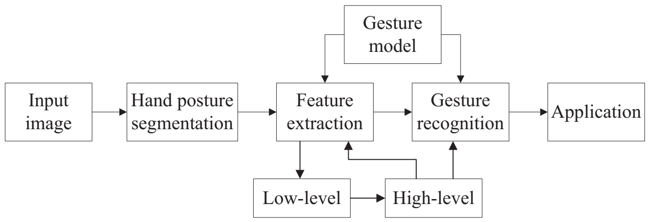


Fig. 1. The general framework of computer based hand posture recognition.

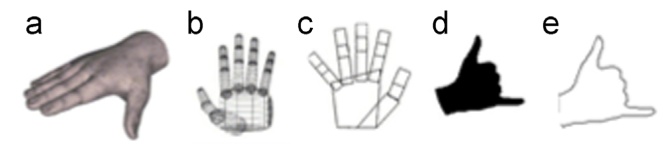


Fig. 2. Hand gesture models with different complexities (a) 3D strip model; (b) 3D surface model; (c) paper model [36]; (d) gesture silhouette; and (e) gesture contour.

gesture recognition, which is also an unsolved problem to accommodate human habits and facilitate computer recognition. It should be noted that human hand has deformable shape in front of a camera due to its own characteristics. The extraction of a hand image has to be executed in real-time independent of the users and device. Human motion possesses a fast speed up to 5 m/s for translation and 300 °C/s for rotation. The sampling frequency of a digital camera is about 30–60 Hz, which could result in fuzzification on the collected images with negative impact on further identification. On the other hand, with the hand gesture module added in the system, the dealt frame number per second for the computer will be even less, which will bring more serious pressure on the relatively lower sampling speed. Moreover, a large amount of data have to be dealt in computer visual system, especially for high complex versatile objects. Under current computer hardware conditions, a lot of high-precision recognition algorithms are difficult to be operated in real-time.

Our developed algorithm focuses on single camera based real-time hand gesture recognition. Some assumptions are made without loss of generality: (a) the background is not too complex without large area skin color disturbance; (b) lightness should avoid too low or too light such worse conditions; (c) the palm is right faced to the camera with distance in the range ≤ 0.5 m. These three limitations are not difficult to be realized in the actual application scenarios.

Firstly, a new finger detection algorithm is proposed. Compared to previous finger detection algorithms, the developed algorithm is independent of the finger tip feature but can extract fingers directly from the main edge of the whole fingers. Considering that each finger has two main “parallel” edges, a finger is determined from convolution result of salient hand edge image with a specific operator G. The algorithm can not only extract extensional fingers but also flexional fingers with high accuracy, which is the basis for complete hand pose high-level feature extraction. After the finger central area has been obtained, the center, orientation of the hand gesture can be calculated. During the procedure, a novel high-level gesture feature extraction algorithm is developed. Through weighted radius projection algorithm, the gesture feature sequence can be extracted and the fingers can also be localized from the local maxima of angular projection, thus the gesture can be estimated directly in real-time.

The remainder of the paper is organized as follows. Section 2 describes hand gesture recognition procedure and generally used methods. Finger extraction algorithm based on parallel edge characteristics is introduced in Section 3. Salient hand image can also be achieved. The specific operator G and threshold is explained in detail in Section 4. High-level hand feature extraction through convolution is demonstrated in Section 5. Experiments in different scenarios are performed to prove the effectiveness of the proposed algorithm in Section 6. Conclusions and future works are given in Section 7.

2. Methods of hand gesture recognition based on computer vision

2.1. Hand modelling

Hand posture modelling plays a key role in the whole hand gesture recognition system. The selection of the hand model is dependent on the actual application environments. The hand model can be categorized as gesture appearance modelling and 3D modelling. Generally used hand gesture models are demonstrated in Fig. 2.

3D hand gesture model considers the geometrical structure with histogram or hyperquadric surface to approximate finger joints and palm. The model parameters can be estimated from single image or several images. However, the 3D model based gesture modelling has quite a high calculation complexity, and too many linearization and approximation would cause unreliable parameter estimation. As for appearance based gesture models, they are built through appearance characteristics, which have the advantages of less computation load and fast processing speed. The adoption of the silhouette, contour model and paper model can only reflect partial hand gesture characteristics. In this paper, based on the simplified paper gesture model [36], a new gesture model is proposed where each finger is represented by extension and flexion states considering gesture completeness and real-time recognition requirements.

Many hand pose recognition methods use skin color-based detection and take geometrical features for hand modelling. Hand pose estimation from 2D to 3D using multi-viewpoint silhouette images is described in [35]. In recent years, 3D sensors, such as binocular cameras, Kinect and leap motion, have been applied for hand gesture recognition with good performance [5]. However, hand gesture recognition has quite a limitation, since 3D sensors are not always available in many systems, i.e., Google Glasses.

2.2. Description of hand gesture feature

The feature extraction and matching is the most important component in vision-based hand posture recognition system. In early stage of the hand gesture recognition, colored glove or labeling methods are usually chosen to strengthen the feature in different parts of the hand for extraction and recognition. Mechanical gloves can be used to capture hand motion, however, they are rigid with only certain free movements and relatively expensive cost [23]. Compared with the hand recognition methods with additional assistance of data glove or other devices, computer-vision based hand gesture recognition will need less or no additional device, which is more adaptable and has bright application prospect. A real-time algorithm to track and recognize hand gesture is described for video game in [23]. Only four gestures can be recognized, which has no generality. Here, the hand gesture images without any markers are discussed for feature extraction.

The generally used image feature for hand gesture recognition can be divided into two categories, low-level and high-level, as shown in Fig. 3. The low-level features such as edge, edge orientation, histogram of oriented gradients (HOG) contour/silhouette and Haar feature, are basic computer image characteristics and can be extracted conveniently. However, in actual

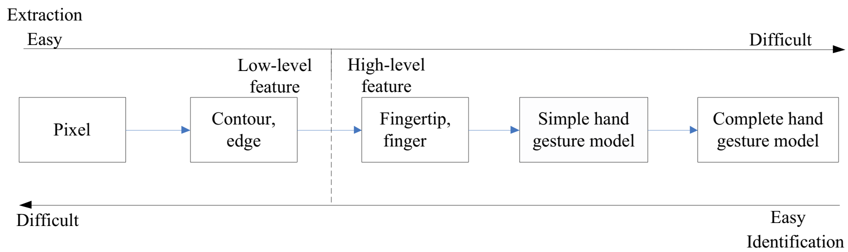


Fig. 3. The generally used feature and classification for hand gesture recognition.

applications, due to the diversities of hand motions, even for the same gesture, the subtle variation in finger angle can result in large difference in the image. With rotational changes in hand gesture, it is much more difficult to recognize gestures with direct adoption of low-level feature matching.

Since the skin color is a distinctive cue of hands which is invariant to scale and rotation, it is regarded as one of the key features. Skin color segmentation is widely used for hand localization [16,31]. Skin detection are normally achieved by Bayesian decision theory, Bayesian classifier model and training images [22]. Edge is another common feature for model-based matching [36]. Histogram of oriented gradients has been implemented in [30]. Combinations of multiple features can improve the accuracy and robustness of the algorithm [8].

Fingertip position, finger location and gesture orientation such high-level gesture features are related to the hand structure, which has direct relationship to the hand recognition. Therefore, they can be easily matched for various gesture recognition in real-time. However, this type of features is generally difficult to be extracted accurately.

In [1], fingertips were located from probabilistic models. The detected edge segments of monochrome images are computed by Hough transform for fingertip detection. But the light and brightness would seriously affect the quality of the dealt images and detection result. Fingertip detection and finger type determination are studied with a model-based method in [5], which is only applicable for static hand pose recognition. In [26], fingertips are found by fingertip masks considering their characteristics, and they can be located via feature matching. However, objects which share similar fingertip shapes could result in a misjudgment.

Hand postures can be recognized through the geometric features and external shapes of the palm and fingers [4]. It proposes a prediction model for showing the hand postures. The measurement error would be large, however, because of the complexity of the hand gestures and diversified hand motions. In [3], palm and fingers were detected by skin-colored blob and ridge features. In [11], a finger detection method using grayscale morphology and blob analysis is described, which can be used for flexional finger detection. In [9,13], high-level hand features were extracted by analyzing hand contour.

2.3. Methods of hand gesture segmentation

Fast and accurate hand segmentation from image sequences is the fundamental for gesture recognition, which have direct impact on the followed gesture tracking, feature extraction and final recognition performance. Many geometrical characteristics can be used to detect hand existence in image sequences via projection, such as contour, fingertip and finger orientation [26]. Other non-geometrical features, i.e., color [2], strip [34] and motion can also be used for hand detection. Due to the complex background, unpredictable environmental factors and diversified hand shapes, hand gesture segmentation is still an open issue.

Typical methods for hand segmentation are summarized as follows. Increasing constraints and building hand gesture shape database are usually used for segmentation. Black or white wall and dark color cloth can be applied to simplify the backgrounds. Besides, particular colored gloves can be worn to divide the hand and background through emphasized front view. Although these kinds of methods have good performance, it adds more limitation at the cost of freedom. A database can be built to collect hand sample images at any moment with different positions and ratios for hand segmentation through matching. It is a time consuming process, though, the completeness of the database can never be achieved which has to be updated all the time.

Methods of contour tracking include snake-model based segmentation [17], which can track the deformation and non-rigid movement effectively, so as to segment the hand gesture from the backgrounds. Differential method [20] and its improved algorithm can realize the segmentation by the deduction from the object images to the background images. It has a fatal defect that the camera has to be fixed and the background should be kept invariant during background and hand image extraction.

Skin color, assumed as one of the most remarkable surface features of human body, is often applied in gesture segmentation [31]. However, only adoption of this feature would be easily affected by the ambient environmental variations, especially when a large area of skin color disturbance is in the background, i.e., hand gesture overlapped by human face. Motion is another remarkable and easily extracted feature in gesture images. The combination of these two features becomes more and more popular in recent years [15].

Depth information (the distance information between the object and the camera) in the object images can also be used for background elimination and segmentation since human hands are the closet objects to the camera. Currently, the normally used depth camera are Swiss Ranger 4000 from Mesa Imaging company, Cam cube 2.0 from PMD Technologies, Kinect from Microsoft and Depth camera from PrimeSense.

2.4. Methods of gesture recognition

2.4.1. Methods of static gesture recognition

Methods of static gesture recognition can be classified into several categories:

(1) *Edge feature based matching*: Gesture recognition based on this type of feature is realized through the calculated relationship between data sets of the features and samples to seek the best matching [22,28,33,36]. Although it is relatively simple for feature extraction and adaptable for complex background and lightness, the data based matching algorithm is quite complicated with heavy computational load and time cost. A large amount of templates should be prepared to identify different gestures.
(2) *Gesture silhouette based matching*: Gesture silhouette is normally denoted as binary images of the wrapped gesture from segmentation. In [20], matching is calculated through the size of the overlapped area between the template and silhouette.

Zernike matrix of the images is used to cope with the gesture rotation [14] and feature set is developed for matching. The disadvantages of this type of method are that not all gestures can be identified with only silhouette and accurate hand segmentation is required.

(3) *Harr-like feature based recognition*: Harr-like feature based Adaboost recognition algorithm has achieved good performance in face recognition [21]. This method can be used for hand detection [37] and simple gesture recognition [10]. Experiments demonstrate that the method can recognize specific gestures in real-time under complex background environments. However, Harr-like feature based algorithm has high requirement on the consistency for the dealt objects, whereas hand gesture has diversified shape variations. Currently, this type method can only be applied in predefined static gestures recognition.

(4) *External contour based recognition*: External contour is an important feature for gesture. Generally speaking, different gestures have different external contours. The curvature of the external contour is varied in different positions of a hand (i.e., curvature is large at fingertip). In [9], curvature is analyzed for CSS (Curvature Scale Space) feature extraction to recognize gesture. Fingertip, finger root and joint such high-level features can be extracted from contour analysis [13]. A feature sequence is constructed by the distances from the contour points to the center for gesture recognition [32]. This type of method is adaptable for the angle variation between fingers but also dependent on the performance of the segmentation.

(5) *Finger feature based recognition*: Finger is the most widely applied high-level feature in hand pose recognition since the location and states of the fingers embody the most intuitional characteristics of different gestures. Only the finger positions, finger states and hand center are located thus simple hand gestures can be determined directly.

Several fingertip recognition algorithms are compared in [6]. In [26], circular fingertip template is used to seek the fingertip location and motion tracking. Combined with skin color feature, Blob and Ridge features are used to recognize palm and fingers [3]. However, only extensional fingers can be recognized via this type of methods.

2.4.2. Methods of motion gesture recognition

Time domain models are normally adopted for motion gesture recognition, which includes HMM (Hidden Markov Model) and DTW (Dynamic Time Warping) based methods:

(1) *HMM-based method*: It has achieved good performance in voice recognition area and applied in gesture recognition as well. Different motion gesture sequences are modelled via HMM, and each gesture is related to a HMM process. HMM-based method can realize recognition through feature matching at each moment, whose training process is a dynamic programming (DP) process. This method can provide time scale invariance and keep the gestures in time sequence. However, the training process is time consuming and the selection of its topology structure is determined by the expert experience, i.e., trial and error method used for number of invisible states and transfer states determination.

(2) *DTW-based method*: It is widely used in simple tracking recognition through the difference between the dealt gestures and standard gestures for feature matching at each moment. HMM and DTW are essentially dynamic programming processes, and DTW is the simplified version of HMM. DTW-based recognition has limitation on the word database applications.

In summary, methods of hand modelling, gesture segmentation and feature extraction are discussed. Most used hand gesture recognition methods are also illustrated. The following sections will introduce the proposed algorithm for real-time hand gesture recognition in detail.

3. Finger extraction algorithm based on parallel edge feature

The most notable parts of a hand to differentiate other skin objects, i.e, human face, arm, are fingers. As it is known, finger feature extraction and matching have great significance in hand segmentation. Contour can be extracted from silhouette of a hand region as the commonly used feature for hand recognition. Due to nearly 30 degrees of freedom in hand motion, hand image extraction will be executed regarding a hand as a whole. Moreover, arm should be eliminated. It should be noted that occlusion among four fingers (except thumb) could frequently occur, especially for flexional fingers.

To solve these problems associated with hand image extraction, a model-based approach for finger extraction is developed in this paper. It can obviate the finger joint location in hand motion and extract finger features from the silhouette of the segmented hand region. In complex background circumstances, models with fixed threshold can result in false detection or detection failure. However, fixed threshold color model is still selected for segmentation in this paper because of its simplicity, low computational load and invariant properties with regard to the various hand shapes. The threshold is predefined to accommodate the general human hand sizes. Then the selected pixels are transformed from RGB-space to YCbCr-space for segmentation. Finger extraction is explained in detail.

3.1. Salient hand gesture edge extraction

3.1.1. Finger modelling

Combined with the skin, edge and external contour such easily extracted low-level features, a novel finger extraction algorithm is proposed based on the approximately parallel finger edge appearance. It can detect the states of the extensional and flexional fingers accurately, which can also be used for further high-level feature extraction. It is known that fingers are cylindrical objects with nearly constant diameter from root to tip. As for human hand motion, it is almost impossible to move the distal interphalangeal (DIP) joint without moving the adjacent proximal interphalangeal (PIP) joint with no external force assistance and vice versa. Therefore, there is almost a linear relationship between these two types of joints, where the finger description can be simplified accordingly.

Firstly, each finger is modelled by its edges, which is illustrated in Fig. 4(a). C_{fi}, the boundary of the ith finger, is the composition of arc edges (C_{ti}, fingertip or joints) and a pair of parallel edges ($C_{ei} \cup C'_{ei}$, finger body), described as

$$C_{fi} = (C_{ei} \cup C'_{ei}) \cup \sum_{j=1,2} C_{ti,j} \tag{1}$$

where the arc edge $C_{ti,j}$ denotes the finger either in extensional ($j=1$) state or flexional state ($j=2$) (see two green circles in Fig. 4(b)).

The finger center line (FCL), C_{FCLi}, is introduced as the center line of the parallel finger edges to represent the main finger body. The distance between finger edges is defined as $2d$, which is the averaged diameter of all the fingers. Fingertip/joint center O_{ti} is located at the end of FCL, and it is the center of the arc curve C_{ti} as well. The finger central area along with C_{FCLi} will be extracted for finger detection. Compared with many algorithms based on fingertip feature [26], the proposed method is more reliable which can also detect the flexional fingers successfully.

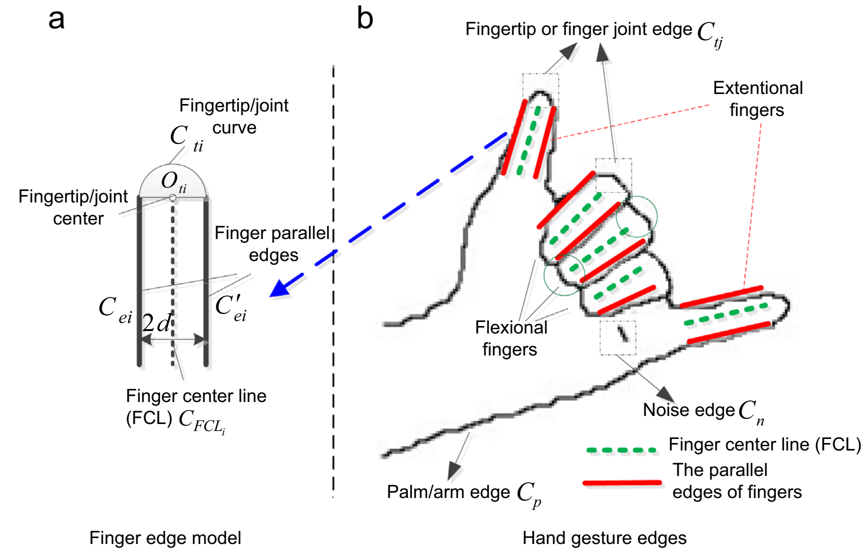


Fig. 4. The diagram of the finger edge model. (For interpretation of the references to color in this figure, the reader is referred to the web version of this paper.)

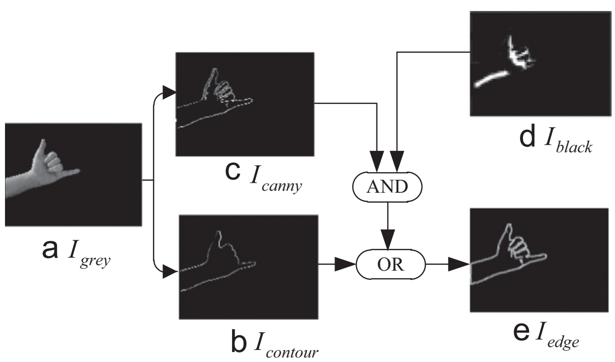


Fig. 5. The diagram of extracting hand salient edge procedure.

3.1.2. *Structure of hand gesture edge*

The remarkable hand edge C_{hand} of a gesture can provide concise and clear label for different hand gestures. Considering the hand structure characteristics and the assumed constraints, C_{hand} consists of the following curves:

$$C_{hand} = \sum_{i=1}^{5} C_{fi} + C_p + C_n \tag{2}$$

where C_p is the palm/arm edge curve and C_n is the noise edge curve. The diagram of the hand edge is shown in Fig. 4(b). Finger edges and palm/arm edges have direct relationship with the gesture structure, which are the main parts of the hand gesture edges and have to be detected as fully as possible. Edge curves formed by palmprint, skin color variation and skin rumple are noise, which has no connection with the gesture structure and should be eliminated completely.

3.1.3. *Extracting the salient hand gesture edge*

As for the hand gesture image input with complex background, skin color based segmentation algorithm is selected for initial segmentation. Morphology filter is then used to extract hand area mask and gesture contour $I_{contour}(x, y)$. The gray image $I_{gray}(x, y)$ of the gesture can be obtained at the same time, where arm area might be included. Canny edge detection algorithm [7] can extract most of the remarkable gesture edges. However, the detection results will also contain some noisy edges formed by the rumple and reflection from hand surface, which should be separated from the finger edges.

The salient hand edge image $I_{edge}(x, y)$ is mainly made up of hand contour which includes boundaries of extensional fingers, flexional fingers and arm/palm. Adduction (approximation) and abduction (separation) movements of the fingers can be referenced with finger III (middle finger), and it has slight movement without external force disturbance during motion. When the fingers are in extensional states, they are free to carry out adduction and abduction movements, whose edges are easily obtained. When the fingers are clenched into a fist or in 'six' number gesture such flexional states, as shown in Fig. 4, obvious ravine would be formed in the appressed part with lower gray values in the related pixels. Based on this characteristic, most noisy edges can be eliminated.

The procedure of extracting the salient hand edge is depicted in Fig. 5. One of the hand posture shown in Fig. 4(b) is used as an example, where its grayscale hand image can be seen in Fig. 5(a). The steps of $I_{edge}(x, y)$ extraction are summarized as follows:

1. Extract grayscale hand image $I_{gray}(x, y)$ (see Fig. 5(a)) and hand contour image $I_{contour}(x, y)$ (see Fig. 5(b)) from source color image using skin color segmentation method in [16].
2. Extract canny edge image $I_{canny}(x, y)$ (see Fig. 5(c)) from $I_{gray}(x, y)$ [7].
3. Applying the threshold Th_{black} (predefined) to grayscale hand image for extraction, then the obtained $I_{black}(x, y)$ (see Fig. 5(d)) is

$$I_{black}(x, y) = \begin{cases} 1, & (I_{gray}(x, y) < Th_{black}) \\ 0, & (I_{gray}(x, y) \geq Th_{black}) \end{cases} \tag{3}$$

   The boundaries of the flexional fingers are extracted from the overlapped area, i.e., $I_{canny}(x, y) \cap I_{black}(x, y)$.
4. Then the salient hand edge image $I_{edge}(x, y)$ is obtained:

$$I_{edge}(x, y) = I_{black}(x, y) \cap I_{canny}(x, y) \cup I_{contour}(x, y) \tag{4}$$

   The curve denoted by binary image $I_{edge}(x, y)$ shown in Fig. 5(e)) is the remarkable edge C_{hand}.

3.2. *Finger extraction via parallel edge feature*

Finger images can be extracted from the hand edge profile I_{edge} directly. In order to extract obtruding fingers from hand edge images, a rotation invariant operator G is introduced, which is

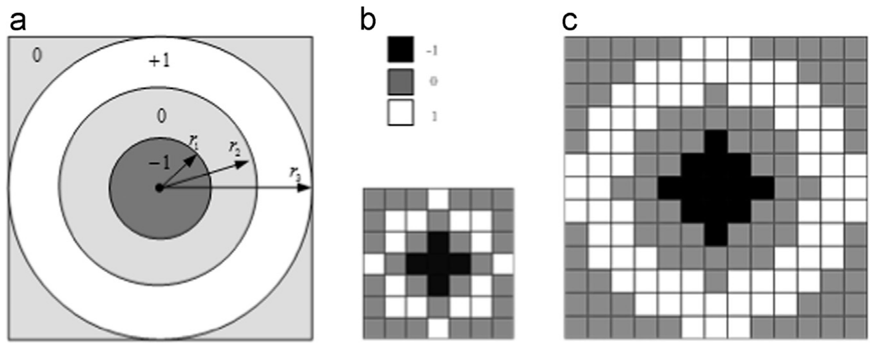


Fig. 6. (a) Continuous operator G; (b) discrete operator G ($N=1$); and (c) discrete operator G ($N=2$).

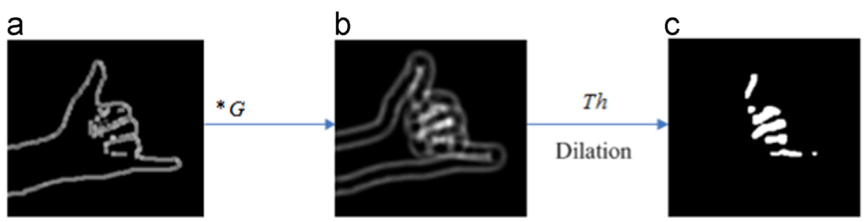


Fig. 7. Diagram of extracting fingers from hand edge image. (a) Salient hand edge image $I_{edge}(x,y)$, (b) convolution result $I_w(x,y)$, and (c) finger image $I_f(x,y)$.

given by

$$G(x,y)=\begin{cases}-1, & 0\le D(x,y)\le r_1\\ 1, & r_2<D(x,y)\le r_3\\ 0 & \text{else}\end{cases}\tag{5}$$

and

$$D(x,y)=\sqrt{(x-x_0)^2+(y-y_0)^2}\tag{6}$$

where (x_0,y_0) denotes the center coordinates of G; $D(x,y)$ is the distance of the points to the center on the operator; r_1,r_2 and $r_3(r_1<r_2<r_3)$ are defined as constant coefficients. Here we simply assume $r_1=r_2-r_1=r_3-r_2=N$, and $N=0.5d$, which is demonstrated in Fig. 6. Furthermore, the size of the continuous G operator is defined as $(6N\times 6N)$, while the discretized G is with $((6N+1)\times(6N+1))$, without loss of generality. The definition of G will be explained later.

The finger image $I_f(x,y)$ can be extracted by the following steps:

1. To define the size of operator G with $N=0.5d$. When the dealt hand image is too large or too small, it should be adjusted to proper size firstly.
2. To calculate the convolution $I_w(x,y)=I_{edge}(x,y)\times G$.
3. To extract rough finger image $I_{f'}(x,y)$ from $I_w(x,y)$,

   $$I_{f'(x,y)}=\begin{cases}1, & I_{w(x,y)}>T_h\\ 0, & I_{w(x,y)}\le T_h\end{cases}\tag{7}$$

   where T_h is the proper threshold in the range of $\left(\sqrt{5}d,2\sqrt{5}d\right)$. The determination of T_h will be explained in the next section.
4. To apply morphological dilation operation to $I_{f'}(x,y)$ to get the finger image $I_f(x,y)$.

The whole finger extraction procedure is demonstrated in Fig. 7. Through the convolution operation, the proposed finger feature extraction algorithm has robustness to the noise and can extract correct finger images.

Human palm size is approximately proportional to his/her phalanx length. When facing the camera, the palm size can be estimated and the finger length can be calculated consequently [25]. Hence the size of operator G can be adjusted accordingly. It should be noted that in the extracted finger images, noise pixels are mainly generated by edges of palmprint near palm center, while faulty detection is often caused by incomplete extraction of flexional finger boundaries.

In summary, a new finger detection algorithm is proposed. Different from previous finger detection algorithms, the developed one is independent of the fingertip characteristic and can extract the main edges of whole fingers directly. Besides, the algorithm can detect the extensional and flexional fingers accurately, which is the basis for high-level gesture feature extraction.

4. Determination of operator *G* and threshold T_h

In order to extract the central areas of fingers, the size of operator G and threshold T_h should be determined according to the averaged finger width $2d$.

Since G appears in a circular shape, it is an isotropic operator. It has relatively smaller size compared with the magnitude of the palm. If only palm/arm edge curve C_p and G are convoluted in local area, the curve C_p can be approximated as a straight line with convolution. Due to the isotropous characteristic of G, the simplified situation with line equation $x=0$ is considered. Hence, the convolution results are irrelevant to the vertical coordinate. $f(x,y|N)$, the convolution of line $x=0$ with continuous operator $G(6N\times 6N)$ is expressed as

$$f(x,y|N)=\begin{cases}2(\sqrt{9N^2-x^2}-\sqrt{4N^2-x^2}-\sqrt{N^2-x^2}),\\ |x|\in[0,N)\\ 2(\sqrt{9N^2-x^2}-\sqrt{4N^2-x^2}),\\ |x|\in[N,2N)\\ 2\sqrt{9N^2-x^2}, |x|\in[2N,3N)\\ 0,\quad |x|\in(3N,+\infty)\end{cases}\tag{8}$$

where $f(x,y|N)$ is an even function symmetrically about the y-axis with maximum value $2\sqrt{5N}$ at $|x|=2N$, depicted in Fig. 8.

The finger edges $(C_{ei}\cup C'_{ei})$, regarded as a pair of parallel lines with distance $2d$, are described by line equations $x=0$ and $x=2d$.

$F(x,y|N)$ is the result of the two independent convolution from paired finger edges:

$$F(x,y|N)=f(x,y|N)+f(x-2d,y|N)$$
$$\le f(2N,y|N)+f(-2N,y|N)=4\sqrt{5N} \qquad (9)$$

It can be shown when $x=2d-x=2N$, it means $N=0.5d$, and $F(x,y|N)$ will have maximum value $4\sqrt{5N}$ on $x=d$ (the finger center line (FCL)).

The average finger width is $8(=2d)$ without loss generality. Let take $G(N=2)$ as an example. When the applied finger width varies from 4 to 10 (different people), the finger central area can still be obtained with proper threshold T_h selection. Fig. 9 shows the convolution results of parallel lines with different widths $(2d=4,6,8,10)$ and 12×12 operator $G(N=2)$. Our proposed method can accommodate different sized fingers. Furthermore, it demonstrates that if the width of the parallel lines and operator G satisfies, $N=0.5d$, the corresponded convolution of the central line FCL have the maximum value, i.e., $N=2,\ d=4$.

Therefore, the maxima $4\sqrt{5N}$ can be used as the threshold to detect the central line in the parallel lines with $4N$ interval from convolution. Considering the noise disturbance, the adopted threshold should be a bit less than $4\sqrt{5N}$. To suppress the output from C_p and C_n edge curves in the convolution, the minimum value of the threshold should be larger than $2\sqrt{5N}$. Therefore, with the knowledge of averaged finger width $2d$, the operator G is determined by

$$N=0.5d \qquad (10)$$

and the threshold T_h should be

$$T_h\in(\sqrt{5d},2\sqrt{5d}) \qquad (11)$$

so that the finger central area related to parallel edges $(C_{ei}\cup C'_{ei})$ is detected.

Generally, the operator G and the threshold T_h are selected according to different finger widths. Table 1 lists the values in continuous situations.

Similar to continuous situations, the convolution results of line $x=0$ with discrete operator G $((6N+1)\times(6N+1))$ are as follows:

$$f(x,y|N=1)=[1,4,1,-1,1,4,1],\quad \text{where}\quad x=-3,-2,-1,0,1,2,3; \qquad (12a)$$

$$f(x,y|N=2)=[3,7,8,6,3,3,-1,3,3,6,8,7,3],\quad \text{where}\quad x=-6,-5,\ldots,0,\ldots,5,6; \qquad (12b)$$

$$f(x,y|N=3)=[7,11,13,10,8,6,5,1,-1,1,5,6,8,10,13,11,7],$$

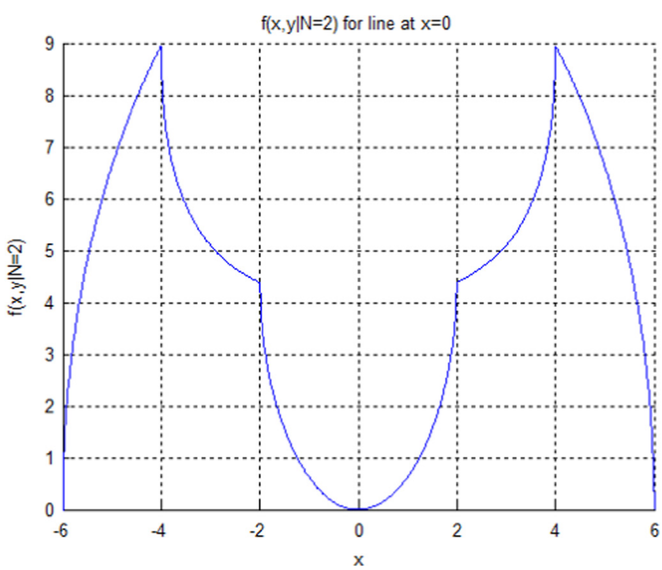


Fig. 8. Maximum $2\sqrt{5N}$ when $|x|=2N$.

Table 1
The usually used operator G and threshold T_h.

Finger width (pixels)	Average finger width ($2d$) (pixels)	N	Continuous operator G	Continuous threshold T_h
3–5	4	1	6×6	5
6–10	8	2	12×12	11
11–18	12	3	18×18	18

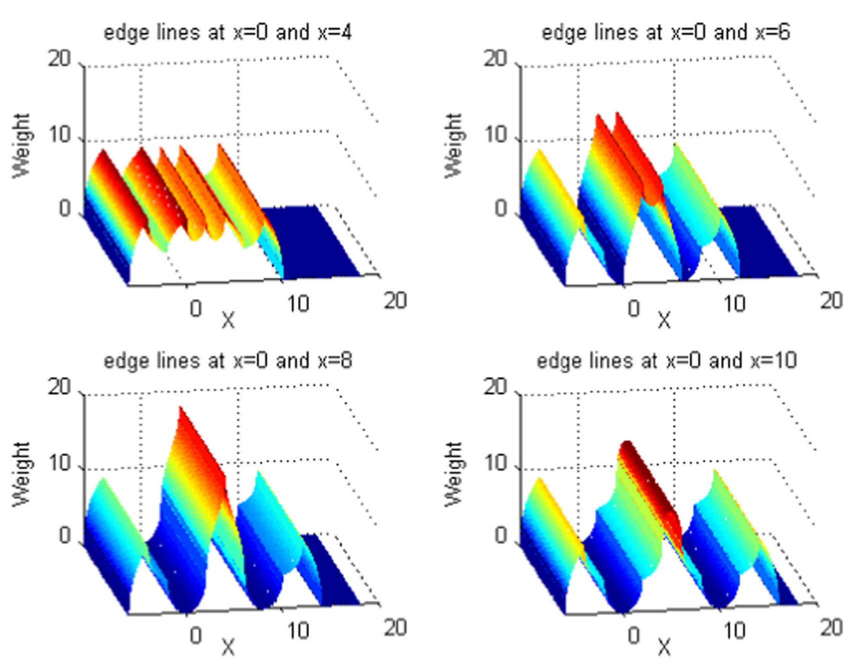


Fig. 9. Convolution for parallel edges with different distances $G(N=2)$.

Table 2
The usually used operator G and threshold T_h.

Finger width (pixels)	Average finger width ($2d$) (pixels)	N	Discrete operator G	Discrete threshold T_h
3–5	5	1	7×7	6
7–11	9	2	13×13	12
12–17	15	3	19×19	18

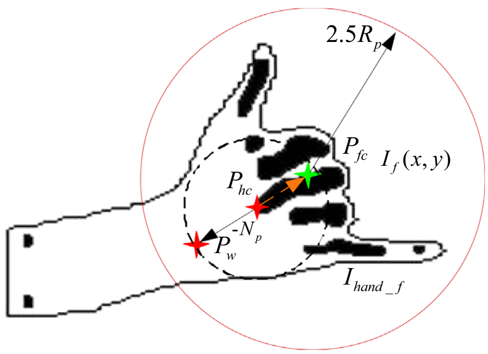


Fig. 10. The diagram of the parameter determination for the hand model.

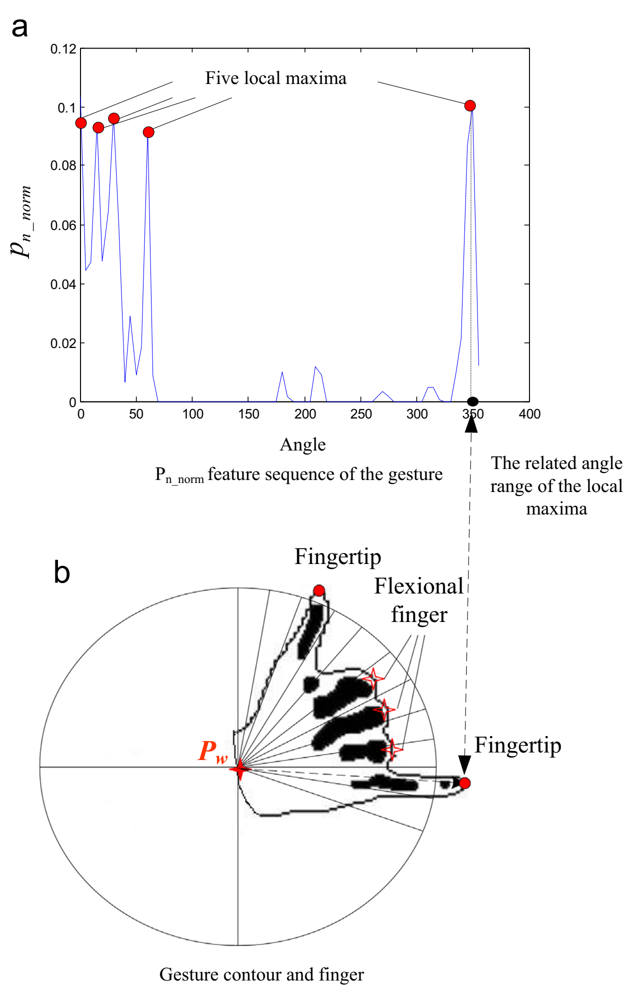


Fig. 11. The diagram of finger location algorithm.

$$\text{where} \quad x = -9, -8, \ldots, 0, \ldots, 8, 9. \tag{12c}$$

Table 2 shows the selected operators G and thresholds T_h values when operated in discrete condition.

It should be noted that the operator G and the threshold T_h are calculated in continuous condition. However, the discrete values will be used in the actual calculation and application.

5. High-level hand feature extraction based on weighted radial projection

Based on the paper model [36], all finger states can be simplified into extensional and flexional states. A new hand gesture model is thus developed with the parameters: hand center P_{hc}, orientation of the hand gesture $\overrightarrow{N}_p$, the status of the five fingers $\{Status\_i\}$ and fingertip position (extensional) or joint position (flexional) $\{P_{finger\_i}(x,y)\}$. On the basis of the finger extraction algorithm described in Section 3, high-level hand feature extraction can be achieved.

5.1. Locating hand center

Normally, the palm center is regarded as the gesture center since it is insensitive to varied hand gestures. It is defined as the point which has the maximum distance to the closest hand external boundary in the hand region. In [26], morphological erosion algorithm is adopted to remove the extensional fingers from the acquired hand area mask. Then the center of the remaining region is extracted as the hand gesture center, which is a relatively stable estimation. However, in this algorithm, the arm area should be eroded in the first place, since it would bring negative effect on the hand center determination due to the similar arm width and wrist width.

In the paper, the palm center/hand center P_{hc} can be located as follows. Without loss of generality, the average finger width and palm radius are assumed as $2d$ and R_p, respectively. Fig. 10 depicts the finger center and hand center determination. Based on the extracted finger central area $I_f(x,y)$, the mass center of this area is calculated, which is denoted as the finger center P_{fc}. A circle with a diameter $5R_p$ and center at P_{fc} is drawn (see big solid circle in Fig. 10) and used to segment the hand area from the arm area. Thus the hand gesture silhouette I_{hand_f} is obtained and the morphological erosion operation is applied afterwards. The mass center of the erosion remain is located, which is defined as the palm center P_{hc}.

5.2. Hand orientation determination

Hand orientation is another important feature for hand posture recognition. Generally speaking, the orientation of the index finger can be regarded as the orientation of the gesture. However, it is quite difficult to extract the index finger directly during hand motion, especially when it is in flexional state. Therefore, hand gesture orientation is normally obtained from mediated approaches. Some experts analyze the hand gesture contour and extract the ellipse long axis of the fitted external contour as the hand gesture orientation [9]. This method can only obtain a rough orientation, and it is not always applicable because the silhouettes of some postures, i.e., fist, do not have obviously principal axis. Arm orientation is assisted for hand gesture orientation determination [26], however, arm may not be detected in all circumstances.

Since fingers are always located on one side of the hand palm, the unit vector from the hand center P_{hc} to the finger center P_{fc} will give a more reliable estimation of the hand orientation $\overrightarrow{N}_p$, which is determined by

$$\overrightarrow{N_p} = \frac{\overrightarrow{P_{hc}P_{fc}}}{\left|\overrightarrow{P_{hc}P_{fc}}\right|} \tag{13}$$

where $\vec{\cdot}$ is the vector denotation, $|\cdot|$ is the norm operation. The wrist position P_w is determined through the palm radius R_p

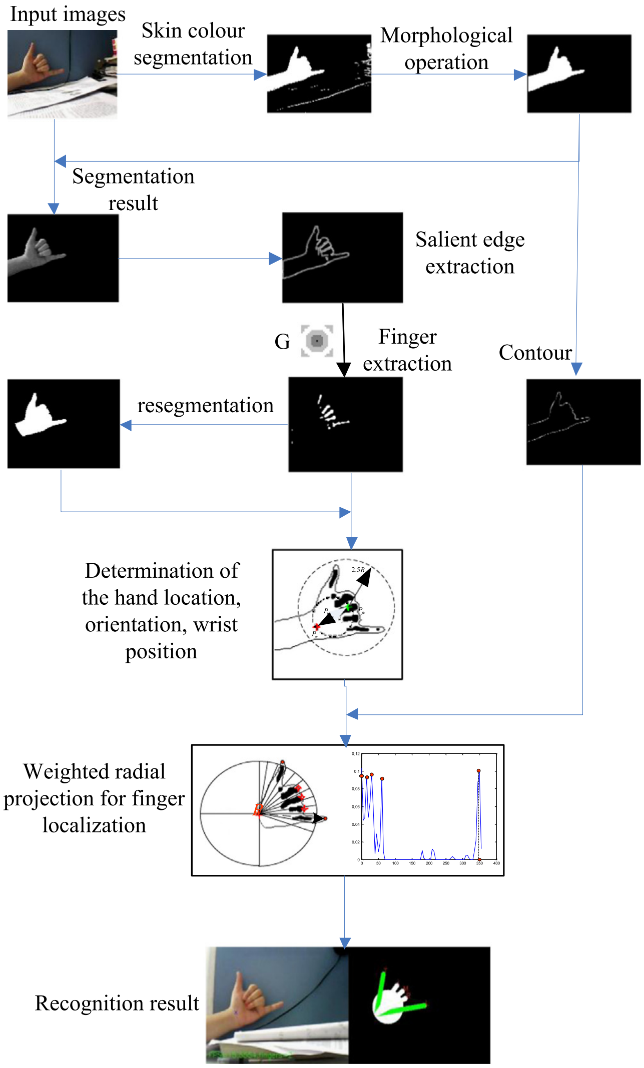


Fig. 12. The general framework for hand pose recognition procedure.

calculation:

$$P_w = P_{hc} - R_p \overrightarrow{N_p} \tag{14}$$

The diagram of the hand center P_{fc} , hand orientation $\overrightarrow{N_p}$ and wrist position P_w determination is shown in Fig. 10.

5.3. Weighted radial projection for finger localization

According to the kinematic hand model [12], if there is no finger crossing in the gesture, all fingers appear as a sector distribution on the palm with different angular rays starting from the wrist position. Based on this characteristic, the radial projection algorithm [19] is applied on the extracted finger central area, $I_f(x,y)$, to get the angular distribution of the finger pixels so as to determine the angles of the fingers.

Here, $I_f(x,y)$ (see Fig. 7) is transformed into polar coordinate $I_f(r,\theta)$ (see Fig. 11), with the origin as the wrist position P_w. The possibility of the finger existence is low when pixel points are close to hand wrist but the possibility of noise occurrence is rather high and vice versa. The projection algorithm used in [19] has been improved with a weighted coefficient, r. With equally divided angle in the circle, the weighted radial projection in each angle interval is calculated as

$$p_n = \int_{r1}^{r2} \int_{\theta_n}^{\theta_{n+1}} r I_f(r,\theta)\, d\theta\, dr, \quad n = 0, 1, 2, \ldots, M-1 \tag{15}$$

where P_n is denoted as the eigenvalue of the nth polar angle in the range of $[\theta_n, \theta_{n+1}]$; $[r_1, r_2]$ is the range of integral variable r, which can take the values $r_1 = 0.5R_p$ and $r_2 = 2.5R_p$; M is the number of polar angle interval, normally $M=360$. The finger feature sequence $\{P_n\}$ is arranged according to their angle values. A low-pass filter is used to smooth the sequence $\{P_n\}$ and the filtered sequence $\{P'_n\}$ is normalized as follows:

$$P_{n\_norm} = \frac{P'_n}{\sum_{i=0}^{M-1} P'_n}, \quad n = 0, 1, 2, \ldots, M-1 \tag{16}$$

where $\{P_{n\_norm}\}$ is used for finger localization.

Five fingers are located by the largest five local maxima of $\{P_{n\_norm}\}$, which is shown in Fig. 11. Within the angle range of finger presence, the farthest distance, D_c, from the external contour I_{hand_f} to the wrist position P_w is searched and the point is denoted as $P_{finger\_i(x,y)}$. It can be regarded as the precise location for fingertip (for extensional fingers) and finger joints (for flexional fingers). D_c is used to conclude the finger status $\{Status\_i\}$:

$$\begin{cases} \text{if } D_i > \alpha_i R_p & \text{extensional state;} \\ \text{elseif } D_i < \alpha_i R_p & \text{flexional state} \end{cases} \tag{17}$$

where α_i is a constant determined by the different finger lengths.

Till now, high-level gesture feature has been extracted completely and the hand gesture can be estimated successfully. The whole hand pose recognition procedure is illustrated in Fig. 12. With the proposed finger modelling approach and projection operator, finger edges and finger can be located without computational burden. The algorithm can be easily adapted to identify different hand poses in real-time with user independence and rotational invariance properties.

As the hand poses are constrained in frontal view of the camera, the fingers are ordered around the hand center in either extensional or flexional states. It should be noted that the recognition can be achieved with high accuracy when lighting conditions have little fluctuation. Different experiments have been performed to prove the effectiveness of the proposed hand gesture recognition algorithm.

6. Experimental results for real-time hand gesture recognition

The proposed hand feature extraction algorithm is used in a real-time hand posture estimation system for matching and tracking. The experiments are running with a PC (CPU: Pentium Core 2 Duo E7200, RAM: 2 GB) and a monocular web camera. The system could process real-time video of resolution 320×240 pixels with frame rate 15 fps. The tested hand stays at maximum 0.5 m distance in front of the camera, keeping the palm right facing the lens with $\pm 30^\circ$ anterior–posterior motion freedom and no finger crossing. In practice, this constraint is easily satisfied since it is natural to face the camera for hand motion. A video is attached as parts of the experiments.

The algorithm of edge extraction for fingers and hand described in Section 3 is tested in the experiments. In Fig. 13, the left column is the input images and the middle column is the salient hand edges extracted from the input images. The right column contains the finger central pictures, denoted in the white areas. The finger edges can be detected with satisfied performance without too much noise. All the fingers can be extracted accurately either in extensional or flexional status. However, it should be noted that the salient hand edge extraction would be disturbed easily by the environmental lighting. Severe variation in lightness would result in edge extraction failure.

Static gestures are performed to evaluate the performance of the proposed recognition algorithm. The test data are collected from videos captured by web-camera. The whole data set contains 14 randomly selected different types of static gestures, with 85–

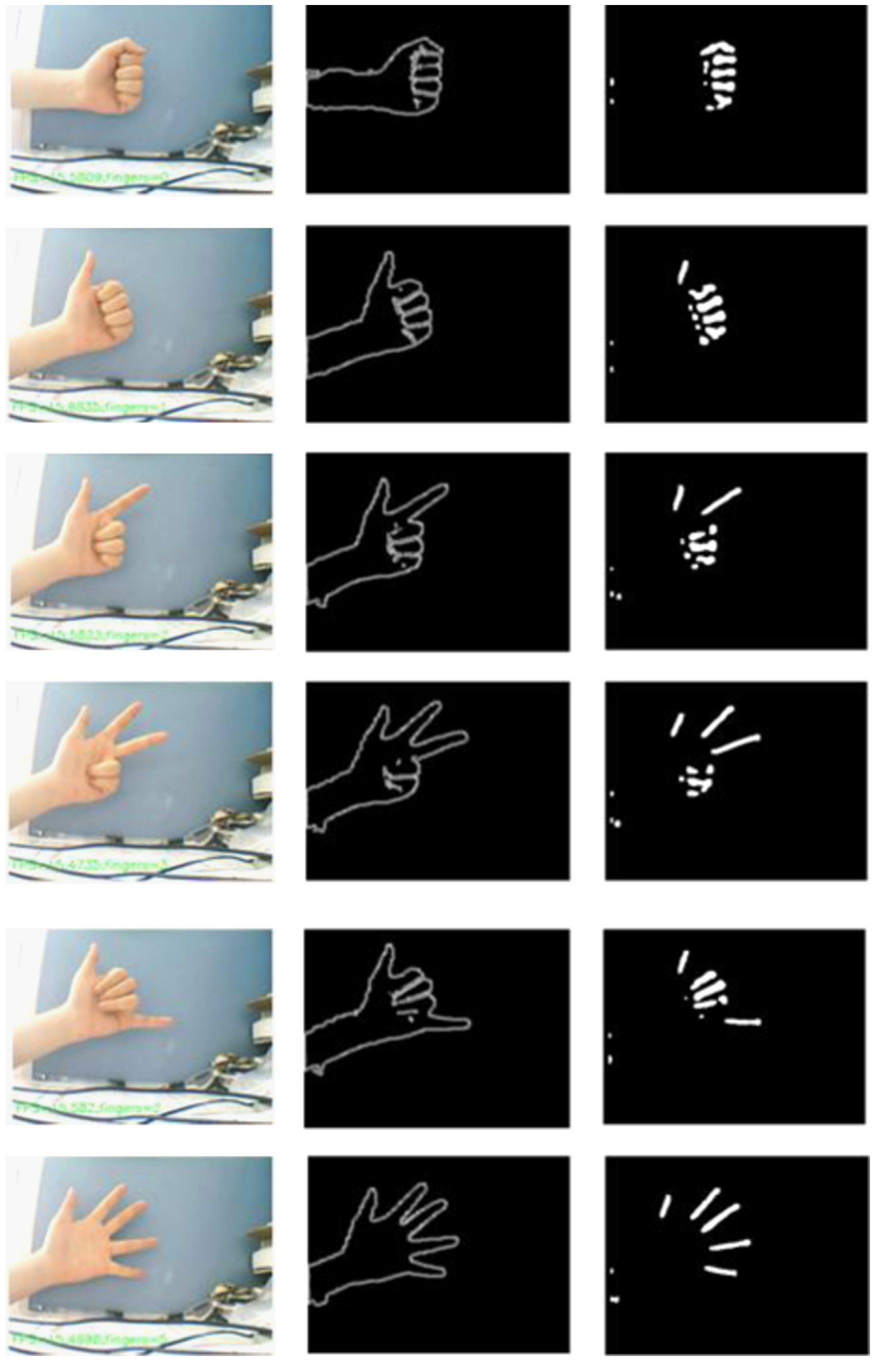


Fig. 13. Finger detection for different hand gesture: input images (left); edge images (middle); and finger detection (right).

150 samples for each gesture (shown in Table 3). The ID of a gesture is named by its number of extensional fingers and type ID. For example, 'ID 2-3' represents the gesture containing two extensional fingers and in the 3rd' type of gestures.

Each gesture recognition result is evaluated as follows:

(I) *Partial Correction Ratio (PCR)*: The extensional fingers can be recognized accurately, and the angle error of each finger is less than 8 degree:

$$PCR = \frac{\textit{Partially correctly detected number}}{\textit{Total sample number}} \times 100\%$$

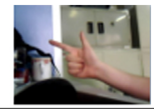
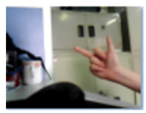
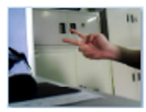
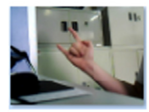
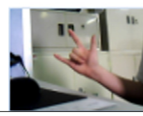
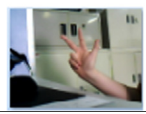
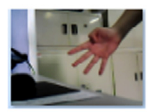
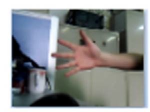
(II) *Full Correction Ratio (FCR)*: Both extensional fingers and flexional fingers are recognized accurately, and the angle error of each finger is less than 8 degree:

$$FCR = \frac{\textit{Full correctly detected number}}{\textit{Total sample number}} \times 100\%$$

The evaluation result is shown in Fig. 14, where the blue bars denote the percentage of *PCR* and red bars denote the percentage of *FCR*. It is concluded that all extensional fingers and flexional fingers except thumb can be detected and recognized correctly. It should be noted when the thumb is in flexional state, the proposed algorithm cannot always recognize its correct position due to lack of corresponding pre-definition.

Fig. 15 shows the hand orientation estimation result for a gesture sequence randomly selected 200 frames from the

Table 3
Examples of the tested gestures.

ID	0-1	1-1	1-2	1-3	2-1
Sample					
ID	2-2	2-3	2-4	2-5	3-1
Sample					
ID	3-2	3-3	4-1	5-1	
Sample					

experiments. The blue solid line denotes the real hand orientation, determined by the markers of the palm center and wrist position. The red dashed line denotes the estimated hand orientation with the proposed method. The black small dashed line is the estimation error, which is calculated by

$$Error = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2} \tag{18}$$

where y_i is the actual hand orientation or final angle and $\hat{y}_i$ is the estimation values.

In Fig. 16, it shows the middle finger angle estimation result for another gesture sequence (200 frames) selected randomly in the experiments. The middle finger direction is determined by the vector from the wrist position to the fingertip or joint of the third finger. The blue solid line denotes the real middle finger angle obtained by the markers on the middle finger and the wrist directly. The red dashed line denotes the estimated middle finger angle via the proposed method, and the black dashed line is the estimation error.

Experimental results demonstrate that the proposed method can extract the gesture center, orientation, location angle of the extensional fingers accurately. As for flexional fingers, good recognition performance can also be realized under accurate edge detection results. In summary, there are many advantages for the proposed hand gesture recognition algorithm:

1. It can identify almost all hand gestures under certain conditions.
2. It is insensitive to the hand rotation and finger subtle angle variation.
3. It can extract all finger features in extensional and flexional states.
4. It can determine the hand gesture orientation without additional assistance and will not be disturbed by the exposed arm area.

It has to be noted that the proposed hand pose recognition approach can only deal with the frontal view poses. The recognition of the side view of the hand poses cannot be guaranteed. The proposed algorithm relies mainly on the hand gesture segmentation performance. When there are a large area of skin color disturbance in the background, i.e., overlapped hand and face or worse lighting, it would cause recognition failure. As for some specific gestures, i.e., when fingers are in extensional state and clinging to each other,

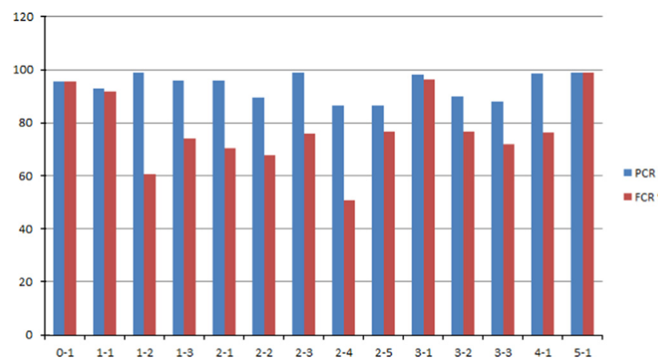


Fig. 14. Evaluation results of the gesture recognition. (For interpretation of the references to color in this figure, the reader is referred to the web version of this paper.)

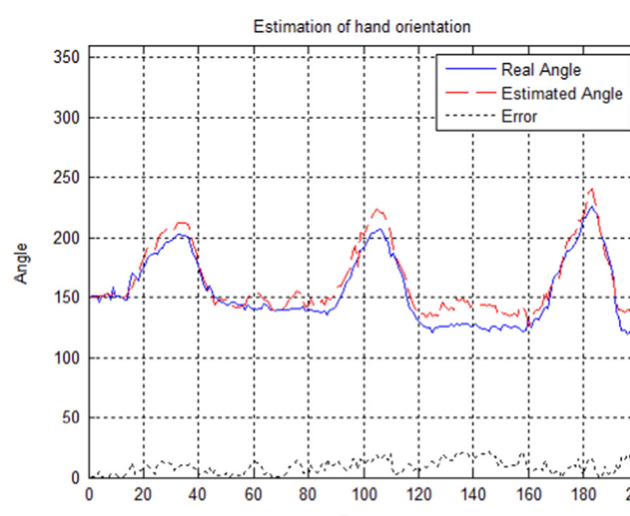


Fig. 15. Estimation of the hand orientation. (For interpretation of the references to color in this figure, the reader is referred to the web version of this paper.)

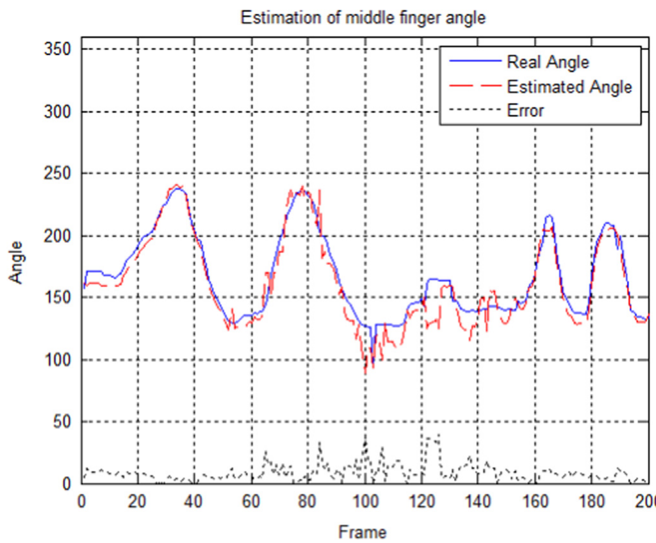


Fig. 16. Estimation of the middle finger angle. (For interpretation of the references to color in this figure, the reader is referred to the web version of this paper.)

which do not satisfy the sector shape condition, the hand orientation may not be located with the weighted radial projection algorithm.

7. Conclusions and future work

Feature extraction is a crucial module in computer vision based hand gesture recognition system. The performance of the module has a considerable effect on the accuracy, sensitivity and processing speed of the system. High-level hand features are desirable because they can provide compact representation of the input hand image in real-time operational mode.

In this paper, a new finger detection algorithm is put forward with a simplified finger model. The edges are extracted from the gesture images and then finger central area is obtained from the obtained edges. Salient hand edges are extracted for detecting extensional and flexional fingers via the parallel edge characteristics. The algorithm has little influence from the hand rotation angle variation such disturbances. Furthermore, a high-level hand feature extraction method is developed for hand gesture recognition in real-time. Angular projection centered on wrist position is then used for obtaining the angle, orientation and length of each finger. A simple 2D hand model is produced directly from these features. The proposed method can make accurate estimation for hand location, hand orientation and finger recognition. Hence, hand gesture can be recognized accurately and efficiently in real-time with high performance.

However, if the salient hand edge is not well detected, false or miss detection for flexional fingers could occur. Future work will focus on improving salient hand edge detection algorithm with tracking modules. Thumb and little finger models will be developed separately to improve the detection accuracy. More flexible posture segmentation algorithm will be explored to adapt to complicated application environments, i.e., depth camera can be adopted with depth information fusion. Moreover, an adaptive algorithm for finger width and palm width adjustment will be developed to accomplish the back-forward hand movement recognition. Combined with motion postures, substantial applications, such as human–robot interaction, can be applied in certain situations to execute dangerous operations.

Conflict of interest

None declared.

References

[1] J. Barrho, et al., Finger localization and classification in images based on generalized hough transform and probabilistic models, in: The Ninth International Conference on Control, Automation, Robotics and Vision, 2006. ICARCV '06, 2006, pp.1–6.

[2] M.A. Berbar, H.M. Kelash, A.A. Kandeel, Faces and facial features detection in color images, in: Geometric Modeling and Imaging—New Trends, IEEE Computer Society, London, England, 2006, pp. 209–214.

[3] L. Bretzner, I. Laptev, T. Lindeberg, Hand gesture recognition using multi-scale colour features, hierarchical models and particle filtering, in: The Fifth IEEE International Conference on Automatic Face and Gesture Recognition, 2002, pp. 423–428.

[4] M.K. Bhuyan, et al., Hand pose recognition using geometric features, in: 2011 National Conference on Communications (NCC), 2011, pp. 1–5.

[5] M.K. Bhuyan, D.R. Neog, M.K. Kar, Fingertip detection for hand pose recognition, Int. J. Comput. Sci. Eng. 4 (3) (2012) 501–511.

[6] A.-M. Burns, B. Mazzarino, Finger tracking methods using EyesWeb, in: Lecture Notes in Computer Science, vol. 3881, 2006, pp. 156–167.

[7] J. Canny, A computational approach to edge detection, IEEE Trans. Pattern Anal. Mach. Intell. PAMI-8 (1986) 679–698.

[8] C. Cao, et al., Real-time multi-hand posture recognition, Presented at the 2010 International Conference on Computer Design and Applications (ICCDA), 2010.

[9] C.-C. Chang, et al., Feature alignment approach for hand posture recognition based on curvature scale space, Neurocomputing 71 (2008) 1947–1953.

[10] Q. Chen, N.D. Georganas, E.M. Petriu, Real-time vision-based hand gesture recognition using Haar-like features, in: Instrumentation and Measurement Technology Conference Proceedings, 2007, pp. 1–6.

[11] N. Dung Duc, et al., Finger extraction from scene with grayscale morphology and BLOB analysis, in: IEEE International Conference on Robotics and Biomimetics, 2008. ROBIO 2008, 2009, pp. 324–329.

[12] A. Erol, et al., Vision-based hand pose estimation: a review, Comput. Vis. Image Underst. 108 (2007) 52–73.

[13] Z. Feng, et al., Features extraction from hand images based on new detection operators, Pattern Recognit. 44 (2011) 1089–1105.

[14] L. Gu, J. Su, Natural hand posture recognition based on Zernike moments and hierarchical classifier, in: IEEE International Conference on Robotics and Automation, 2008, pp. 3088–3093.

[15] N. Habili, L. Cheng-Chew, A. Hand Moini, Face segmentation using motion and color cues in digital image sequences, in: IEEE International Conference on Multimedia and Expo, 2001, pp. 261–264.

[16] M.J. Jones, J.M. Rehg, Statistical color models with application to skin detection, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1999.

[17] M. Kass, A. Witkin, D. Snakes Terzopoulos, Active contour models, Int. J. Comput. Vis. 1 (4) (1988) 321–331.

[18] C. Keskin, F. Kirac, Y.E. Kara, L. Akarun, Real time hand pose estimation using depth sensors, in: 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops), 2011, pp. 1228–1234.

[19] H.I. Kim, et al., Rotation-invariant face detection using angular projections, Electron. Lett. 40 (2004) 726–727.

[20] J.J. Kuch, T.S. Huang, Vision based hand modeling and tracking for virtual teleconferencing and telecollaboration, in: The Fifth International Conference on Computer Vision, 1995, pp. 666–671.

[21] R. Lienhart, J. Maydt, An extended set of Haar-like features for rapid object detection, in: International Conference on Image Processing, 2002, pp. 900–903.

[22] J.Y. Lin, W. Ying, T.S. Huang, 3D model-based hand tracking using stochastic direct search method, in: The Sixth IEEE International Conference on Automatic Face and Gesture Recognition, 2004, pp. 693–698.

[23] C. Manresa, J. Varona, et al., Real-Time Hand Tracking and Gesture Recognition for Human–Computer Interaction, 2000, pp. 1–7.

[24] Z. Mo, U. Neumann, Real-time hand pose recognition using low-resolution depth images, in: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, 2006, pp. 1499–1505.

[25] K. Nickel, R. Stiefelhagen, Visual recognition of pointing gestures for human–robot interaction, Image Vis. Comput. 25 (12) (2007) 1875–1884.

[26] K. Oka, et al., Real-time fingertip tracking and gesture recognition, IEEE Comput. Graph. Appl. 22 (2002) 64–71.

[27] K. Oka, Y. Sato, H. Koike, Real-time tracking of multiple fingertips and gesture recognition for augmented desk interface systems, in: The Fifth IEEE International Conference on Automatic Face and Gesture Recognition, 2002, pp. 429–434.

[28] S. Padam Priyal, Prabin Kumar Bora, A robust static hand gesture recognition system using geometry based normalizations and Krawtchouk moments, Pattern Recognit. 46 (8) (2013) 2202–2219.

[29] V.I. Pavlovic, R. Sharma, T.S. Huang, Visual interpretation of hand gestures for human-computer interaction: a review, IEEE Trans. Pattern Anal. Mach. Intell. 19 (7) (1997) 677–695.

[30] J. Romero, et al., Monocular real-time 3D articulated hand pose estimation, in: The Ninth IEEE-RAS International Conference on Humanoids 2009. , 2009.

[31] L. Sha, et al., Hand posture recognition in video using multiple cues, in: IEEE International Conference on Multimedia and Expo, 2009, pp. 886–889.

[32] N. Shimada, et al., Real-time 3-D hand posture estimation based on 2-D appearance retrieval using monocular camera, in: The Proceedings of the IEEE

ICCV Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems (RATFG-RTS'01), 2001.
[33] A. Thayananthan, B. Stenger, P.H.S. Torr, et al., Shape context and chamfer matching in cluttered scenes, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003, pp. 127–133.
[34] J. Triesch, C. von der Malsburg, A system for person-independent hand posture recognition against complex backgrounds, IEEE Trans. Pattern Anal. Mach. Intell. 23 (12) (2001) 1449–1453.
[35] E. Ueda, Y. Matsumoto, et al., Hand pose estimation using multi-viewpoint silhouette images, in: Proceedings of the 2001 IEEE/RSJ International Conference on Intelligent Robots and Sytems, Maui, Hawii, USA, October, 2001.
[36] W. Ying, et al., Capturing natural hand articulation, in: The Eighth IEEE International Conference on Computer Vision Proceedings, vol. 2, 2001, pp. 426–432.
[37] J.A. Zondag, T. Gritti, V. Jeanne, Practical study on real-time hand detection, in: The Third International Conference on Affective Computing and Intelligent Interaction and Workshops, 2009, pp. 1–8.

Yimin Zhou has obtained her Ph.D. degree from the Department of Engineering Sciences, University of Oxford, UK, in 2008. After that, she worked as a Research Associate at Loughborough University. Since 2011, she joined Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences and worked as an Associate Professor in Automotive Research Center. Her research interests include automatic control, image processing, fault diagnosis, electric vehicles and sensor optimization.

Guolai Jiang has received his M.Sc. at South China University of Technology in computing science area. Currently, he has worked as a Researcher at the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences and he is also a Ph.D. Candidate in the institute.

Yaorong Lin is a Professor at South China University of Technology. He has completed a lot of projects related to image processing and robot–human interaction techniques.