J Ambient Intell Human Comput (2018) 9:755–769
DOI 10.1007/s12652-017-0512-6

ORIGINAL RESEARCH

Fast marching method and modified features fusion in enhanced dynamic hand gesture segmentation and detection method under complicated background

Eman Thabet[1,2] · Fatimah Khalid[1] · Puteri Suhaiza Sulaiman[1] · Razali Yaakob[1]

Received: 19 February 2017 / Accepted: 21 May 2017 / Published online: 6 July 2017

Abstract Recent development in the field of human–computer interaction has led renewed interest in dynamic hand gesture segmentation based on gesture recognition system. Despite its long clinical success, dynamic hand gesture segmentation using webcam vision becomes technically challenging and suffers the problem of non-accurate and poor hand gesture segmentation where the hand region is not integral due to complicated environment, partial occlusion and light effects. Therefore, for segmenting complete hand gesture region and improving the segmentation accuracy, this study proposes a combination of four modified visual features segmentation procedures, which are skin, motion, skin moving as well as contour features and fast marching method. Quantitative measurement was performed for evaluating hand gesture segmentation algorithm. Besides, qualitative measurement was done to conduct a comparison based on segmentation accuracy with previous studies. Consequently, the experiment results showed a great enhancement in hand area segmentation with a high accuracy rate of 98%.

✉ Eman Thabet
Eman.Thabt.H@gmail.com

Fatimah Khalid
fatimahk@upm.edu.my

Puteri Suhaiza Sulaiman
psuhaiza@upm.edu.my

Razali Yaakob
razaliy@upm.edu.my

[1] Faculty of Computer Science and Information Technology, University of Putra Malaysia (UPM), Seri Kembangan, Malaysia

[2] Department of Computer Science, Faculty of Education for Pure Science, University of Basra, Basra, Iraq

Keywords Hand gesture segmentation · Skin segmentation · Motion segmentation · Fast marching method · Entropy filter · Viola Jones · Contour segmentation

1 Introduction

In recent years, dynamic hand gesture recognition system has become a major area of interest within the field of human–computer interaction. Hand gesture recognition has many applications in different fields such as medicine, virtual and augmented reality, telecommunications, and machines controls (Stergiopoulou et al. 2014). In contrast of traditional human–computer interaction devices such as keyboards and mice, hand gestures provide a more natural and intuitive contact between human and computers. This trend has become even more famous with the development in daily technology and intelligent computing (Simões et al. 2015). Basically, hand gesture recognition is separated into two categories; data-gloves based approaches (Oz and Leu 2011; De Marsico et al. 2014) and computer-vision based approaches (Zhao et al. 2012). The first type of approaches relies on extra hardware sensors linked to hand region to identify hand shape and trajectories, thus providing hand and fingers locations. On the contrary, such devices are costly, bulky and beyond the intuitiveness and natural interaction. Meanwhile, the second kinds provide natural contact and are more intuitiveness since they rely on camera vision, employing only video images processing and pattern recognition. Moreover, vision based approaches are divided into 3D vision based approaches (Bernardos et al. 2016) and 2D vision based approaches (Yeo et al. 2013). 3D vision based approaches utilise smart cameras supported by camera sensors including Microsoft Kinect.

Although such type of cameras provides detailed information and simplifies the process of hand gesture segmentation, it is yet costly, may cause health risks and do not included in most portable digital computing devices such as laptops and cell phones (Rautaray and Agrawal 2012). Nevertheless, 2D vision based approaches use only ordinary webcam camera making it to be more affordable, easy to use and included in most of portable digital computing devices (Rautaray and Agrawal 2012).

There are two categories of hand gestures, which are static (Kelly et al. 2010) and dynamic (Han et al. 2009). Static hand gestures refer to the local motion of hand fingers forming different shapes and postures. On the other hand, dynamic hand gestures represent free-air movement of hand in the spatial–temporal space. In addition, it involves gestures with different scales and postures of the hand. In this regard, dynamic hand gesture and 2D vision based approaches are the topic interests of this study. Dynamic hand gesture recognition systems comprised four stages, which are gesture segmentation, gesture tracking, feature extraction and gestures recognition. In segmentation, the goal is to find the exact position of the hand in camera scene by extracting hand doing gesture from the whole input scene. In tracking, this step describes the ability to follow hand region across consecutive video frames so that the system can identify which and where the hand object is moving at any particular time interval. Feature extraction step considers extract critical information on hand region discriminating different hand patterns. The final step is the recognition, which is responsible for interpreting the semantics of hand positions as well as hand trajectories and/or hand postures (Rautaray and Agrawal 2012). Consequently, hand gesture segmentation technology in vision-based approaches is one of the most challenging tasks (Choudhury et al. 2015). The quality of segmentation is able to directly affect other recognition systems stages.

Dynamic hand gesture segmentation based on 2D vision based approaches is a challenging process. Yet, there are many issues that influence the performance of dynamic hand gesture segmentation. The main problems of hand gesture segmentation include the poor hand gesture area segmentation accuracy due to complicated environment and various lighting conditions. The dynamic hand gesture segmentation algorithm from implementing motion feature of two frames difference method combined with skin feature of generic thresholding skin segmentation algorithm suffers complicated environment and different illumination conditions. Thus, it resulted in the extracted hand region producing holes, missing and incomplete parts (Asaari et al. 2014). Furthermore, several studies (Bhuyan et al. 2014; Sgouropoulos et al. 2014; Vafadar and Behrad 2014; Yeo et al. 2013) have presented dynamic hand gesture segmentation with their performance limited and degraded under complicated environment and various lighting conditions.

2 Previous works and motivation

To highlight the challenges of vision-based hand gesture segmentation, several approaches in the literature were adapted in terms of visual features such as colour, motion information, shape or a combination of these features (Zabulis et al. 2009; Stergiopoulou et al. 2014). Most hand gesture segmentation approaches tend to either segment hand region doing gesture or estimate the shape of hand (Zabulis et al. 2009). Bhuyan et al. (2014) used the fusing of Cb and Cr, H and S chrominance components in YCbCr and HSV colour spaces, as well as the largest connected component method to get palm region of hand. However, their method still lack of segmentation and tracking accuracy. On the other hand, Sgouropoulos et al. (2014) have utilised Viola Jones algorithm to detect face and get skin colour tones trained immediately from face region in YCbCr colour space. Besides, hand blob localising and gab filling algorithms were utilised to completely segment and track hand gesture in the entire image. In contrast, the segmentation process depended on the success of Viola Jones method for face detection. In addition, extracting skin tones immediately form face region can result in noises due to the effects of background on face area. Vafadar and Behrad (2014) used colour information corresponding to H, S and Q, I, in HSV and YIQ colour spaces and K-mean algorithm. In contrast, hand segmentation algorithm needs to be improved since its performance was degraded under complex background and overlapped with other objects problems. Yeo et al. (2013) utilised skin colour in Y, Cb and Cr colour space components, Haar like feature to remove face region and Canny edge approach. Nonetheless, the algorithm was observed to be better in indoor situation under normal lighting condition and may therefore degrade under very dark or very bright illumination conditions.

Yun et al. (2012) employed HSV colour space for skin colour detection to obtain hand shaped region and extract the hand region contour by getting the maximum contour of the hand shaped area. Although this method had shown a great potential, it is still not feasible in the case of dynamic hand gesture moving in complex background and practicing partial occlusion with a face or other objects of the same skin colour. Additionally, their method for contour extraction is not feasible in the case of closed hand forming a fist.

Stergiopoulou et al. (2014) demonstrated a combination of existing techniques involving four primary stages; motion detection, skin colour detection, morphological descriptors, and the combination of extracted information.

In motion detection, they used a hybrid technique that includes three frames differencing in detecting sudden motion to obtain motion region of interest (MROI) followed by background subtraction applied on MROI to capture the hand even when it stops moving momentarily. Then, skin detection was done based on colour categorising technique and specifically on updated version of skin probability map (SMP) or histogram-based Bayes classification approach in HSV colour space. They employed morphological descriptors as feedback stage, utilising the final detected hand to be described in the aspect of weight factors to measure the minimum distance of hand pixels to hand contour and approximate the probability that this pixel belongs to the hand region in the current frame. Finally, the fusion of extracted information was accomplished in the region-based approach to get final hand detection. In their proposed approach, they aimed to detect the hand in real time, not necessarily constantly, in front of a non-unvarying background under different illumination cases. However, they assumed that the hand should be the largest object moving in the scene and the background should be somewhat static. They estimated that the occurrence of the user's faces in the view is not difficult as long as it is static so that it can be treated as a part of the background. Although this method has produced good results in terms of hand region detection since the detection rate can be as high as 98.75%, it is yet to be further improved to detect the hand shape where the accuracy of shape detection reaches 88.02%.

Wang et al. (2014) detected skin colour by utilising HSV colour space (Hue, Saturation, and Value). A binary image was obtained according to the HSV colour range of hand skin. Then, the "1" region was enclosed by longest "1" contour filled with "1" with the rest of them filled with "0" to consequently obtain a binary image for hand region. However, their method has failed under such cluttered background when hand region is occluded with other objects of the same skin colour range. As such, the whole method needs to be improved in the future.

Meanwhile, Mazumdar et al. (2015) suggested adaptive hand segmentation and tracking system. They proposed several skin colour detection models associated with morphological operation addressing the problems such as complexity of the background and conditions of changed illumination. However, the output target in their method was located and segmented using bare and gloved hand to consider the situations of illumination and skin colour changes. Although the results turned out to be good, there is a restriction to their method caused by dynamic background.

Asaari et al. (2014) proposed an adaptive tracking method for dynamic hand gesture trajectory recognition system. In their method, the detection and segmentation method used a combination of motion feature from two frames differences method and skin colour feature of generic thresholding model, which employed Cb–Cr components as thresholding values. The values of Cb and Cr chrominance thresholds were calculated manually or offline from the histogram out of many skin colour samples based on mean and standard deviation of Cb and Cr components. However, their segmentation method revealed weak or incomplete segmentation for hand area and degraded under low lighting conditions, thus influencing the segmentation accuracy rate of the entire method. The segmentation accuracy rate obtained after analysing the method and proving results is 64.2%.

Chidananda et al. (2015) developed automated ear detection in facial images method to localise ear image under challenging situations such as varying backgrounds, occlusion and various postures. This study proposed an integration of entropy texture analysis filter and Hough transform to improve and get accurate detection of ear image.

Hence, to improve the dynamic hand gesture segmentation accuracy rate against the problem of weak hand feature segmentation or incomplete hand region segmentation under complicated environment, different illumination conditions and partial occlusion, this paper proposes fast marching method (FMM) combined with four modified visual features segmentation procedures of skin, motion, skin moving and Contour features. The contributions of this study are summarised as follow;

1. Developing dynamic hand gesture segmentation method based on a good segmenting visual features fusion.
2. Using thresholding technique upon two frames difference method and FMM to segment motion feature as well as improving poor motion segmentation using two frames difference alone.
3. Developing skin feature segmentation scheme based on generic threshold factors in Cb–Cr colour components calculated using online-training procedure out of nose pixels in the face region based on Viola Jones algorithm. This is to get rid of background influences over face region and to make the algorithm adaptive to different users.
4. Alternatively, to cope with Viola jones performance limitation under low illumination conditions and side view of the face or face rotation, this study proposed offline-training procedure. This procedure calculates threshold factors in Cb–Cr components using skin colour data samples taken from face region and weighted equation by Park et al. (2009) for obtaining skin colour threshold factors used to segment skin colour.
5. Using Entropy filter on fused skin moving feature has contributed in extracting smooth portions inside hand region and strengthening the feature by increasing the

intensity (enlarging the entropy) inside skin moving region.
6. Using two image-frames subtraction, canny edge detector algorithm, and range filter to segment hand contour feature has correctly formulated the hand shape and emphasised the edges of hand region.
7. The proposed formula for features fusion has successfully detected the seed location mask of moving hand for fast marching method (FMM) to accurately segment hand region besides employing FMM for dynamic hand gesture segmentation.

3 Moving hand gesture segmentation system using FMM

The proposed system includes six phases. They are skin feature segmentation, motion feature segmentation, enhanced skin moving feature segmentation, contour feature segmentation, features fusion and the use of FMM. Figure 1 shows the block diagram of the proposed system for hand gesture segmentation. The details for every model are included in the following subsections.

3.1 Fast marching method

Fast marching method (FMM) is a numerical method created by James Sethian (1996) for solving boundary value problems in the Eikonal equation. The algorithm is similar to that of Dijkstra's considering that information only streams outward from the seeding region. The main advantages of this technique are that it depends on a fixed Eulerian mesh, cop

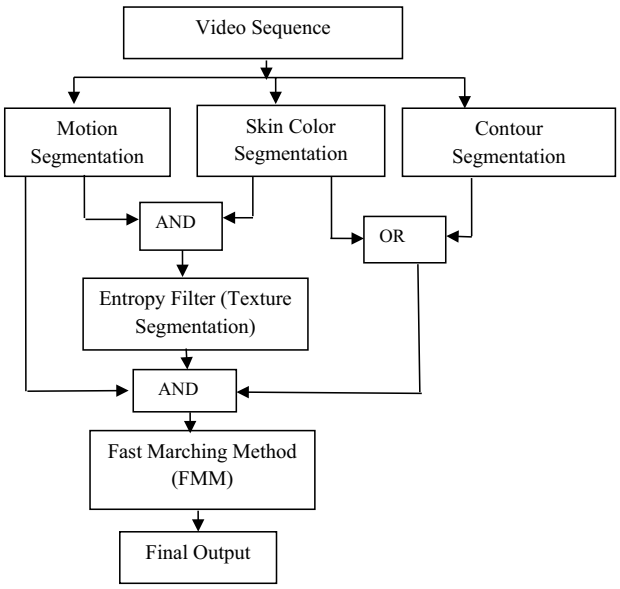


Fig. 1 Block diagram of moving hand gesture segmentation system

topological changes in the interface naturally and can be easily formulated in higher dimensions (Zhu and Tian 2003). In addition, the FMM work is based on pixel seeds propagation and object boundary tracking with rapid computation time (Sethian 1999). Accordingly, Fast marching method (FMM) in the proposed method was employed in many stages of this study including motion feature segmentation, skin colour feature segmentation, and finally in getting hand gesture segmented from the input video scene. This was done to get an accurate segmentation to dynamic hand region while moving. More information and explanation about the FMM can be found in the studies of (Sethian 1996, 1999; Monneau 2010). In this study, the FMM method was implemented based on image gradient or intensity difference and threshold level using formula in Eq. 1.

$$BW = FMM(W, MASK, THRESH) \tag{1}$$

where BW is binary segmented image; W is a weight array computed based on intensity difference or image gradient, which identifies non-sparse, non-negative digital array in which high values identify the foreground (object) and low values identify the background; MASK is the mask of seed locations specified as a logical array with a similar size as W so that the seed locations can be located where mask is true; *and THRESH* is non-negative scalar species in the range [0 1] used to obtain the binary image.

3.2 Motion feature segmentation

A successful motion segmentation method has to cope with difficulties of noisy background and different illumination conditions. In this regard, the most known and computationally effective approach is the frame difference method, which has the ability to detect sudden motions and has a low computation time in particular for scenes captured by stationary camera (Stergiopoulou et al. 2014; Ren et al. 2013). The flowchart of the proposed motion segmentation algorithm is illustrated in Fig. 2.

Based on the data flow of the proposed algorithm, $fram_t$ represents the RGB image at time (t) and $fram_{t+1}$ is the RGB image at time (t+1) of video sequences. In the first step, $fram_t$ and $fram_{t+1}$ were converted to gray level. Then, $Framdif_t$ was calculated based on the difference between $fram_t$ and $fram_{t+1}$, as defined in Eq. 2.

$$Framdif_t(x, y) = \left| fram_{t+1} - fram_t \right|. \tag{2}$$

Then, the $Framdif_t$ image was converted into binary image $Framdifb_t$ by applying Eq. 3.

$$Framdifb_t = \begin{cases} 1, & \text{if } Framdif_t \div 55 \geq Thr \\ 0, & \text{otherwise} \end{cases}. \tag{3}$$

The threshold value (Thr) was calculated using Eq. (4) based on the study of Asaari and Suandi (2010).

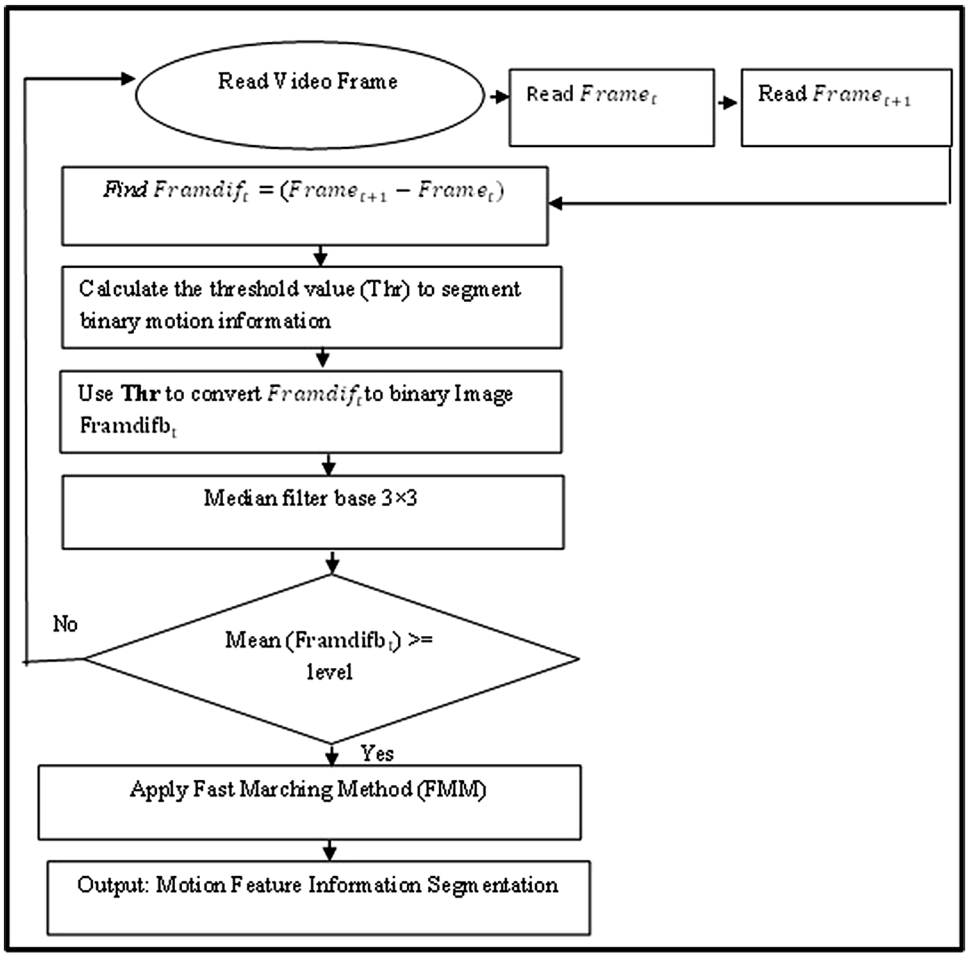


Fig. 2 Flowchart of motion feature segmentation algorithm

$$Thr = \begin{cases} 0.05 \times mean(Framdif) \leq 1 \\ 0.2, \quad \text{otherwise} \end{cases}. \tag{4}$$

To further smooth out the segmented image, median filter, which is a spatial filter with a 3×3 masking window, was applied to eliminate the unwanted noises remained in the motion image. Next, as depicted in Fig. 2, the mean intensity of smoothed motion feature mask resulted from the previous step *Framdifb* was calculated and thresholded with the threshold factor namely level (level was experimentally determined and set to 0.3). Thus, the motion information of the smoothed binary image *Framdifb* is considered if the mean intensity of smoothed *Framdifb* is greater than level threshold value.

However, the motion information segmented based on two frames difference and threshold operations (*Framdifb*) may result in poor motion feature segmentation with holes. Therefore, to enhance the segmented motion feature and compensate holes, this study applied the FMM as illustrated in Eq. 5.

$$Framdifb = FMM(W, \ Framdifb, \ gthresh), \tag{5}$$

where *Framdifb* is the segmented motion feature after compensate holes and incomplete parts, W is gradient weight function in Matlab (represents the weight-array) calculating weights for image pixels depending on image gradient as depicted in Eq. 6.

$$W = gradientweight\ (Image, \ \sigma)\ , \tag{6}$$

where σ represents the standard deviation for Gaussian and experimentally set to 1.5; *gthresh* is a positive scalar in the range of [0 1] that identifies the threshold level and set to 0.01, which was obtained experimentally.

3.3 Skin color feature segmentation

Skin colour information is more robust against geometric variations caused by scaling, rotation or translation. On the other hand, to cope with the complexity of environment, different illumination conditions, time complexities and data nature challenges of skin colour segmentation, this study proposes a new skin colour information segmentation scheme. The presented scheme proposed threshold factors (thresholds values) based on maximum and minimum range values of Cr–Cb colour spaces. The threshold factors

range was calculated utilising either online-training procedure from nose-pixels of face region—or offline-training procedure out of a number of skin samples alternatively. Figure 3 depicts the flowchart of colour skin segmentation.

According to Fig. 3, the proposed scheme was observed trying to segment skin features using threshold factors gained from online-training procedure at the beginning of implementation. In online training procedure, the Viola Jones algorithm (Lienhart and Maydt 2002; Viola and Jones 2001) in the YCbCr colour space was used to detect face and nose regions accordingly for extracting the minimum and maximum values of the Cb chrominance component and the Cr chrominance component, respectively, so that the threshold values calculated from nose region will be in the range (minCr, maxCr, minCb, maxCb). In order to reduce computation time of Viola jones algorithm, the online training procedure was carried out once for individual user. Moreover, to cope with high complexity drawback, the threshold factors were trained for one time where the skin threshold factors for the whole system are calculated as depicted in Fig. 4.

After getting threshold factors from nose region, the face region was removed to get rid of noise. Next, binary information of skin areas in the image were segmented via thresholding operation as depicted in Eq. 9.

Furthermore, FMM was applied into the segmented skin feature to correct the boundary and compensate the holes or missing pixel parts of hands skin and other skin regions in the image-frame since the FMM method is based on pixel seeds propagation and object boundary tracking with low computation time. The formula to call the FMM method is illustrated the Eq. 7.

$$BW = FMM(W,\ MASK,\ THRESH), \tag{7}$$

where BW represents the enhanced binary segmented skin feature using online thresholds training procedure; W is the weight array that takes the same values of input image frame but in the grayscale level; $MASK$ is the binary segmented skin colour feature from online thresholds training procedure; and $THRESH$ is a positive scalar in the period [0 1]. Basically, THRESH identifies a level at which the outcome of FMM performs a thresholding operation. Here, $THRESH$ was set to 0.001 by experiments and observations.

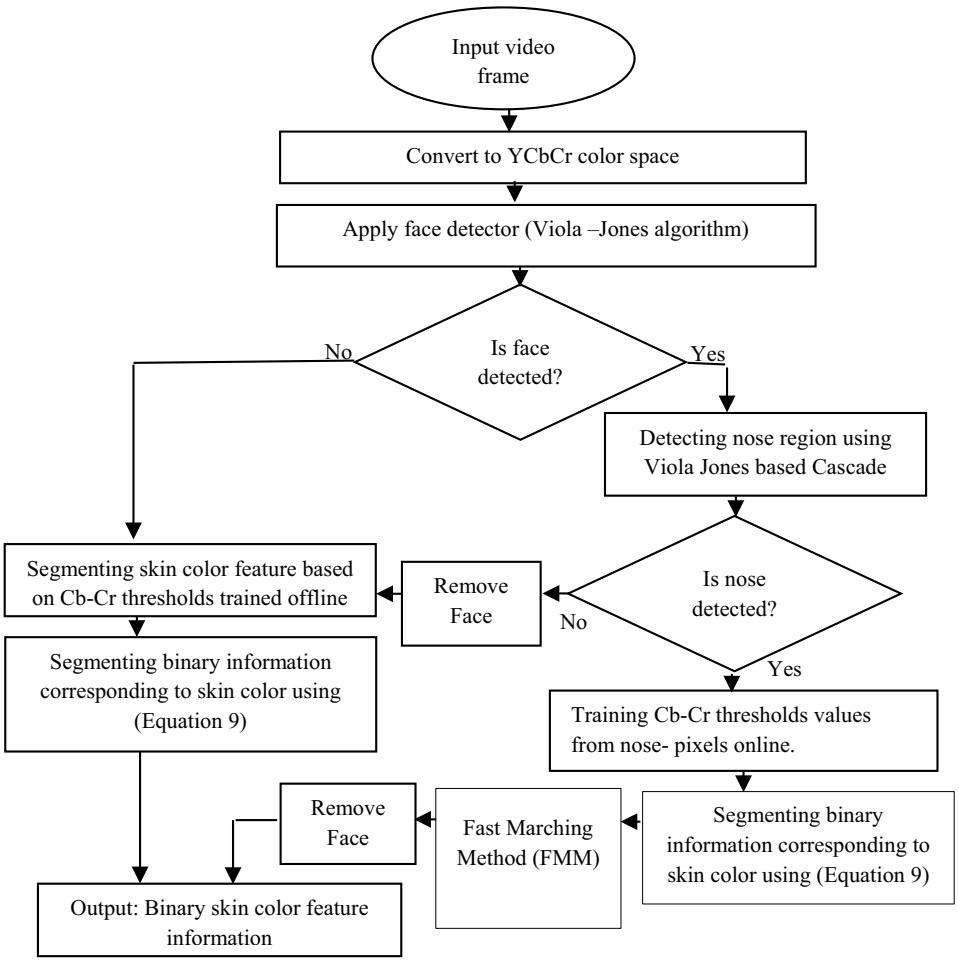


Fig. 3 Skin colour segmentation scheme

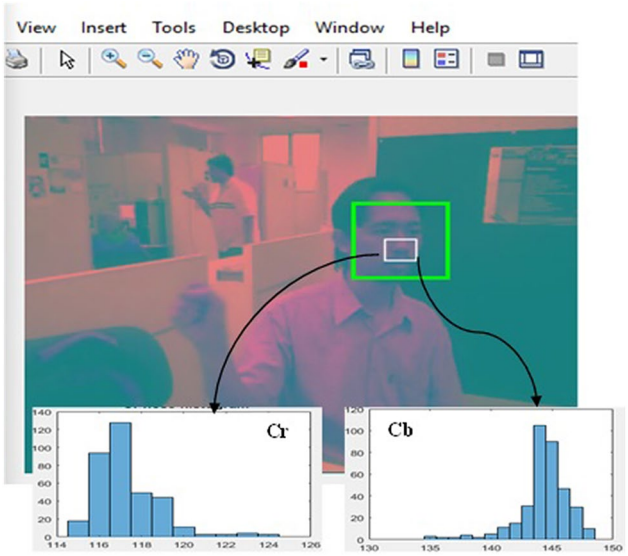


Fig. 4 Online training thresholds

However, under situations of low lighting conditions and face rotation, Viola Jones showed degradation in its performance to detect the face or/and nose region. To cope with this problem, offline training procedure was designed to be carried out alternatively. The offline training procedure calculated threshold factors by adjusting the weight parameter *alpha* between the calculated parameters thr1 and thr2 using weighted equation (Eq. 8) inspired by (Park et al. 2009). In this regard, the impact of weight *alpha* here within the Eq. 8 is to create adaptive skin feature segmentation based generic offline thresholds training model that can represent different human's skin colour, differentiate between skin and non-skin objects and even similar to skin objects as possible, robust to camera characteristics and nature of data as possible, as well as adaptive to various lighting conditions and complex background.

Moreover, to implement offline training procedure, 11 skin samples of video frames of IBGHT dataset were chosen and the values for skin range were taken out of face region from different 11 users. In fact, based on previous studies, there is no standard measurement and clear statement for the number of skin samples that should to be used for training skin thresholds. Based on the literature, in particular for generic approaches based skin detection and segmentation, every presented study has utilised its own number of samples according to experimental results. Therefore, here in this study, skin samples were selected by considering as many skin detection and segmentation challenges as possible. In another word, these skin samples displayed the upper part of human body including moving hand under various lighting conditions and environmental situations as well as variety of skin colour among users by using video frames of IBGHT dataset.

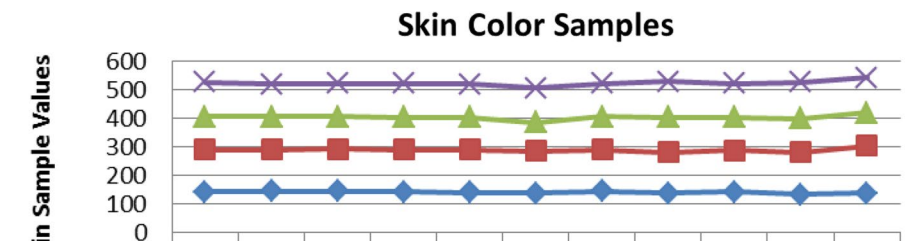
The values of thr1 and thr2 parameters for (Eq. 8) were picked based on maximum and minimum values in each Cb and Cr skin sample region taken out as threshold values (Fig. 5). Then, from every Cr_min, Cb_min and Cr_max, Cb_max of manually calculated skin color samples values ranges as in Fig. 5, two minimum and maximum thresholds values were extracted for parameters in thr1 and thr2 of Eq. 8, consecutively.

$$thresholdsvect = alpha \times thr2 + (1 - alpha) \times thr1, \quad (8)$$

where *alpha* is the weight with its value by experiments and observation set to 0.02; thr1 and thr2 are first and second extracted threshold vectors based on minimum and maximum range values of Cr_min, Cr_max, Cb_min and Cb_max of skin samples (Fig. 6). Meanwhile, *thresholdsvect* represents calculated thresholds for skin colour segmentation based on Cr–Cb range.

In the end, to segment skin regions, binary pixels corresponding to skin regions were segmented by performing thresholding operations using Eq. 9 where $S_n(x, y)$ is the binary image of skin segmentation.

Fig. 5 Extracted skin colour samples



	1	2	3	4	5	6	7	8	9	10	11
Cb_ max	119	115	116	117	117	121	117	125	119	127	121
Cb_ min	117	114	113	113	113	100	114	123	115	118	116
Cr_ max	147	147	149	149	149	148	147	141	146	148	165
Cr_ min	143	145	145	143	142	138	145	140	143	134	140

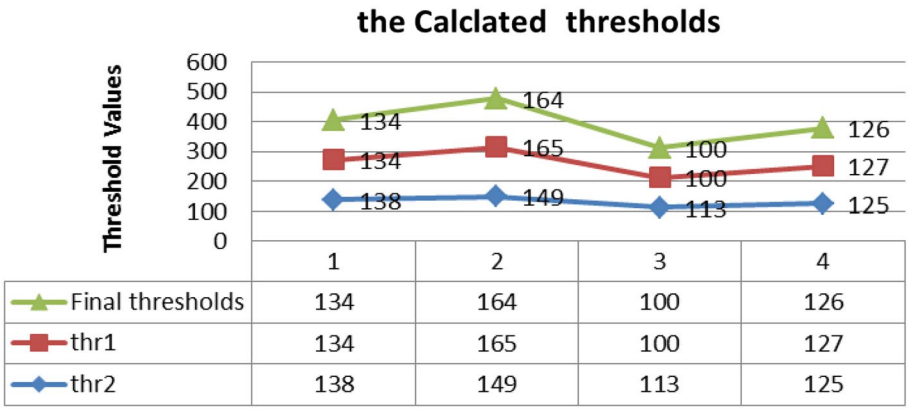


Fig. 6 The obtained threshold factors based on offline training procedure for skin segmentation

$$S_n(x,y) = \begin{cases} 1, & \text{if } Cb \in Cb_{rang} \text{ and } Cr \in Cr_{rang} \\ 0, & \text{if otherwise} \end{cases}. \quad (9)$$

3.4 Contour feature segmentation

Plenty of information can be gained from contour extraction in an image. It can formulate the shape of hand gesture. Therefore, in this study, to get an accurate segmentation of the hand region and to avoid lacking in hand contour, contour features extraction model was proposed as a part of moving hand gesture segmentation method. In this study, the contour feature of the moving hand region was segmented using two image-frames subtraction, canny edge detector algorithm and range filter. The contour feature segmentation algorithm is depicted in Fig. 7.

Basically, contour extraction depends on edge detection outcomes. In this regard, Canny edge detector algorithm was employed since it has the ability to detect weak edges and adapt to environmental variations (Sgouropoulos et al. 2014). The proposed algorithm took the same inputs of motion segmentation algorithm, which are two successive frames $fram_{t-1}$ and $fram_t$. Firstly, the Canny algorithm was applied on the red component of every RGB image frame $fram_{t-1}$ and $fram_t$ to reduce the influence of shadow projected from subtracting both images. The red image frame of $fram_{t-1}$ is $fram_{t-1}(:,:,1)$, whereas $fram_t$ is $fram_t(:,:,1)$. The results from this step are $Canny - fram_{t-1}$ and $Canny - fram_t$. On one hand, the Canny edge algorithm resulted in a large amount of edges describing hands and to unrelated objects on the background. For that reason, this study proposed the subtraction operation between current $Canny - fram_{t-1}$ and $Canny - fram_t$ (subtraction

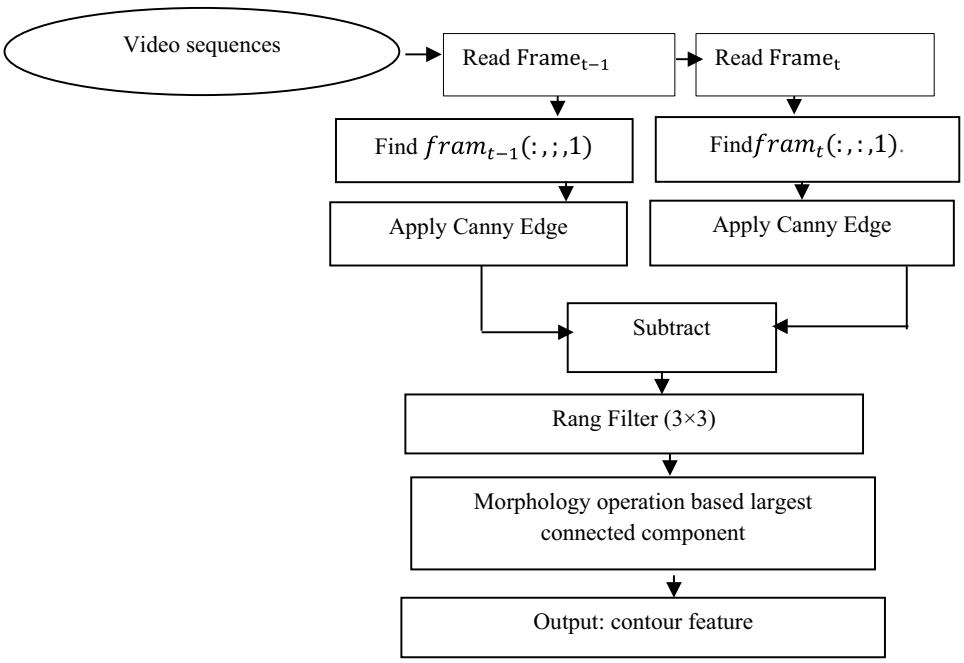


Fig. 7 Block diagram of contour feature segmentation algorithm

of two canny image frames). On the other hand, the post-processing algorithm is needed to enlarge the reliability of such algorithms. As a result, the Range filter (Bailey and Hodgson 1985) using a range value of 3 by 3 neighbourhood around the corresponding pixel in the input image (image resulted from subtraction) was applied to refine the result since the range filter can emphasise edges and change images texture areas (Eq. 10). Then, a morphology operation extracting the largest connected component was applied to filter the segmented image and eliminate unrelated small blobs and undesirable noise.

$$CIMGF = Range\ Filter(Imsubtract(Canny - fram_{t-1}, Canny - fram_t), [3\ 3]), \quad (10)$$

where the range filter is a texture analysis filter with the ability to emphasise local differences in brightness independent of the average brightness in the area using a short calculation time, CIMGF is the resulted binary contour image feature. More information regarding principle work of range filter can be found in (Bailey and Hodgson 1985). Furthermore, an example of the steps in moving hand gesture contour segmentation algorithm is illustrated in Fig. 8.

As a result, the contour feature of the moving hand gesture was successfully segmented by the developed algorithm under different scenarios and environmental circumstances of different video sequence. However, the obtained hand contour feature sometimes is not very pure and has some noises from the background. Using only the hand contour feature is not the best choice for hand gesture segmentation; so, there is always a need to incorporate other features.

3.5 Enhanced skin moving feature extraction

Frame difference has a low computation time and the ability to detect sudden motion for scenes captured using stationary camera (Ren et al. 2013; Stergiopoulou et al. 2014) with poor motion segmentation. This problem can impact the correctly segmented skin moving information inside the image of video sequences for the moving hand gesture segmentation process. The segmented skin moving feature results in a lot of noises, making the segmented skin moving feature to become weak (with a lot of holes inside). Such noises would be influenced by poor segmentation of motion feature because of similar values of intensity (Sgouropoulos et al. 2014).

In order to enhance and strengthen the extracted skin moving feature belongs to the hand region for hand gesture segmentation, entropy filter-based texture analysis was

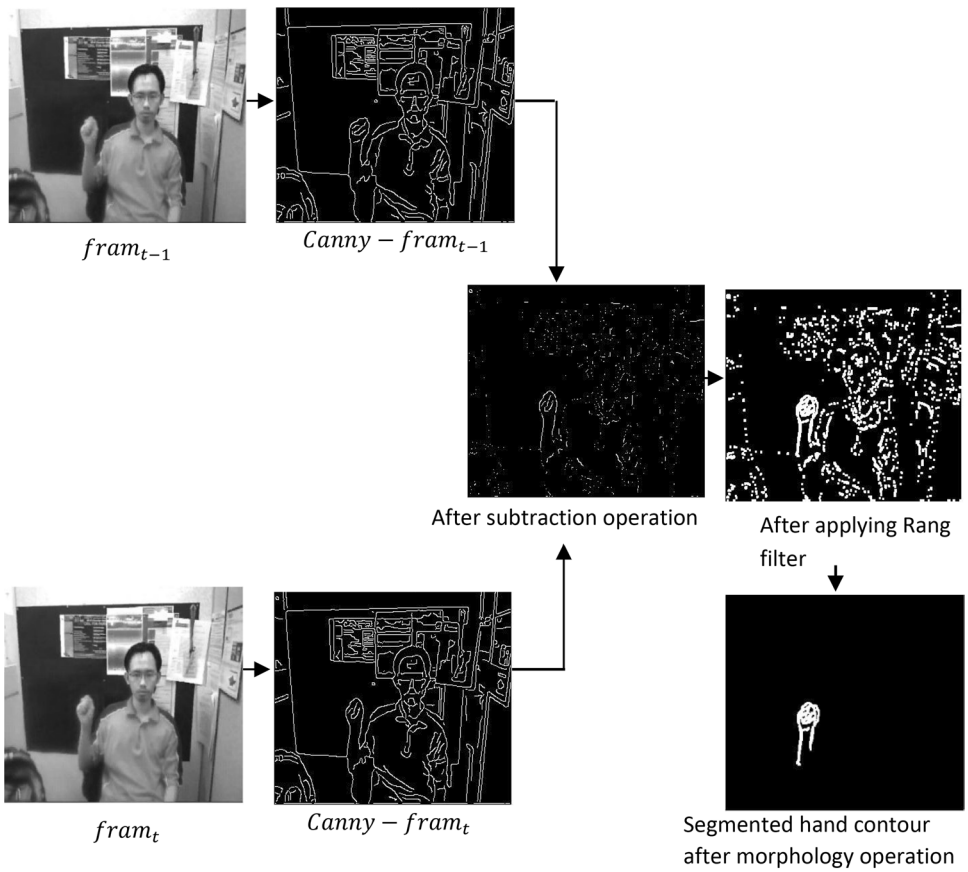


Fig. 8 Experiment steps of moving the hand contour feature segmentation algorithm

proposed as an integration for the skin moving feature since it can effectively extract a smooth portion of an image without being affected by radiance and darkness. In addition, it employs the property making an image to become clear in proper focus (enlarge the entropy or intensity), whereas the image becomes smooth out of focus when reduces the entropy (Hamahashi et al. 2008). The operation can be accomplished using Entropy filter equation (Eq. 11).

$$Entropy\ value = \sum_{l=l_{min}}^{l_{max}} P(l) log P(l), \tag{11}$$

where P (l) is a normalised intensity histogram produced by getting an intensity histogram H (l) for the region of image. Based on the equation above, the entropy filter function can be called as $J = ENTROPYFILT(I, Nod)$, where I represents the input image and Nod describes the neighbourhood, which is a multidimensional matrix of zeros and ones in which non-zero elements identify the neighbours. *Nod* must have an odd size in each dimension and should be a structure, element or object.

In this study, every production pixel has a specified 9 by 9 neighbourhood entropy value around the corresponding pixel in the input image *I*. The value 9 by 9 represents the Nod matrix and its size was determined empirically. *J* is the enhanced returned array of binary skin moving segmented information. The above explanation can be represented in the next two Eqs. (12) and (13).

$$SM_t = S_t \& Framdifb_t \tag{12}$$

$$ESM_t = Entrpy\ Filter\big(SM_t, [9\ 9]\big), \tag{13}$$

where SM_t is a skin moving feature and ESM_t is the enhanced skin moving feature.

3.6 Locating hand region

Accordingly, this study has proposed the use of fast marching method (FMM) to segment binary image of dynamic hand gesture region during its movement. For FMM, the locations of seed pixel propagation and weight array for image frame pixels and threshold factor are required to implement the FMM and segmenting hand gesture region.

Therefore, after successfully extracting the motion, skin colour, skin moving (ESM) and contour features (CIMGF), the seed location of hand region (*HROIMask*) was estimated by fusing these extracted features using the proposed formula based logical operators as illustrated in Eq. 14. The weight array (W) was calculated based on grayscale intensity difference function of input image frame. Consequently, the binary region of the hand was segmented using FMM as in Eq. 15. By implementing Eqs. 14 and 15, the face regions and other skin coloured objects as well as unrelated moving objects were discarded.

$$HROIMask = ESM_t\ \&\ (CIMGF_t |\ S_t\ \&\ Framdifb_t) \tag{14}$$

$$Binary - HROI = FMM(W,\ HROIMask,\ gthresh), \tag{15}$$

where W is computed utilising the average of intensity values of the whole pixels in $fram_t$ marked as logical true in the calculated or obtained (*HROIMask*), using the formula $W = graydiffweight(grayscale\_img, HROIMask)$; $gthresh$ is the non-negative threshold value experimentally obtained and its value was set to 0.001.

As a result, the hand region of interest (HROI) was detected using Eq. 16; the output of hand detection is a rectangular bounding box denoting the region of hand in form [c, r, w, h], which was later utilised to benefit from width and high parameters (w and h), as well as the segmented region of hand was treated as a reference region in the matching operation between two features to find hand region on the next frame. It should be noted that the reference region (current segmented HROI) will be updated with time at the end of each correct segmentation to hand region in $frame_t$.

$$HROI = Rectangle\ (Binary - HROI,\ ([c,\ r,\ w,\ h]), \tag{16}$$

where w = max [Col] − min [Col], h = max [Row] − min [Row], c = min [Col] and r = min [Row]; [Col] and [Row] are vectors containing column and row coordinates extracted from white pixels of Binary-HROI image, w and h represent the width and height of the bounding box, c and r are the starting points for the bounding box in x and y, respectively. The illustration of hand region of interest HROI localisation is depicted in Fig. 9.

Often when a gesturer vigorously moved his hand, the segmented hand region image included portions of the gesturer arm part. Therefore, local improvement is necessary to eliminate the arm region. According to (Asaari and Suandi 2010), the arm region can be discarded from the hand region by utilising a simple local enhancement technique based on the criterion in Eq. 17.

$$h = \begin{cases} \gamma w, & \text{if } h < w\ |\ h > \gamma w \\ h, & \text{Otherwise} \end{cases}, \tag{17}$$

where | is the logical operation OR and γ is the ratio between HROI height and its width. To carry this out, the value of γ was set to 1.2 according to the method proposed by (Asaari and Suandi 2010).

Accordingly, the Eq. 17 works on updating the height parameter of segmented hand region that could be denoted to arm region. This happened when height is less than the width of segmented hand region or greater than the ratio between segmented hand region height and its

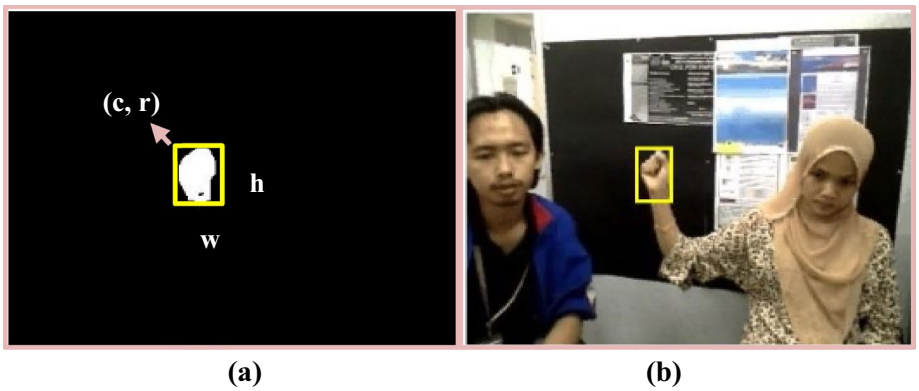


Fig. 9 The illustration of hand region of interest HROI localisations

width by executing Eq. 17 where the unwanted part representing arm region will be discarded (Fig. 10).

4 Results and discussion

All experiments were performed using Matlab 9.0 on an Intel Core i7 processor with 8 GB RAM on Windows 10 operating system. All video sequences were taken from the Intelligent Biometric Group Hand Tracking (IBGHT) database (Asaari et al. 2012). Basically, the IBGHT database consists of 60 video sequences containing a total of 15,554 RGB colour images with annotated ground truth data. These video sequences are ranging from an easy indoor tracking scene to extremely high challenging outdoor scenarios. The general setting of video acquisition focuses on the upper part of the subject's body at some relative cameras to subject (CS) distance. This dataset was divided into two categories and four parts namely Dataset #1(Part1–1, Part1–2) and Dataset #2(Part2–1, Part2–2). In Dataset #1, there are a total of 16 video sequences performed by the same actor and recorded in indoor environment. In the second category (Dataset #2), there are a total of 44 video sequences performed by a number of different actors and includes both indoor and outdoor. The complexities in Dataset #2 are designed to include more extremely challenging scenes than the Dataset #1. The video sequences of IBGHT dataset were recorded so that hand can move in trajectory manner simulating real world challenges involving complex environment, various lighting conditions (indoor and outdoor), independent users and partial occlusion. More information and details regarding every video sequence in the IBGHT dataset can be found in (Asaari et al. 2012).

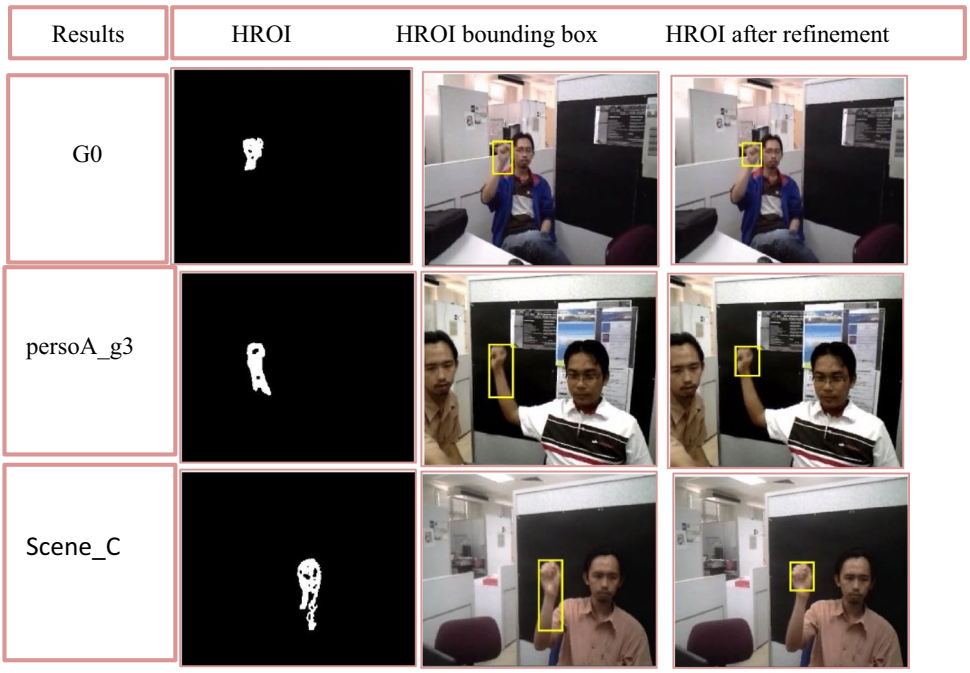


Fig. 10 Removing arm region

The performance of dynamic hand gesture segmentation method was evaluated using quantitative and qualitative measurement to confirm the accuracy of the proposed segmentation approach. According to Asaari et al. (2014), the segmentation accuracy rate was calculated as the ratio of the number of successfully segmented frames to the total number of files in the IBGHT to secure quantitative measurement with the dataset given in the following equation (Eq. 18).

Segmentation accuracy

$$= \frac{\sum \begin{matrix} Number\ of\ videos\ that\ sucess\ hand \\ gesture\ segmentation \end{matrix}}{total\ number\ of\ video\ files} \times 100. \quad (18)$$

To implement Eq. 18, the centre position of every segmented hand region in the x and y direction was calculated. To be used within Eq. 18, the hand region is considered accurately segmented if the distance between its centre location in the x and y directions of the manually labelled "ground truth" data falls within the 9×9 neighbourhood.

The segmentation results are depicted in Table 1. It is evident from the table that the proposed segmentation method has managed to offer promising segmentation accuracy. Using the Entropy filter, FMM and range filter over a mixture of extracted skin, motion, skin moving and contour features contributed towards gaining good segmentation outcomes.

As a qualitative measurement, the results of proposed hand gesture segmentation algorithm were compared with that of previous methods Asaari et al. (2014) based on the average accuracy. The experiment was performed on video sequences of Dataset#1 and Dataset#2 from the IBGHT dataset. The comparison results are illustrated in Fig. 11. Meanwhile, the segmentation results are shown in Figs. 9 and 10.

As observed in Fig. 11, on an average, the comparison results presented that the incidental flaws in the hand gesture segmentation algorithm of Asaari et al. (2014) have been successfully reduced by the proposed hand gesture segmentation algorithm. Consequently, it also managed to optimise the segmentation algorithm of Asaari et al. (2014) with a very convincing segmentation rate and accuracy (Table 2).

As shown in Fig. 12, the segmentation method suggested in this study has better performance and managed to improve the poor segmentation of the Asaari et al. (2014) method in G0, G1, g6, and G8 video sequences in

Table 1 Experimental results upon hand gesture segmentation

File name	The accuracy (%)
Dataset#1 (Part1–1)	100
Dataset#1 (Part1–2)	96.66
Dataset#2 (Part2–1)	100
Dataset#1 (Part2–2)	97.2222

Table 2 Comparative outcomes based on average accuracy rate

File name	Segmentation rate of proposed method (%)	Segmentation rate of Asaari et al. (2014) method (%)
Dataset#1 (Part1–1)	100	62
Dataset#1 (Part1–2)	96.66	78
Dataset#2 (Part2–1)	100	61.25
Dataset#1 (Part2–2)	97.2222	55.55
Average	98%	64.2%

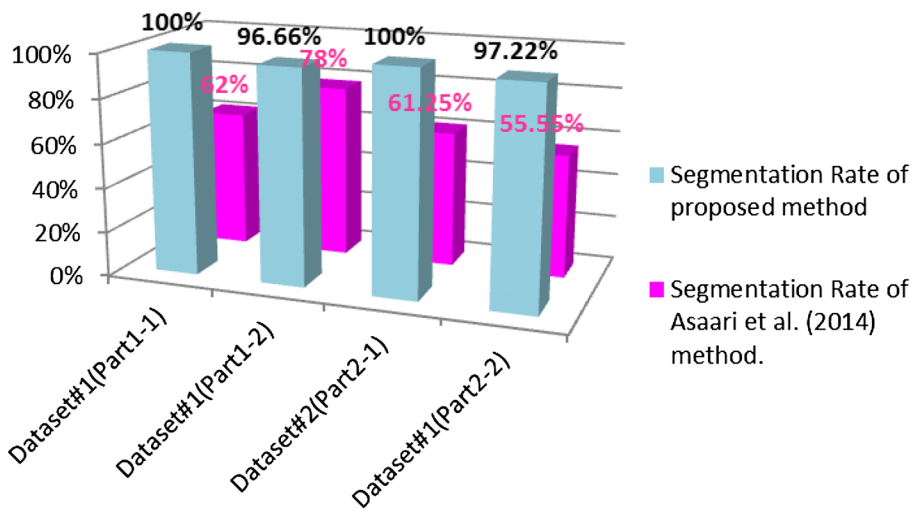


Fig. 11 Comparison results based on average segmentation rate for the dynamic hand gesture segmentation on the video sequences of the IBGHT dataset

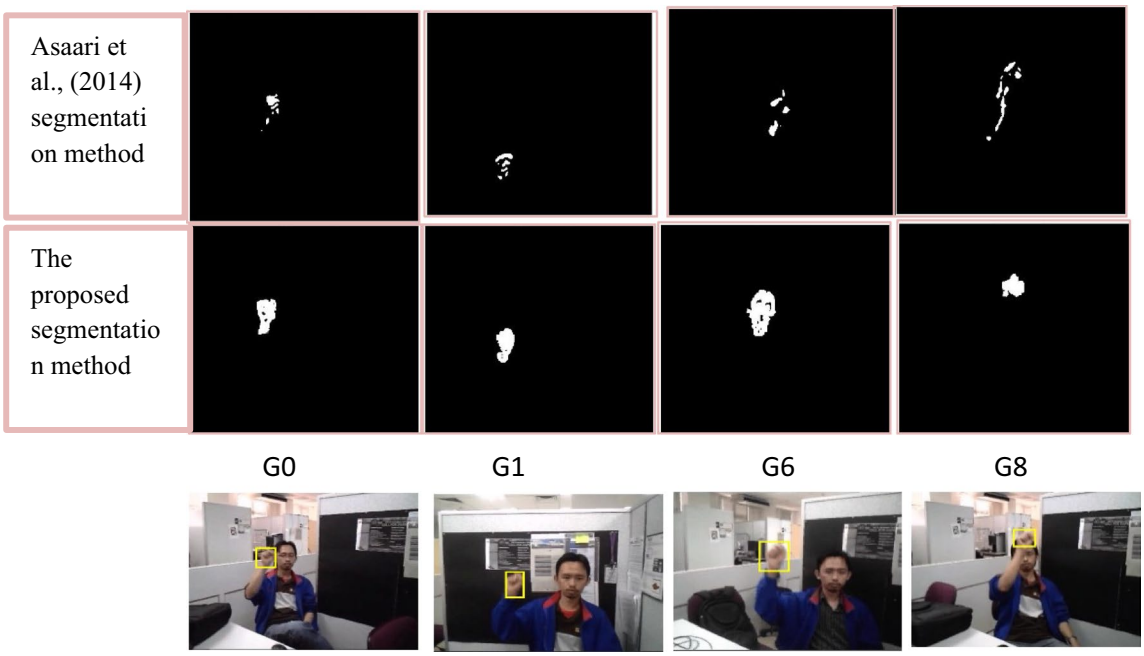


Fig. 12 The comparison results of hand gesture segmentation for Dataset#1 (Part1–1)

Dataset#1 (part1–1) despite the cluttered background, use of either short or long sleeve, partial occlusion with the face region and hand appearance variations that arise due to different kinds of movement trajectory.

Figure 13 illustrates the results of segmentation in Scene_C, Scene_D and Scene_E of Dataset#1 (Part1–2), respectively. In the situation of Scene_C, the proposed segmentation method managed to segment hand region despite the erratic motion of the hand, which speedily intersects with the skin colour, shirt involved and the face. In Scene_D and Scene_E, the segmentation approaches managed to segment and detect the hand region under uncommon illumination effects and surprisingly, it happens even when the two hands are moving together with non-target hand moving in a small manner. In this regard, the thresholding process based on certain threshold value proposed

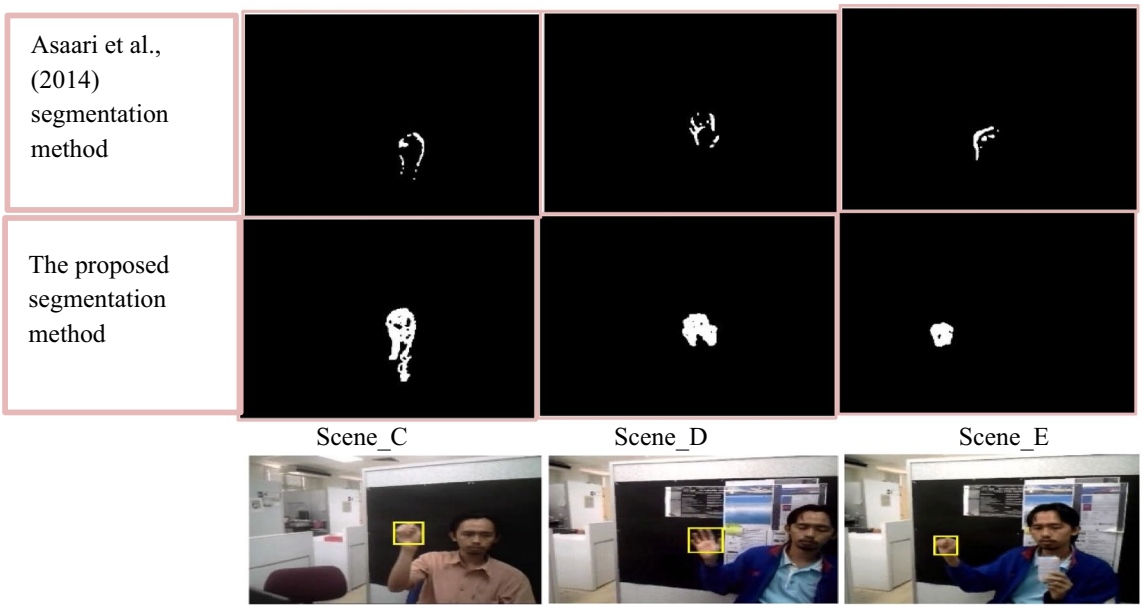


Fig. 13 Comparison results of hand gesture segmentation for Dataset#1 (Part1–2)

for motion feature segmentation algorithm has helped in discarding unrelated moving objects including another moving hand as in scene E. Besides, the proposed modified visual features and the proposed formula based features fusion have led to better and accurate results for dynamic hand gesture segmentation under different and challenging scenarios as shown in the experimental results of this study.

As depicted in Fig. 14, experiment results displayed different indoor and outdoor real life scenarios. For example, the proposed segmentation method successfully managed to segment and map the targeted hand region in indoor_2 behind the static object occlusion and in the outdoor environment where over-exposed phenomena occurs due to illumination variations and skin features becomes hard to resolve. Here, the correct segmentation of the skin features was brought by the proposed skin extraction scheme. Additionally, the optimisation by the Entropy texture analysis filter and the Range filter was applied to each skin moving and contour features, which contributed in extracting small portions of pixel inside the hand region of interest. Consequently, this leads to the strengthening of segmented feature and accurately segmented hand gesture.

Consequently, based on the above comparison results and discussions, the hand gesture segmentation method introduced by this study has outperformed the performance of AKFIE method (Asaari et al. 2014) as depicted in Fig. 15.

5 Conclusion

In this paper, a method for dynamic hand gesture segmentation was proposed. The hypothesised approach aimed at overcoming the problem of non-accurate and poor hand gesture segmentation where the hand region is not integral due to complicated environment, partial occlusion and light effects. Hence, to obtain accurate hand gesture segmentation to the moved hand gesture region, this paper proposed fast marching method and mixture of four modified and low cost visual features segmentation procedures involving

Fig. 14 Experimental results for different indoor and outdoor real life scenes of Dataset#2 (Part2–1)

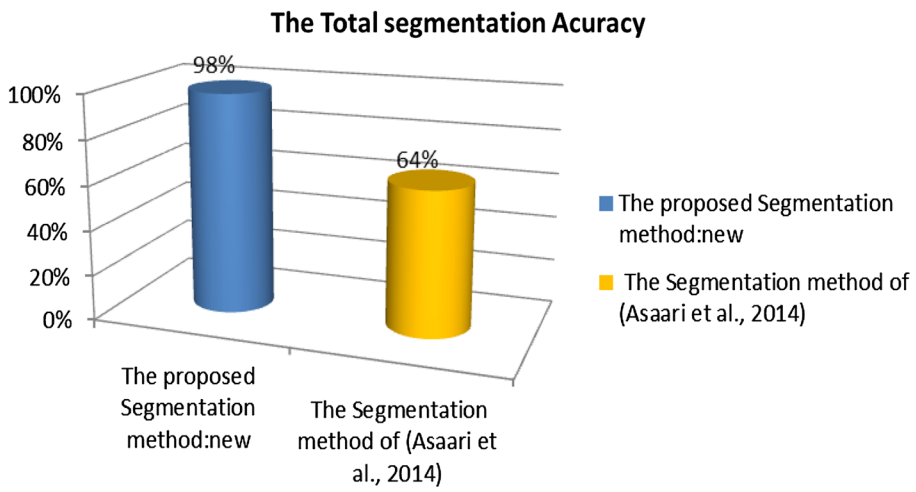


Fig. 15 Comparison of the proposed algorithm with another state of-the-art method on the IBGHT dataset

skin, motion, skin moving and contour feature extracting procedures. Quantitative and qualitative evolution measurements have been conducted based on video files of IBGHT dataset achieving great enhancement in hand gesture segmentation in comparison with AKFIE method and segmentation accuracy by 98%.

Acknowledgements The authors would like to acknowledge the COMPSE 2016 for supporting this work.

References

Asaari MSM, Suandi SA (2010) Hand gesture tracking system using Adaptive Kalman Filter. In: 2010 10th International Conference on Intelligent Systems Design and Applications. IEEE, pp 166–171

Asaari MSM, Rosdi BA, Suandi SA (2012) Intelligent biometric group hand tracking (IBGHT) database for visual hand tracking research and development. Multimed Tools Appl 70(3):1869–1898

Asaari MSM, Rosdi BA, Suandi SA (2014) Adaptive Kalman filter incorporated eigenhand (AKFIE) for real-time hand tracking system. Multimed Tools Appl 74(21):9231–9257

Bailey DG, Hodgson RM (1985) Range filters: localintensity subrange filters and their properties. Image Vis Comput 3(3):99–110

Bernardos AM, Sánchez JM, Portillo JI, Wang X, Besada JA, Casar JR (2016) Design and deployment of a contactless hand-shape identification system for smart spaces. J Ambient Intell Humaniz Comput 7(3):357–370

Bhuyan MK, Kumar DA, MacDorman KF, Iwahori Y (2014) A novel set of features for continuous hand gesture recognition. J Multimodal User Interfaces 8(4):333–343

Chidananda P, Srinivas P, Manikantan K, Ramachandran S (2015) Entropy-cum-hough-transform-based ear detection using ellipsoid particle swarm optimization. Mach Vis Appl 26(2–3):185–203

Choudhury A, Talukdar AK, Sarma KK (2015) A Review on vision-based hand gesture recognition and applications. In: Intelligent Applications for Heterogeneous System Modeling and Design. IGI Global, pp 256–281

De Marsico M, Levialdi S, Nappi M, Ricciardi S (2014) FIGI: floating interface for gesture–based interaction. J Ambient Intell Humaniz Comput 5(4):511–524

Hamahashi S, Onami S, Kitano H (2008) U.S. Patent No. 7,460,702. U.S. Patent and Trademark Office, Washington, DC

Han J, Awad G, Sutherland A (2009) Modelling and segmenting subunits for sign language recognition based on hand motion analysis. Pattern Recognit Lett 30(6):623–633

Kelly D, McDonald J, Markham C (2010) A person independent system for recognition of hand postures used in sign language. Pattern Recogn Lett 31:1359–1368

Lienhart R, Maydt J (2002) An extended set of haar-like features for rapid object detection. In: Image Processing. 2002. Proceedings. 2002 International Conference on vol 1. IEEE, pp I-900

Mazumdar D, Nayak MK, Talukdar AK (2015) Adaptive hand segmentation and tracking for application in continuous hand gesture recognition. In: Recent Trends in Intelligent and Emerging Systems. Springer India, pp 115–124

Monneau (2010) Introduction to the Fast Marching Method. Technical report, Centre International de Math´ematiques Pures et Appliqu´es

Oz C, Leu MC (2011) American Sign Language word recognition with a sensory glove using artificial neural networks. Eng Appl Artif Intell 24:1204–1213

Park J, Lee Y, Ko H (2009) Dynamic time warping based identification using gabor feature of adaptive motion model for walking humans. Int J Control Autom Syst 7(5):817–823

Rautaray SS, Agrawal A (2012) Vision based hand gesture recognition for human computer interaction: a survey. Artif Intell Rev 43(1):1–54

Ren Z, Yuan J, Meng J, Zhang Z (2013) Robust part-based hand gesture recognition using kinect sensor. IEEE Trans Multimed 15(5):1110–1120

Sethian JA (1996) A fast marching level set method for monotonically advancing fronts. Proc Natl Acad Sci 93(4):1591–1595

Sethian JA (1999) Level set methods and fast marching methods: evolving interfaces in computational geometry, fluid mechanics, computer vision, and materials science, vol 3. Cambridge University Press, Cambridge

Sgouropoulos K, Stergiopoulou E, Papamarkos N (2014) A dynamic gesture and posture recognition system. J Intell Robot Syst 76(2):283–296

Simões WC, Barboza RDS, de Jr Lucena VF, Lins RD (2015) A fast and accurate algorithm for detecting and tracking moving hand gestures. In: Developments in Medical Image Processing and Computational Vision. Springer International Publishing, pp 335–353

Stergiopoulou E, Sgouropoulos K, Nikolaou N, Papamarkos N, Mitianoudis N (2014) Real time hand detection in a complex background. Eng Appl Artif Intell 35:54–70

Vafadar M, Behrad A (2014) A vision based system for communicating in virtual reality environments by recognizing human hand gestures. Multimed Tools Appl 74(18):7515–7535

Viola P, Jones M (2001) Rapid object detection using a boosted cascade of simple features. In: Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on vol 1. IEEE, pp I-511

Wang Z, Zhang Z, Wang F, Sun Y (2014) Vision-based hand gesture interaction using particle filter, principle component analysis and transition network. J Inf Comput Sci 11(4):1037–1045

Yeo HS, Lee BG, Lim H (2013) Hand tracking and gesture recognition system for human–computer interaction using low-cost hardware. Multimed Tools Appl 74(8):2687–2715

Yun L, Lifeng Z, Shujun Z (2012) A hand gesture recognition method based on multi-feature fusion and template matching. Procedia Eng 29:1678–1684

Zabulis X, Baltzakis H, Argyros A (2009) Vision-based hand gesture recognition for human–computer interaction. The Universal Access Handbook. LEA, pp 34-1

Zhao Z-Y, Gao W-L, Zhu M-M, Yu L (2012) A vision based method to distinguish and recognize static and dynamic gesture. Procedia Eng 29:3065–3069

Zhu F, Tian J (2003) Modified fast marching and level set method for medical image segmentation. J X-Ray Sci Technol 11(4):193–204