BRISK Local Corner Keypoints based Hand Position Estimation for Hand Gesture Trajectory Tracking Method

Article Info

Received:3/1/2017
Accepted: 7/2/2017
Published online: 1/3/2017

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

[1]Faculty of Computer Science and Information Technology
University Putra Malaysia

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

Eman.Thabt.H@gmail.com, fatimahk@upm.edu.my,
psuhaiza@upm.edu.my, razaliy@upm.edu.my

ISSN 2231-8844

ABSTRACT

In recent years, dynamic hand gesture tracking has attracted the attention of many researchers. However, 2D-vision-based approaches have motivated a sizeable body of research and the rapid improvement of web cameras has facilitated their application to hand-held (portable) appliances, it is a critical process which has encountered many difficulties in the real world. These problems include variations in hand appearance, complicated background as well as confusion and drifting problems caused by the need to look for the hand in the total scene. To increase the accuracy of moving hand position estimation, this paper proposed a novel algorithm to track moving hand positions by searching corner keypoints of the hand last position, assuming that the difference between current and previous hand positions is little since the aim is to track the trajectory of hand movement. The proposed method uses corner keypoints of BRISK and minimum eigenvalue techniques extracted from last segmented region of hand to create searching windows and estimate hand region position on current frame image of video sequences. The experimental results revealed about correct estimation to hand position in the video sequences with non-linear hand movement in the challenging environment

Key words: **hand gesture tracking, BRISK keypoints, skin color segmentation, motion segmentation, minimum eigenvalue features.**

1. Introduction

Hand gestures play an important role in human-computer interaction, providing an interactive communication channel for different application domains such as Robotics interaction, sign language, medical systems, augmented reality, virtual reality, device control, and smart homes (Rautaray et al., 2015).

Hand gestures are categorized into static or local motion and dynamic or global motion gestures. Static hand gestures concern the hand fingers movements that form different postures or shapes. On the other hand, dynamic hand gestures consider the static hand gestures in spatial-temporal domain since the hand region moves from one point to another point in time under different backgrounds and environmental circumstances. Dynamic hand gesture recognizes the relationship between the hand positions on successive video frames that form or draw hand trajectory (Rautaray et al., 2015; Dixit et al.,2015; Dardas et al., 2012). In addition, a hand gesture recognition system could be used to facilitate a lot of tasks in our daily life. It can be applied to be a part of user- machine interface application replacing the use of a mouse, keyboard, and touchscreen with a more interactive and intuitive controlling tool. Basically, a dynamic hand gesture recognition system consists of hand gesture segmentation, gesture tracking, feature extraction, and gesture recognition. Hand gesture tracking is regarded as a median mean between hand feature extraction and hand recognition. Dynamic hand gesture tracking is considered as an essential step for any dynamic hand gesture classification system. This is because hand gesture tracking is a comprehensive process demands localizing hand positions during its movement in the air (trajectory of hand) despite a huge number of challenges that exist in the real world (Kılıboz et al., 2015; Kumar et al., 2010).

In order to cope with these problems, the literature has divided hand gesture tracking approaches into two main categories: data-glove-based approaches and vision-based approaches. Data-glove-based approaches demand the user to wear gloves equipped with wire sensors that facilitate hand motion detection. Although data-glove-based approaches give accurate results, they consider far away from intuitiveness and natural communication. Vision-based approaches that deal with camera vision provide more intuitive and natural interaction (Asaari et al., 2015; Kumar et al., 2010; Rautaray et al., 2015; Joshi et al., 2014).

2. Review of literature

Dynamic hand gesture tracking is a significant step to develop human – computer interface (HCI), where a computer can detect and track the characteristics of hand motion over time as input and then translate or classify these gained features into certain commands as outputs.

However, hand gesture tracking employing webcams despite their nature and the lower cost is critical and not an easy task as it encounters numerous challenges. Hence, many approaches have been proposed in order to cope with visual hand gesture tracking problems. To provide a review of the challenges and work done regarding hand gesture tracking approaches, Guo et al. (2013) Proposed sufficient and fast hand localization and tracking approach for hand gesture classification system. The visual acquisition information used to recognize the trajectory of hand movements. Hand detection and tracking is implemented utilizing integral HOG and a new multi stages SVM classifier. Skin segmentation is initial step of multi stage classifier, followed by filtering out utmost detection or localization window. The proposed approach uses 5 kinds of background divisions which are arm, face, fist, and complex and simple background. Each kind of background respectively builds a recognizer with hand. The window of detection is considers a hand only if managed to pass entire 5 SVM classifiers. The multi stages classifier speed is influenced through lost order of weak classifier. Camshift algorithm lastly applied in order to track hand in hue (H) channel of HSV color space when detection is unsuccessful. However, the success rate that achieved by detection method is 93.1% and 0.33% false –positive rate (FPR).

Zou et al.(2013) aimed to overcome the problem of the target hand being influenced by variations of the surrounding environment such as sudden change in illumination conditions (becoming strong, or weak), or overlapping with non-target objects and becoming so serious that it may result in failed tracking due to drift of the target hand. Thus, their method basically used updating Mean Shift integrated by Kalman filter. For segmentation, skin color detection in YCbCr and motion detection using background subtraction are used to segment the hand region in the binary image. The mean-shift algorithm is utilized to track and update the gesture model. The Kalman filter is applied to outcomes of Mean Shift model to estimate the results of gesture tracking and reduce the drifting problem caused by clutter environment. However, this algorithm is still inaccurate and needs further enhancement since the algorithm is degraded by light changes under shadow and refuge.

Dixit and Agrawal (2015) introduced real-time detection and tracking method using Adaboost algorithm based on Viola–Jones detector and Haar-like features classifier extended by Camshift algorithm and Kalman filter. Douglas Peucker method is used to extract the trajectory feature by centre position extraction. However, Viola–Jones algorithm shows good results only when hands gesture direction is vertical while the hand is missed in other directions. Besides, it is sensitive to differences in illumination conditions in addition to changes in hand appearance caused by rotation for example. Camshfit method is limited when there are objects with the same skin color of tracked hand moving in the background. The method is also sensitive to light changes. The Kalman filter is degraded under non-linear motion and even though this defect can be rectified by using a centre position feature based tracking, it may still not be very effective. Also, it may fail or drift due to the wrong estimation caused by the defect in either the Adaboost algorithm or Camshfit method.

Premaratne et al. (2015) utilized hand gesture segmentation algorithm uses skin color detection in Cb-Cr chrominance components and segmentation depending on two schemes. The first scheme utilizes threshold values that consider bright lighting conditions while the other adopts threshold values that consider fluorescent lighting conditions. After skin color has been segmented, morphology operation is added to remove small blobs from the background. As hand region is being segmented and detected, the tracking process is done relying on the extraction of the center coordinates of each segmented region overall video sequences. This project worked on a subset of 16 English letters of the alphabet. Although the skin color is invariant to hand appearance changes, using skin color alone may make this method encounter difficulty regarding hand segmentation that comes from similar skin color in the background since the tracking method depends on extracting the hand from the entire view.

Leutenegger et al. (2011) put forward binary descriptor that is both rotation and scale invariant. It is called Binary robust invariant scalable keypoints (BRISK). The main characteristics of BRISK are that, in each scale pyramid octave, a corner detection method called Features from Accelerated Segment Test (FAST), used it to detect the keypoints, instead of simply locating local extreme values as in SURF and SIFT. That makes the keypoints selection more efficient in BRISK.

Shi (1994) developed a good feature tracking method known as minimum eigenvalue algorithm return a corner keypoints object. The object holds information about the feature keypoints selected in a 2-D grayscale input image. The minimum eigenvalue algorithm based tracker is features monitoring method that can detect occlusion as well as it can work under affine image transformation.

Nevertheless, Kalman filtering is an optimum estimation technique for linear dynamic systems with Gaussian noise distributions. However, the performance of Kalman filter limits to nonlinear object tracking such as dynamic hand gesture tracking. In order to overcome such limitations, a kind of adaptive Kalman filter (Asaari et al., 2015) is proposed via utilizing inaccurate parameters tuning and measurement noise covariance matrices; with help of inefficient observed database on ROI based tracker algorithm that estimates hand position using raw features of fused skin-motion feature. ROI algorithm is being integrated by adaptive Kalman filter. In fact, the proposed framework is still suffering clear weakness and its performance degraded to cope with the problem of non-linear hand motion so that the tracker fail and drift. Tracking hand object in the spatial-temporal domain is still far off optimal solution despite great efforts that have been made (Stergiopoulou et al., 2014; Asaari et al., 2015; Xu et al., 2012; Yeo et al., 2015).

Thus, to increase the accuracy of moving hand position estimation, this study proposes Local Corner keypoints based tracker algorithm (LCKPs). This algorithm based on last segmentation result to hand region and its corner keypoint's coordinates positions. The principle work of our proposed algorithm commences after getting hand object detected and segmented for the first time. We benefit from the detection stage parameters that represent bound box width and height, as well as the binary segmented area of the hand, which is used as input for feature detection. BRISK (Leutenegger et al., 2011) and minimum eigenvalue (Shi, 1994) are used to extract coordinates positions of corner keypoints. Candidate samples such as searching windows will extract from the current video frames but from corner keypoints locations of previously segmented hand area of the previous frame. These candidate samples are used to search for the hand region on the current video frame. After that, in order to locate the true and accurate location or candidate sample that represents the target hand region, we use a set of conditions or rules that are based on measuring the mean density of skin color, motion and combination of two skin-motion information inside each searching window (candidate sample), as well as orientation cue in comparison with the last segmented region of hand using hamming distance.

This technique concentrates on the search for a single bar hand within video sequences captured by webcam. This study employs image processing techniques.

3. Methodology

3.1 Overview of the proposed method

The proposed algorithm employs corner keypoints positions extracted from the last location of hand by using a combination of BRISK (Leutenegger et al., 2011) and minimum eigenvalues (Shi, 1994) corner keypoints detector techniques. Fig.1 shows the sequence of steps for the proposed technique. When hand region is located on the current video frame, the hand region is segmented from depending on searching widow, to be used in next sequence

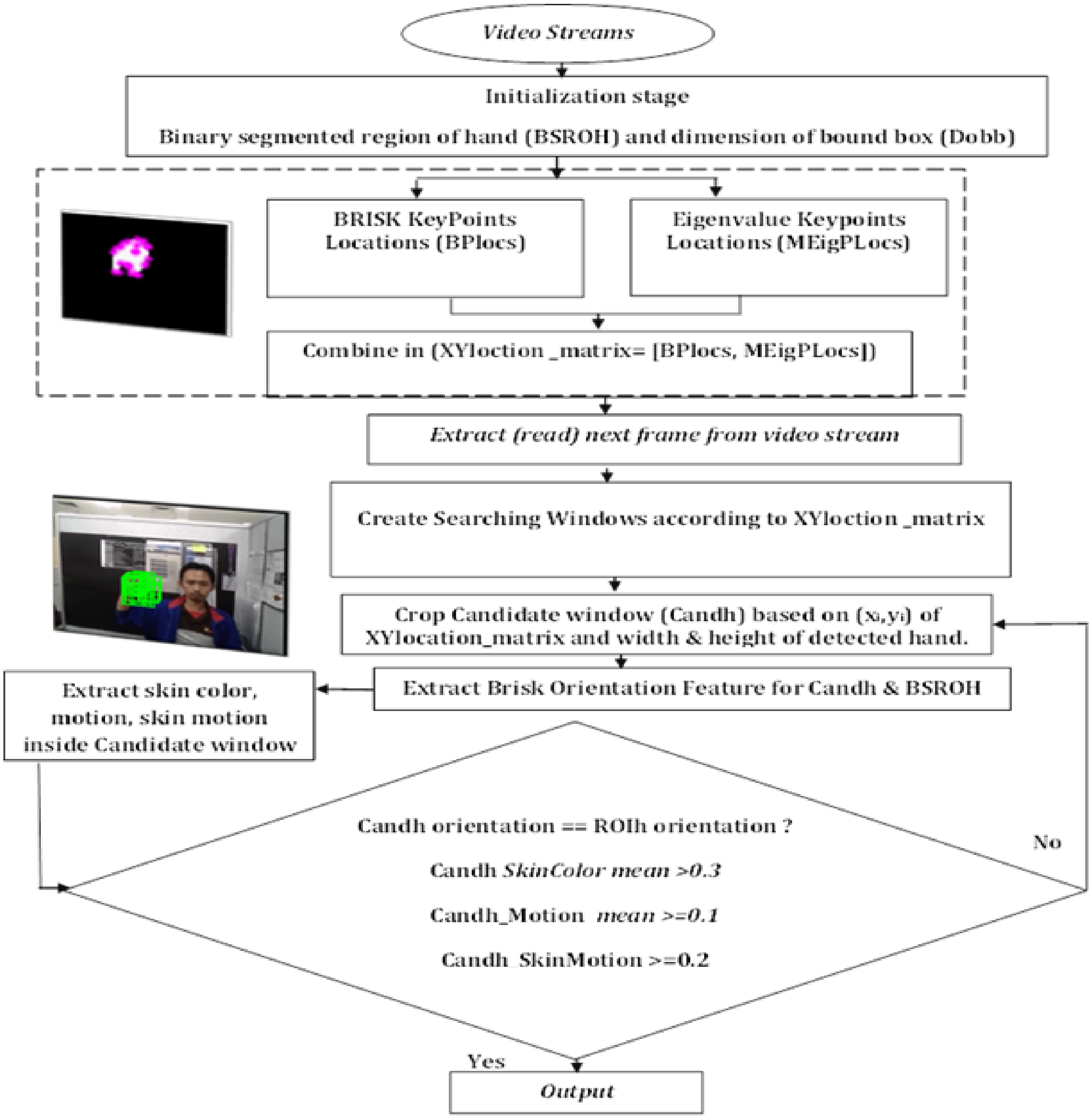

Video Streams
Initialization stage
Binary segmented region of hand (BSROH) and dimension of bound box (Dobb)
BRISK KeyPoints Locations (BPlocs)
Eigenvalue Keypoints Locations (MEigPLocs)
Combine in (XYloction _matrix= [BPlocs, MEigPLocs])
Extract (read) next frame from video stream
Create Searching Windows according to XYloction _matrix
Crop Candidate window (Candh) based on (xi,yi) of XYlocation_matrix and width & height of detected hand.
Extract Brisk Orientation Feature for Candh & BSROH
Extract skin color, motion, skin motion inside Candidate window
Candh orientation == ROIh orientation ?
Candh SkinColor mean >0.3
Candh_Motion mean >=0.1
Candh SkinMotion >=0.2
No
Yes
Output

Fig.1. Block diagram of proposed algorithm

3.2 Skin Color Segmentation

For skin color representation, YCbCr color space is one of the most common color spaces employed for skin color detection.

In this study, we use our proposed algorithm for skin feature segmentation in videos (Eman et al., 2017). Our proposed algorithm works on thresholding values extracted from Cb and Cr chrominance components that represent color components and eliminating illumination effects that concentrate on the Y component (Asaari et al., 2015; Viola & Jones, 2001; Joshi et al., 2014; Yeo et al., 2015). In order to get rid of skin color flaws caused by clutter background with similar skin color objects, users with different skin color range (independent users) and lighting changes, this module takes advantage and employs thresholding technique to combine dynamic (online-training) and static (offline- training) thresholding schemes. Skin color areas are obtained as in (Eq. (1). In the online – training scheme, we detect face and nose regions via Viola-Jones algorithm (Viola & Jones, 2001; Shugo et al., 2008), then Cb and Cr range values are automatically obtained by maximum and minimum range values taken from the pixels-nose region. This procedure implements for only once time to reduce computation time since Viola-Jones algorithm consumes considerable time. As well, the computation power will be added from using online – training procedure for every frame- the image of video sequences within the tracking hand gesture stage. So, in that step, we keep a copy of the extracted thresholding values parameters to be used in the tracking stage as both mincr and maxcr, mincb and maxcb. Accordingly, the online scheme is implemented only once.

The online- training scheme gives satisfactory results but under low illumination condition and out of plane face rotation, the Viola-Jones algorithm may fail to detect face or nose regions and this degrades the performance of skin color detection algorithm. Therefore, to overcome this problem, we have adopted the offline- training scheme as an alternative choice, which depends on static threshold obtained manually out of a number of skin samples to obtain skin areas.

$$\mathbf{St}=\begin{cases}\mathbf{1}, & \mathbf{if\ Cb} \in \mathbf{Cb_{rang}}\ \mathbf{and\ Cr} \in \mathbf{Cr_{rang}} \\ \mathbf{0}, & \mathbf{if\ otherwise}\end{cases} \qquad (\mathbf{1})$$

Where St is a segmented skin color fearure**,** Cr_range = (134, 164) and Cb_range = (100, 126), are the thresholds range values of offline training scheme.

However, this proposed skin color detection scheme that is based on switching thresholding scheme perfectly eliminates the drawbacks of skin color detection and saves the skin color detection algorithm from any failure. More details information regarding skin feature segmentation algorithm are explained in (Eman et al., 2017).

3.3 Motion Segmentation

Motion cue has been used in a few approaches for hand gesture segmentation and detection because it is sensitive to shadow, blurring and many moving objects in the background (Rautaray et al., 2015, Wang et al., 2014; Mazumdar et al., 2015). However, motion cue can be efficiently applied to hand detection and segmentation by makings it work in association with other extracted features such as skin color and orientation features (Stergiopoulou et al., 2014; Yao & Li, 2013).

In the literature, most of the motion detection techniques use either frame differencing method or background subtraction method (Rautaray et al., 2015). In this study, we employ frame differencing technique in our motion detection algorithm to extract motion areas between two successive frames (Stergiopoulou et al., 2014; Asaari et al., 2015). Since frame difference has fast detection speed and ability to detect sudden movement (Stergiopoulou et al., 2014).

The frame differencing method may be very sensitive to changes of lighting. Besides that it has a problem avoiding the detection of small blobs and unrelated articles moving in the background. One way to avoid such a problem is by including the thresholding operation technique (Gupta & Kulkarni, 2008). Employing a proper thresholding technique may lead to accurate motion detection algorithm so that the undesired disturbance can be reduced. Suppose that $\mathrm{fram_t}$ is RGB image at time (t) and $\mathrm{fram_{t+1}}$ is the RGB image at time (t+1) of video sequence. So, $\mathrm{Framdif_t}$ is the difference between $\mathrm{fram_t}$ and $\mathrm{fram_{t+1}}$as defined in Eq. (2). The $\mathrm{fram_t}$ and $\mathrm{fram_{t+1}}$are firstly converted to gray level before they are assigned to performing frame differencing operation. By enforcing threshold technique to obtain satisfying results, the resulting image $\mathrm{Framdif_t}$ considers once $\mathrm{mean(Framdif)}$ is grater or equal to threshold value which obtained through experiments. Then a binary image is obtained by converting $\mathrm{Framdif_t}$ into binary $\mathrm{Framdifb_t}$ applying Eq. (3). A threshold technique is used to calculate the value of the threshold (Thr.). In case the measured threshold value (Thr) is greater than 1, then the Thr. value assigned to 0.2 is determined experimentally. . Binary image obtained according to Eq. (4).

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

$$\mathrm{Thr} = \begin{cases} 0.05 * \dfrac{\mathrm{mean(Framdif)}}{\mathrm{size(fram_t)}} \le 1 \\ 0.2, \quad \text{otherwise} \end{cases} \tag{3}$$

$$\mathrm{Framdifb_t} = \begin{cases} 1, \text{if } \mathrm{Framdif_t} \ge \mathrm{Thr} \\ 0, \text{otherwise} \end{cases} \tag{4}$$

However, for hand gesture segmentation, we use a technique mainly based on a combination of skin color, motion information, and texture features by utilizing logical operation.

$$BSROH = EntrpyFilter(S_t \& Framdifb_t, [9\ 9]) \tag{4.18}$$

Where, $BSROH$ is a binary segmented region of hand gesture.

3.4 Local corner keypoints based Tracking

This study uses binary features extraction techniques based on integrating of binary robust invariant scalable keypoints (BRISKs) (Leutenegger et al., 2011) and minimum eigenvalues keypoints (Shi, 1994). The reason for choosing them is their simplicity of implementation and their high performance upon the binary region of hand that our experiments proved, such as their robustness against rotation, great scale variation, translation, and hand postures change.

In this paper, we propose Local Corner Keypoints (LCKPs) based tracking algorithm that can reliably and efficiently estimate our target hand object position inside every image frame of video sequences. This proposed model can effectively work for hand object tracking method. In fact, our work has inspired from other developed techniques for face and human body tracking and recognition methods (Heinly et al., 2012; Mair et al., 2010). These methods are based on dividing the face or human body target region into many halves and trying to estimate face region through feature matching techniques. However, these techniques may not be feasible for hand region tracking because the hand is a flexible object with appearance variations and is also small in size compared to face or human body area. Therefore, our experiments led to uses corner keypoints locations of BRISK and Minimum Eigen techniques, extracted from previously segmented hand area to estimate and localize hand region on the current frame. Our estimation is that the hand moves in a trajectory manner and its motion speed is not that great.

However, in order to estimate hand object region, after initially applying hand segmentation and detection method, we get BSROH, which is binary segmented region of hand and Dobb, which is dimension of bound box drawn around detected hand region, represented in [x, y, w, h], where (x, y) is upper left corner point of Dobb and (w, h) is width and height of Dobb.

$$\mathrm{Dobb} = \{x_j, y_j, w, h\} \quad , \text{where } j = 1,2, \ldots\ldots n \tag{5}$$

Let f_1 and f_2 be the BRISK and Minimum Eigen corner keypoints extracted based on [20, 23] from binary region of hand, let F be a matrix that combines both f_1 and f_2 , and also contains x and y coordinates locations of extracted keypoints.

$$F = \{f_{1,}, f_2\} \tag{6}$$

In general, f is given as follows:

$$f = \{\ (x_1, y_1)\ (x_2, y_2)\ (x_3, y_3) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (x_n, y_n)\} \tag{7}$$

Where n is the total number of extracted corner points.

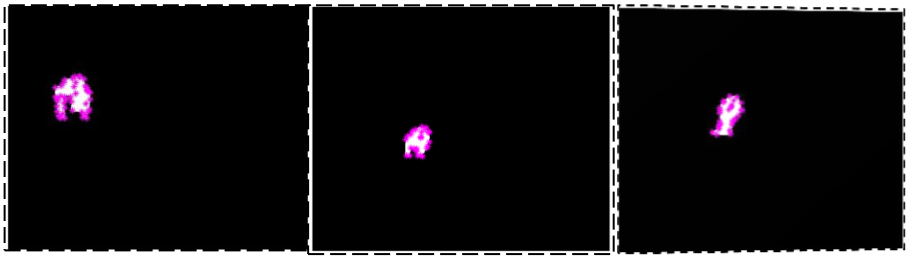


Fig.2. Binary corner features upon segmenting hand of fram_{t-1}

Then, we proceed to the next image of the video sequence, to estimate hand location on the current video frame. Based on each coordinate point (x_j, y_j) of F, we create and crop searching windows treated as candidates regions to have hand object of our interest, and named Candh as follows:

$$\text{candh} = \text{crop}(\ \text{frame}_i, \{\text{Dobb}\}) \qquad (8)$$

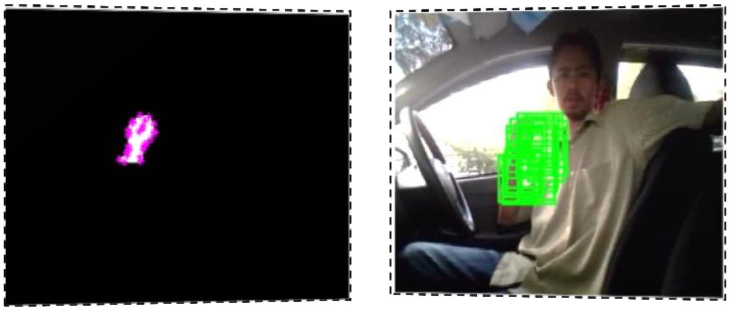


Fig.3. Candidate's window drawn using extracted corner keypoints based on current frame frame_t

In order to decide which candidate window has our target object, a set of criteria and conditions is enforced on each candidate and the candidate window that successfully meets these conditions eventually is identified as the one that has our target. These rules are as follows:

Firstly we resize each candidate to be 32×32, and then we extract BRISK orientation features and compare them to BRISK orientation of previously detected hand region from the last frame according to hamming distance (Norouzi et al., 2012).

$$\boldsymbol{d_{st}} = \frac{\mathbf{1}}{\mathbf{4}}\big||\boldsymbol{h - tanh)Diag(s)v}\big)||_2^2 \qquad (\mathbf{10})$$

Where, $s \in R^q$ is a vector of scaling parameters that control the slope of hyperbolic tangent applied to different bits; Diag(s) is a diagonal matrix with the elements of s on its diagonal. The candidate window will pass this step and proceed to meet the next set of criteria, if $d_{st} \leq 0.99$. In the next step, skin color extracted based on threshold values from the detection phase (those threshold values obtained either online- training or offline- training scheme), and denote them as S_{candh}. After we get the skin color segmented in the candidate window, we calculate mean intensity of skin color S_{mean} as

$$\mathrm{S_{mean}} = \mathrm{mean(S_{candh}), if\ d_{st}} \leq 0.99 \tag{10}$$

So, if the measured mean intensity of skin color ($\mathrm{S_{mean}} \geq 0.3$), the candidate window, passes this step it proceeds to the 3rd set of criteria conditions, where motion information (MF) is segmented inside the candidate window according to two frame differencing method as given below:

$$\mathrm{MF} = |\mathrm{frame_i} - \mathrm{frame_{i-1}}| \tag{11}$$

$$\mathrm{MF_{candh}} = \mathrm{crop(\ MF, \{candh\})} \tag{12}$$

Where:

$\mathrm{MF_{candh}}$ = the extracted motion information inside candidate window.

Then, mean intensity of motion information ($\mathrm{M_{mean}}$) inside the candidate window ($\mathrm{MF_{candh}}$) is calculated. To this end, if ($\mathrm{M_{mean}} \geq 0.1$) then the candidate window proceeds to the 4th and final set of criteria conditions, in which SkinMotion information $\mathrm{SM_{candh}}$ inside the candidate window candh is extracted and mean intensity of SkinMotion is calculated as given below:

$$\mathrm{SM_{candh}} = \mathrm{MF_{candh}} \,\&\, \mathrm{S_{candh}} \tag{13}$$

Entropy filter based on [15] is further applied in order to extract smooth portions inside SkinMotion region and to get precise segmented information. Entropy filter is applied to 7× 7 neighbourhood around the pixel of SkinMotion and compute the statistics for that neighbourhood.

$$\mathrm{SM_{candh}} = \mathrm{entropy\ filter(\ SkinMotion\ )} \tag{14}$$

Mean intensity of SkinMotion information ($\mathrm{M_{SkinMotion}}$) is computed, so that, if($\mathrm{M_{SkinMotion}} \geq 0.2$), then candidate widow has our target hand object (which is the output).

However, once we get our target, we update the width (w) and height (h) of bound box dimension Dbb through compute $|\mathrm{C}|$ and $|\mathrm{R}|$ which are the vectors that involve horizontal and vertical coordinates extracted from white pixels of SkinMoition, so that $\mathrm{w = max|C| - min\,|C|}$, $\mathrm{h = max|R| - min|R|}$, while upper left coordinates of bound box Dbb retain their previous values. After that, the binary hand area of our interest Roihb is segmented from the entire input image of video based on bound box dimension Dbb and skin color information extracted from the entire input image. Finally, segmented hand region $\mathrm{Roih_b}$ at frame (i-1) is updated and replaced by current binary segmented region at frame (i), to be used as reference in next video sequences.

$$\mathrm{Roih_{i-1}} = \mathrm{Roih_i}\ \mathrm{, besides\ Dbb_{i-1}} = \mathrm{Dbb_i}$$

Instead, it is important to remark that if one of the criteria conditions does not work out or degrades, the model immediately skips the reset statements of current iteration to take the next iteration and tries to extract and examine another candidate window till the target hand is located as illustrated in Fig.1 Hence, hand region position is estimated and segmented from the entire input image based on the dimension of candidate widow($\mathrm{Candh} \rightarrow \{x_j, y_j, w, h\}$).

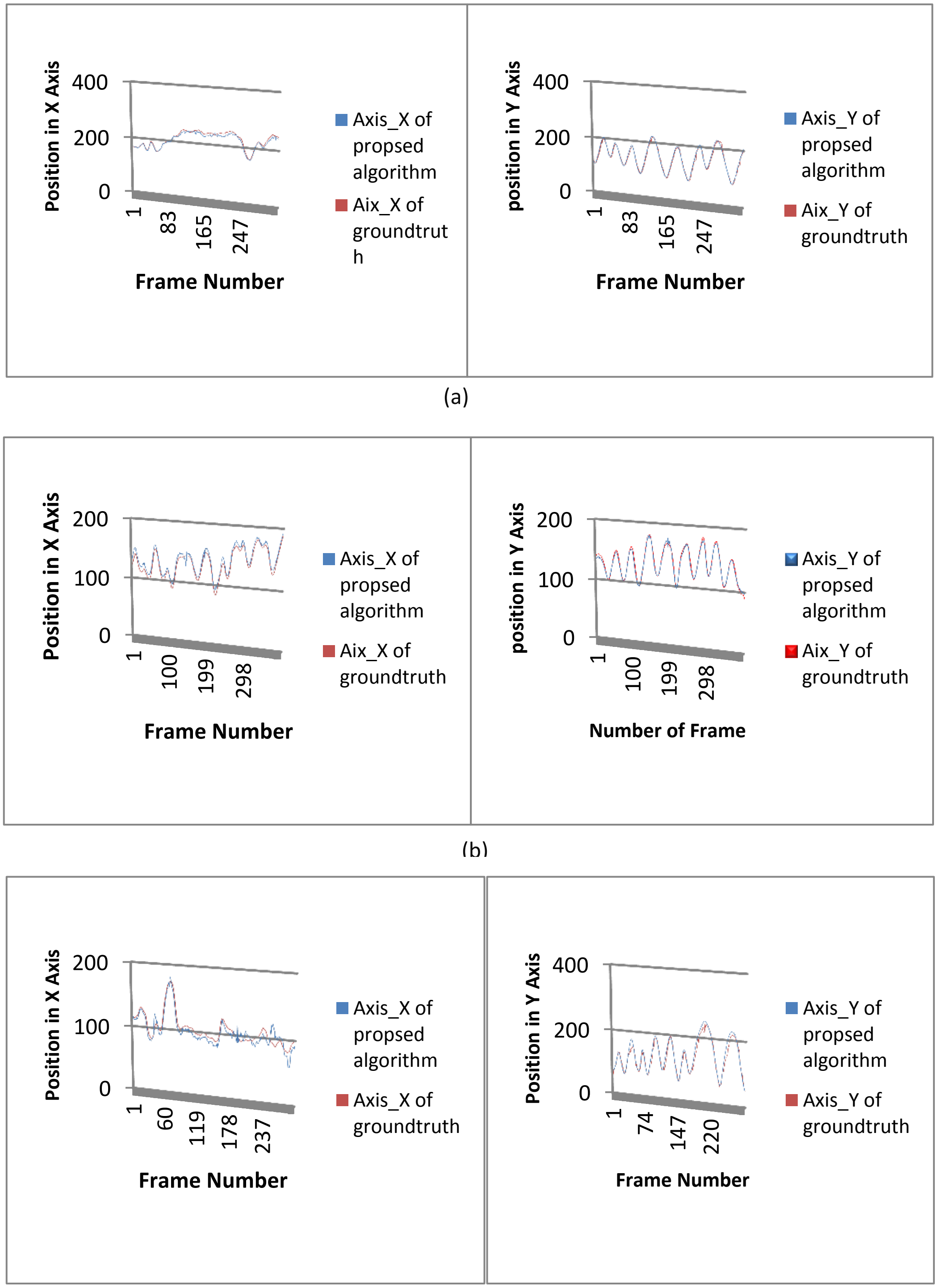
4 Results and Discussion

In order to evaluate the performance of the proposed algorithm, this method was implemented on video sequences of IBGHT dataset (Asaari et al., 2014), using Matlab 9.0. Fig.4 demonstrates the experimental results for positions hand gesture trajectory estimation on files Indoor_2, outdoor_3 and indoor_1 of IBGHT in comparison with ground truth data based on the center position of hand region(x & y).

The experimental results in an indoor_2 video sequence with 328 images –frames are depicted in Fig 4(a). It can be realized that the proposed Local Corner keypoints based tracker algorithm (LCKPs) are capable to correctly estimate the position of hand gesture even hand practice partial occlusion behind rigid objects. The reason is the algorithm takes advantage from BRISK and minimum eigenvalues keypoints that prove its ability to detect binary features of hand region even when hand practices particle occlusion and scale changes and translation.

In the experimental results in an outdoor_3 file that represented in Fig.4 (b), the proposed LCKPs based tracker algorithm shows its capability to work under lighting changes caused by over exposed phenomenon. That due to the searching strategy that the proposed LCKPs algorithm uses and incorporates each of orientation, skin color, motion and skin moving feature in estimating the right position for hand region.

From the experimental results in indoor_1that represented in Fig.4 (c), it can be seen that the proposed LCKPs algorithm can do correct estimation to hand region moving in an irregular manner (erratic motion) and practicing partial overlapping with face region. This is due the proposed algorithm can correctly examine the orientation feature between current candidate region orientation and reference region (Roih) based on hamming distance; in addition to other enforced conditions that contributed in making sure the candidate window would have hand region. As well, even though the proposed algorithm shows pretty drift to the arm region, it would ensure the tracker never stop and fail. The average accuracy rate upon the advocated video sequences is illustrated in Fig.5.



(a)

(b)

(c)

Fig.4. Samples of hand gesture position estimation based (a) Indoor_2, (b) outdoor_3 and (c) indoor_1 of video scenes respectively.

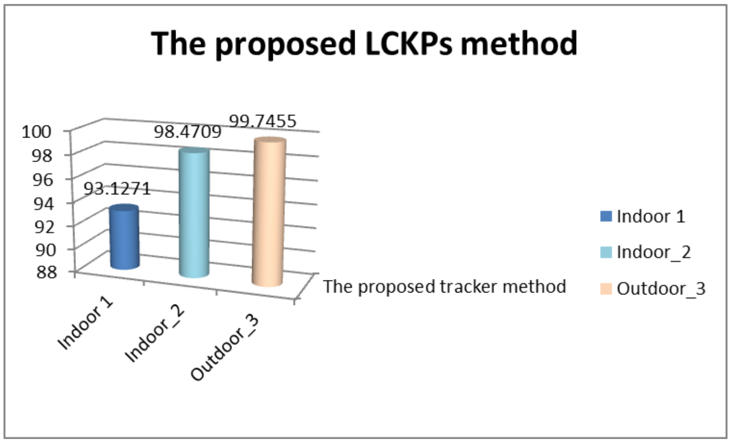


Fig.5. Average accuracy rate.

5 Conclusion

In this paper, we propose a method to hand position estimation for hand gesture trajectory tracking, we named Local Corner keypoints based tracker algorithm (LCKPs). The proposed algorithm used corner keypoints of BRISK and minimum eigenvalue techniques extracted from last segmented region of hand to estimate hand position on a current image frame of video sequences. The experimental results verified that the proposed BRISK and minimum eigenvalue techniques capable to consistency extract hand corner keypoints even hand during its movement practice changes in its scale, translation and partial occlusion behind static objects. In addition, the proposed algorithm can correctly estimate hand position in the video sequences with non-linear hand movement in the challenging environment. In future, we plan to extend this algorithm to a more sophisticated hand gesture tracking method that including more challenging scenarios.

References

Rautaray, S. S., & Agrawal, A. (2015). Vision based hand gesture recognition for human computer interaction: a survey. Artificial Intelligence Review, 43(1), 1-54.

Dixit, V., & Agrawal, A. (2015). Real-time Hand Tracking for Dynamic Gesture Recognition. In Proceedings of Fourth International Conference on Soft Computing for Problem Solving (pp. 153-164). Springer India.

Dardas, Nasser Hasan Abdel-Qader. Real-time hand gesture detection and recognition for human computer interaction. Diss. Université d'Ottawa/University of Ottawa, 2012.

Kılıboz, N. Ç., & Güdükbay, U. (2015). A hand gesture recognition technique for human–computer interaction. Journal of Visual Communication and Image Representation, 28, 97-104.

Kumar, P. P., Vadakkepat, P., & Loh, A. P. (2010). Hand posture and face recognition using a fuzzy-rough approach. International Journal of Humanoid Robotics, 7(03), 331-356..

Asaari, M. S. M., Rosdi, B. A., & Suandi, S. A. (2015). Adaptive Kalman Filter Incorporated Eigenhand (AKFIE) for real-time hand tracking system. Multimedia Tools and Applications, 74(21), 9231-9257.

Joshi, Miss Medha, and Sonal Patil. (2014). Implementation of an Improved HCI application for Hand Gesture Recognization. International Journal of Emerging Trends & Technology in Computer Science (IJETTCS), Volume 3, Issue 5, 2278-6856.

Guo, J., Cheng, J., Guo, Y., & Pang, J. X. (2013). A real-time dynamic gesture recognition system. In Applied Mechanics and Materials (Vol. 333, pp. 849-855). Trans Tech Publications.

Zou, X., Wang, H., Duan, H., & Zhang, Q. (2013, August). A Hand Model Updating Algorithm Based on Mean Shift. In International Conference on Information Computing and Applications (pp. 651-660). Springer Berlin Heidelberg.

Premaratne, P., Yang, S., Vial, P., & Ifthikar, Z. (2015, August). Dynamic Hand Gesture Recognition Using Centroid Tracking. In International Conference on Intelligent Computing (pp. 623-629). Springer International Publishing.

Stergiopoulou, E., Sgouropoulos, K., Nikolaou, N., Papamarkos, N., & Mitianoudis, N. (2014). Real time hand detection in a complex background. Engineering Applications of Artificial Intelligence, 35, 54-70.

Xu, D., Wu, X., Chen, Y. L., & Xu, Y. (2015). Online dynamic gesture recognition for human robot interaction. Journal of Intelligent & Robotic Systems, 77(3-4), 583-596.

Yeo, H. S., Lee, B. G., & Lim, H. (2015). Hand tracking and gesture recognition system for human-computer interaction using low-cost hardware. Multimedia Tools and Applications, 74(8), 2687-2715.

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, pp. I-I). IEEE.

Shugo Hamahashi, Shuichi Onami, & Hiroaki Kitano, (2008). Entropy filter, and area extracting method using the filter. US7460702 B2, Grant, US 10/433,463, PCT/JP2001/009901, Dec 2, 2008.

Wang, Z., Zhang, Z., Wang, F., & Sun, Y. (2014). Vision-based hand gesture interaction using particle filter, principle component analysis and transition network.

Mazumdar, D., Nayak, M. K., & Talukdar, A. K. (2015). Adaptive Hand Segmentation and Tracking for Application in Continuous Hand Gesture Recognition. In Recent Trends in Intelligent and Emerging Systems (pp. 115-124). Springer India.

Yao, Y., & Li, C. T. (2013, July). Real-time hand gesture recognition for uncontrolled environments using adaptive SURF tracking and hidden conditional random fields. In International Symposium on Visual Computing (pp. 542-551). Springer Berlin Heidelberg.

Gupta, K., & Kulkarni, A. V. (2008). Implementation of an automated single camera object tracking system using frame differencing and dynamic template matching. In Advances in Computer and Information Sciences and Engineering (pp. 245-250). Springer Netherlands.

Leutenegger, S., Chli, M., & Siegwart, R. Y. (2011, November). BRISK: Binary robust invariant scalable keypoints. In Computer Vision (ICCV), 2011 IEEE International Conference on (pp. 2548-2555). IEEE.

Heinly, J., Dunn, E., & Frahm, J. M. (2012). Comparative evaluation of binary features. In Computer Vision–ECCV 2012 (pp. 759-773). Springer Berlin Heidelberg.

Mair, E., Hager, G. D., Burschka, D., Suppa, M., & Hirzinger, G. (2010, September). Adaptive and generic corner detection based on the accelerated segment test. In European conference on Computer vision (pp. 183-196). Springer Berlin Heidelberg.

Shi, J. (1994, June). Good features to track. In Computer Vision and Pattern Recognition, 1994. Proceedings CVPR'94., 1994 IEEE Computer Society Conference on (pp. 593-600). IEEE.

Palaio, Hélio, and Jorge Batista. "A kernel particle filter multi-object tracking using gabor-based region covariance matrices." Image Processing (ICIP), 2009 16th IEEE International Conference on. IEEE, 2009.

Chong, Lee-Ying, et al. "2.5 D face recognition under tensor manifold metrics." International Conference on Neural Information Processing. Springer International Publishing, 2014.

Asaari, M. S. M., Rosdi, B. A., & Suandi, S. A. (2014). Intelligent Biometric Group Hand Tracking (IBGHT) database for visual hand tracking research and development. Multimedia tools and applications, 70(3), 1869-1898.

Eman Thabet, Fatimah Khalid, Puteri Suhaiza Sulaiman, and Razali Yaakob, (2017). “Low Cost Skin Segmentation Scheme in Videos Using Two Alternative Methods for Dynamic Hand Gesture Detection Method,” Advances in Multimedia, vol. 2017, Article ID 7645189, 9 pages, 2017. doi:10.1155/2017/7645189.

Norouzi, M., Fleet, D. J., & Salakhutdinov, R. R. (2012). Hamming distance metric learning. In Advances in neural information processing systems (pp. 1061-1069).