Real-Time Hand Gesture Detection and Recognition for Human Computer Interaction

Kapil Yadav and Jhilik Bhattacharya

Abstract This paper presents a gesture based system to interface Microsoft Word document. The system was developed using a two state discrete temporal model for gestures which works with distinct poses. The model fuses the state information along with individual pose recognition to activate the interfacing mechanism. It can be inferred from the experimental results that the model facilitates both accurate gesture recognition as well as promptness in response. The decision fusion of SURF and wavelet features prove to be robust for the current model. The selected features show a 96 percent accuracy when tested with gestures having varying background, scale, transformation and illumination conditions. The response time which varies between 3 to 3.5 second can be further improved by implementing the feature detection steps in VC++ environment instead of Matlab.

1 Introduction

Development of new technologies for Man Machine Interface hardware is in tandem with the corresponding advancement of software algorithms for data interpretation and processing. Technology enhancement thus saw a leap from Swept Frequency Capacitive Sensing techniques used in conventional touch screens, to interactive hardware displays like large displays, flexible displays and wearable displays for mixed reality. Current applications of MMI include smart homes, collaborative working environments, advanced information visualization and many more. The interfacing parameters range from multimodal interaction like touch, speech and gesture, to physiological factors such as ECG, EOG, Heart Rate, Eye blinking, facial expression for example. In particular, research focussed on intelligent human and machine interfaces to obtain a more intuitive and adaptive capability for advanced informa-

K. Yadav (✉) · J. Bhattacharya
Thapar University, Patiala, Punjab, India
e-mail: {kapilyadav1204,bjhilik}@gmail.com

S. Berretti et al. (eds.), *Intelligent Systems Technologies and Applications*,
Advances in Intelligent Systems and Computing 384,
DOI: 10.1007/978-3-319-23036-8_49

tion visualization that analyses and comprehends multi-dimensional information. To mention a few, considerable work has been done in voice based system control [1]. A vision-based system was presented in [2] for voluntary eye blink detection and pattern interpretation for HCI. Sign language development for deaf and dumb people has been a major drive for the advent of gesture based interfaces. Initially, work was done to automate communication between visually impaired and deaf people[3]. This later motivated exploration of gesture based interaction, whose primary application target was gaming and home entertainment. Work done by various researchers in this field also includes gesture based robot locomotion[4],[5] surgical systems. G-Speak spatial computing operating system, offers movement of data between different computing systems and displays through a gesture interface[6]. A Virtual keyboard was proposed [7] which provides a detectable surface on which user can move fingers that replicates the act of key pressing. The current work proposes a system which utilizes vision based hand gestures to interface Microsoft Word document in real time. The work uses gestures for opening, closing, changing font size and color, scrolling up and down in the word document. Gesture based research previously concentrated on offline gesture recognition mainly. Realtime gesture recognition requires accurate gesture modelling techniques for prompt response to the HCI. A probabilistic framework was presented by Ying Yin et. al. [8] for real-time gesture recognition. A taxonomy for Gesture modelling was presented in [9] where gesture was modelled as form, flow, temporal gestures. This work contributes a two state discrete temporal model for gestures which works with distinct poses. The model facilitates both accurate gesture recognition as well as promptness in response.

2 System Description

The system consists of a camera which will acquire images in real time. The software consists of three modules (as seen in figure 1). (i) A feature database which was

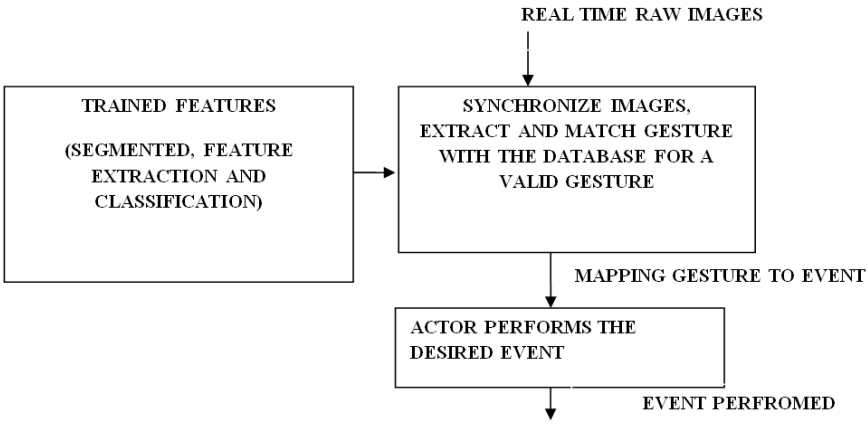


Fig. 1 Gesture based HCI system for Word Document Handling

trained offline (ii) A module called synchronize which detects gestures from captured images and matches it with the database for a valid gesture.(iii) The third module actor performs the desired event based on the matched gesture. The feature database generation and synchronization is done using Matlab. The actor needs to interface Microsoft OLE for the corresponding events to be handled in the word document. Document opening, closing, scroll up, scroll down, font color and size are the different events handled by the interface currently.

3 Methodology

The entire work can be divided in two phases training and testing. In the first phase, a set of gesture images are trained so that each gesture corresponds to a particular pattern. This includes hand region segmentation from an image, extracting features from the segmented region, using the feature vector for training. In the second phase a hand gesture has to be mapped to an event in realtime and the corresponding event needs to be performed.

3.1 Image Sequence by Camera and Acquizition

Images in realtime are acquired using the webcam provided in the laptop. The image acquisition program is looped to continuously capture snapshots every second. Any variations in gesture representation due to snapshot intervals are handled by the synchronization module.

3.2 Segmentation

In this phase, hand regions need to be extracted from the background for feature extraction and recognition purposes. The performance of the feature extractor largely varies on the segmentation algorithm selected. The task becomes challenging with cluttered background. Also, the algorithm should be robust against scene illumination and skin variations. Background modelling with K Gaussian distributions [10], connected-component labelling algorithm [11], Automatic Seeded Region Growing Algorithm (ASRGC) [13] with YCbCr model, meanshift filters are some of the mostly used segmentation algorithms. This paper uses k-means on RGB color image for skin color segmentation.

3.3 Features Extraction

Previous work on Gesture recognition has used shape based , keypoint based as well as region based algorithms. Shape based algorithms used different shape signatures (fourier descriptors for example) to detect hand shape while SIFT [12] is an example of keypoint based techniques. Region based algorithms like wavelet descriptors,

PCBR are also used by many researchers. The current method uses SURF and Wavelet descriptors as feature vectors.

3.3.1 SURF

SURF detector uses box filters to approximate the Gaussian and can be computed in constant time using the integral image [15]. At scale σ, Hessian matrix $H(p, \sigma)$ of a point $p\ (x, y)$ in the image $f1$, is defined as shown in equation 1 .

$$H(p, \sigma) = \begin{bmatrix} L_{xx}(p, \sigma)\ L_(xy)(p, \sigma) \\ L_{xy}(p, \sigma)\ L_(yy)(p, \sigma) \end{bmatrix} \tag{1}$$

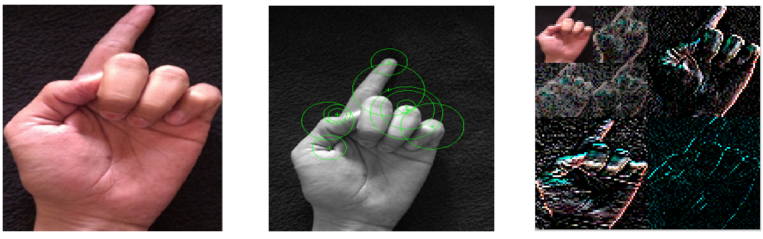
where $L_{xx}(p, \sigma)$ is image convolution of second derivative $\frac{dx}{dx^2} g(\sigma)$.

3.3.2 Wavelet Features

Input hand images are decomposed through the Wavelet Packet Decomposition using the Haar (Daubechies 1) basis function. The image is divided into four bands: LL(left-top), HL(right-top), LH(leftbottom) and HH(right-bottom). The HL band indicated the variation along the x-axis while the LH band shows the y-axis variation[12].

3.3.3 Canonical Correlation Feature

A feature fusion of of SURF and Wavelet Packets is obtained by computing the Canonical Correlation [14] of the two features. CC creates a new feature vector for each set of SURF and wavelet vectors such that the correlation between these variables is maximized and independent of affine transformation. Equation 2 gives the canonically correlated variable Z.



(a) K-Means Clustered image (hand color imposed on the segmented hand cluster)

(b) SURF features on hand image

(c) 2nd level wavelet decomposition

Fig. 2 SURF and Wavelet features extracted from K-means clustered image.

$$Z_i = \begin{bmatrix} A & 0 \\ 0 & B \end{bmatrix}^T \begin{bmatrix} X_i \\ Y_i \end{bmatrix}$$

$$C_x y = \frac{1}{L} \sum_{i=1}^{L} x_i y_i^t \tag{2}$$

Where x_i and y_i denote the SURF and wavelet feature vector of the i^{th} image respectively. A and B are the eigenvectors of $C_{xx}^{-1} C_{xy} C_{yy}^{-1} C_{xy}^T$ and $C_{xx}^{-T} C_{xy} C_{yy}^{-1} C_{xy}$ respectively. C_{xy} gives the covariance matrix of x and y and L denotes the total number of training images.

3.4 Classification

Two types of classification are generally used by recognition applications. One category uses various distance metrics like Euclidean and Mahalanobis to compute the difference between the test vector with the different classes of vectors and assigns the test vector to the class having the least distance. Another category uses machine learning algorithms like SVM, NN, AdaBoost, HMM to classify the data. This work uses a combination of neural network and Euclidean distance classifiers. The feature vectors of SURF and Wavelet features and CC vector Z are tested with neural network (equation 3), whereas the decision fusion (equation4) is tested with Euclidean classifiers.

$$\begin{aligned} x &= [-1, x_1, x_2, x_3, x_4, ...x_p]^T \\ \text{weights } w &= [-1, w_1, w_2, w_3, w_4, ...w_p]^T \\ \text{output } odx &= \sum_{i=0}^{P} x_i * W_i \\ \text{error } e_x &= ox - odx \text{ odx is the actual output} \\ \text{updated weights } \Delta w_i &= -\rho * \frac{\delta e}{\delta W_i} \end{aligned} \tag{3}$$

$$d_i^{comb} = \frac{1}{d_i^y} + \frac{1}{d_i^x} \tag{4}$$

where d_i^x is computed for all image vectors x_i and test vector x^t. The i for which d^{comb} is the maximum is considered as the correct match.

3.5 Synchronization

A software-based system for the real-time synchronization of images captured by a lowcost camera framework is presented (as seen in figure 3). It is highly recommended for cases where special hardware cannot be used. Every gesture is identified in two steps. A start symbol denotes the start of the gesture, which is followed by appropriate document open, close or other gestures. As shown in the figure 4, interim invalid gestures may be captured in snapshots while the user is forming the particular gesture. All these gestures will not find any match in the database, however the recognition module will run unnecessarily wasting processor time. Hence a frame is passed for recognition only after it becomes static. Thus only when last three captured frames have no change, it is forwarded for segmentation, feature detection and gesture recognition. The start state is maintained as long as a valid gesture is not recognized. Once a gesture is recognized and the corresponding event is invoked, the cycle is complete. The next event will be marked by another start state.

3.6 Document Handling

The work uses existing MS Office Automation (OLE) modules and interfaces it with corresponding gesture recognition events. This is shown in figure 5. For example when a gesture denoting document open OLE is invoked, it is recognized as 001 which in turn calls the corresponding module to open the document, and the document is opened. Same is the case for the font color, size or scroll gestures.

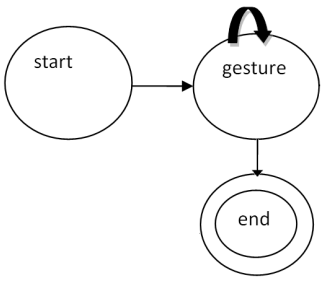


Fig. 3 A start gesture and another corresponding gesture is required for an event invocation.

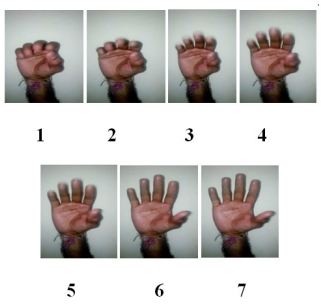


Fig. 4 Transition between 2 gestures captured by the camera.

(a) Start gesture 000 (b) Document open 001 (c) Document scroll up 011 (d) Document scroll down 010 (e) Document font change 100 (f) Document close 101

Fig. 5 The different gestures used for particular events

4 Experimental Results

A dataset of 300 images are taken as test samples. Three types of test sample images are considered for experimental purpose for all the methods and procedures explained in this paper namely clear background (C1) , slightly cluttered background (C2), and slightly cluttered background with changing lighting conditions (C3). All the images are captured in home environment without any special lighting using a consumer quality web camera. The resolution of the images considered for processing after segmentation is 128 X 128. The accuracy and performance of the proposed system was further verified using realtime test cases. Hundred gestures , randomly selected and performed, by different people at different illumination and backgrounds were used for generating the test results shown in table 1 and 2. Table 1 gives a comparison of different feature vectors used. The technique which gives the best performance (decision fusion approach in the present case) is further used for word document interfacing. The time and performance accuracy for each event is shown in table 2.

It was seen that the decision fusion approach with wavelet and SURF features gives a satisfactory performance on the current dataset.

Table 1 Performance comparison of feature detection methods.

No	Method	Percentage
1	wavelet	93
2	SURF	90.5
3	feature fusion	87
4	decision fusion	96

Table 2 Performance of event invocation on gesture (in figure 5) recognition

Gesture	Percentage	Time(s)
5b	95	4.5
5c	92	4
5d	91	3.5
5e	93	3
5f	90	3.25
5a	98	2

5 Conclusion and Future Work

The work proposes a gesture based Microsoft Document handling system which operates using a two state gesture model. The performance of the system depends on the gesture synchronization and the gesture recognition algorithms. The former have been handled by a temporal modelling of gestures. This can further be improved by combining temporal model along with form and path. The gesture recognition shows a performance collation between SURF and wavelet transformation along with feature fusion and decision fusion approaches. The decision fusion of Wavelet and SURF features show a 96 percent accuracy on images with varying background, scale, transformation and illumination conditions. The number of features chosen for decision or feature fusion can be increased based on the available hardware for implementation. It was limited to two in the present work, implemented, on a laptop with Intel core i3 processor (CPU 2.27GHz) on a 64 bit windows platform. A number of extensions of the current work are under progress. A) The work is being implemented on other software. B) Two dimensional image information processed by the recognizer is being extended to include the depth dimension. This is further used by the algorithm to estimate user distance from the screen. Thus there will be an automatic zoom in or out depending on this distance. C) Dynamic gestures are being considered instead of static ones. As a result the same gesture will work for scroll up or down depending whether the hand is moving upwards or downwards.

References

1. Cui, B., Xue, T.: Design and realization of an intelligent access control system based on voice recognition. In: ISECS International Colloquium on Computing, Communication, Control, and Management. CCCM 2009, vol. 1. IEEE (2009)
2. Sumathi, S., Srivatsa, S.K., Uma Maheswari, M.: Vision based game development using human computer interaction (2010). arXiv preprint arXiv:1002.2191
3. Ghotkar, A.S., et al.: Hand gesture recognition for indian sign language. In: 2012 International Conference on Computer Communication and Informatics (ICCCI). IEEE (2012)
4. Lee, C., Xu, Y.: Online, interactive learning of gestures for human/robot interfaces. In: 1996 IEEE International Conference on Proceedings of the Robotics and Automation, vol. 4. IEEE (1996)
5. Nosowitz, D.: Video: MIT's Kinect Hack Tracks All Ten Fingers Simultaneously (2010)
6. Malik, S., Laszlo, J.: Visual touchpad: a two-handed gestural input device. In: Proceedings of the 6th international conference on Multimodal interfaces. ACM (2004)
7. Samanta, D., Sarcar, S., Ghosh, S.: An approach to design virtual keyboards for text composition in Indian languages. International Journal of Human-Computer Interaction **29**(8), 516–540 (2013)
8. Yin, Y., Davis, R.: Real-time continuous gesture recognition for natural human-computer interaction. In: 2014 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC). IEEE (2014)
9. Wobbrock, J.O., Morris, M.R., Wilson, A.D.: User-defined gestures for surface computing. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM (2009)

10. Lucchi, A., et al.: Supervoxel-based segmentation of mitochondria in em image stacks with learned shape features. IEEE Transactions on Medical Imaging **31**(2), 474–486 (2012)
11. Shapiro, Stockman, G.: Computer Vision. Prentice Hall (2002)
12. Kook-Yeol, Y.: Robust hand segmentation and tracking to illumination variation. In: 2014 IEEE International Conference on Consumer Electronics (ICCE), pp. 286–287, January 10–13, 2014
13. Yang, G., et al.: Research on a skin color detection algorithm based on self-adaptive skin color model. In: 2010 International Conference on Communications and Intelligence Information Security (ICCIIS). IEEE (2010)
14. Qi, F., Weihong, X., Qiang, L.: Research of Image Matching Based on Improved SURF Algorithm. TELKOMNIKA Indonesian Journal of Electrical Engineering **12**(2), 1395–1402 (2014)
15. Suaib, N.M., et al.: Performance evaluation of feature detection and feature matching for stereo visual odometry using SIFT and SURF. In: 2014 IEEE on Region 10 Symposium. IEEE (2014)