Dynamic Gesture Recognition Method Based on Improved CIPBR Algorithm

Kavya Sree.M[1], Kavitha.C[2] , Vijay Swaroop[3]
Department of Computer Science Engineering
[1,2,3] Atria Institute of Technology ,Bangalore ,India

Abstract- *Dynamic gesture recognition has been studied actually for it big application in several areas such as virtual reality, games and sign language. But some problems have to be solved in computer applications, such as response time and classification rate, which directly affect the real-time usage. This paper proposes a novel algorithm called Convex Invariant Position Based on Ransac which improved the good results in dynamic gesture recognition problem. The proposed method is combined with a adapted PSO variation to reduce features and a HMM and three DTW variations as classifiers*

Keywords- Gesture recognition, computer vision, CIPBR, dynamic time wrapping, hidden Markov model.

I. INTRODUCTION

In recent years, with the development of computer technology, human-computer interaction technology has been more and more widely used [1], and the gestures recognition is the focus of popular research[2]. Dynamic gesture recognition can be divided into three categories : CRFs, HMM and DTW, which are based on statistics, based on semantics and template [3]

HAND recognition gestures systems brought to several areas a more natural way for Human-Computer Interaction (HCI). Applications in sign language recognition [4], virtual reality [5], and computer games [6] have increasingly used hand gesture recognition approaches to facilitate the learning based on intuition to remember and performer a gesture. There are three different categories for systems based in human gesture recognition: systems based on the hand gesture captured by gloves or external sensors [7], system which make the device tracking to generate a gesture path [8]. The last category use a camera to capture the gesture in images and extract features from it using computer vision techniques to interpret the gesture [9], [10].

The first class, hand gesture captured by gloves or external sensors, uses some devices connected in the user hand what turn difficult the natural articulation of hand gesture. This category has the advantage to being invariant to light conditions and complex background, achieving a better result than others applications but is more expensive, due do the high price of the devices to develop a system based on it.

The second category, is the more limited class, by the limited number of gestures which can be recognized. Most of this systems are presented in-portable devices where predefined gesture execute some action. This gestures complexity is very low in compared with others categories.

The last category, system based on computer vision techniques, uses a camera to capture the hand gesture in images or videos and extract attributes and features, such as position, velocity, color, among others to identify the gesture.

This class has been grown by the present facility to obtain a camera usually coupled in a smart-phone. Another reason for this growth, is the non-invasive techniques which do not require any devices connected and nothing which hinders the natural hand gesture movement.

Systems based on vision computer techniques usually involve two steps: feature extraction and pattern classification. This paper presents a novel technique called Convexity Invariant Position Based on Ransac (CIPBR) for hand gesture image feature extraction as first module in a dynamic hand gesture system. In addition we used two classifiers to evaluate the proposed method: Dynamic Time Wrapper (DTW) [8] and Hidden Markov Model (HMM) [9].

II. RELATED WORKS

Several techniques have been used in hand gesture recognition systems. Meena [10] and El-Salwah [11] use the Local Contour Sequence (LCS) algorithm to reduce the hand posture into a distances vector with the better contour points. Calinon-et-al. [12] use probabilistic techniques as Principal Components Analysis (PCA) and Independent Components (ICA) to recognize and reproduce gestures. The Speed Up Robust Features (SURF) used for Bao-et-al. [13] extract points of interest sets over pyramidal images using a Gaussian Laplacian [14] and recognize dynamic gestures making a track path with SURF features. Some of those algorithms previously cited lose their accuracy when a new test image has a different rotation from the ones observed in training step. To solve this problem Wysoski-et-al. [15] use boundary histograms and neural networks to correct the hand posture angle. Grzeszcuk-et-al. [16] use an approach based on stereo vision to realize in 3D the angle correction. Keogh-et-al. [17] propose a simple method to correct the angle variation based on object form, where do not matter the angle distortion the same contour pixel is extracted of the image as the first one and the rest of the contour pixels were always organized in the same order. Keogh-et-al. achieve good results with this technique in skulls classification, extracting two signatures of the images, distances and angles, and turning them into a single features vector.

Barros-et-al. [18] propose a method to solve another problem in computer vision approaches: the feature vector size. The traditional approaches return a big vector which complicate the real-time classification task. The Convexity Approach solve this reducing the hand posture into a polygonal shape and extracting only the more external and internal contour points. For last, it is calculated the distance between each pair of points. Then, a set of points is selected in order to obtain the minimal contour representation and generate a smaller feature vector. Barros-et-al. evaluates their work using three pattern classification: Elman Recurrent Neural Network(Elman RNN) [19], HMM and DTW achieving better results with the last two. Based on Keogh-et-al. and Barros-et-al. concepts, CIPBR algorithm is proposed and evaluated in this paper using two classifiers: HMM and DTW. The last one is used in three version, the traditional one, an adaptation to CIPBR's two signatures and a variation presented by Salvador [20].

CIPBR Algorithm

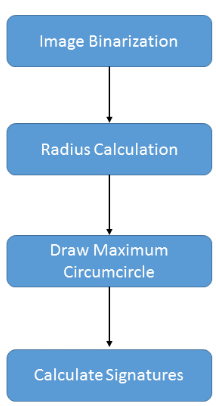

Fig. 1. CIPBR's flow

The CIPBR algorithm is composed by simple tasks to reduce the hand posture images into two signature sets. There is four modules in cascade. Figure 1 presents CIPBR's flow starting with the input image and finishing with the feature vector.

The first module,"Image Binarization", receives a hand posture image as input and outputs two images: the first one in RGB, same as the input, and the second a binary image (based on Otsu ThresholdAlgorithm [21]).

The second module, "Radius Calculation", has three simple
tasks to complete. The first one is find the hand pulse line in RGB image using a simple linear regression. The second one is extract the contour where each pixel is marked as border, if it has one of its eight neighbors black in the binary image and calculate the mass center point by Hu Invariant moments. This module use same principle as Keogh et al. [17] to guarantees the rotation invariance starting the contour organization with the center pulse line pixel. Lastly the distance between the center of mass of the contour point and pulse center point is calculated as shown in Figure 2 (a).

The third module, "Draw Maximum Circumcircle", uses the distance previously calculated as radius to draw a circle inside the hand contour. In order to solve the problem of the circle exceeding the hand contour, a triangle is calculated using the three more distant contour points from mass center point and the biggest circle inside it is used.

Then, in "Calculate Signatures" module, convex Hull is calculate from hand posture contour using Andrew's monotone chain convex hull algorithm [22], thus reducing the number of contour points substantially.

As result a set $P = \{p_1, p_2, \ldots, p_n\}$ from Convex hull points are used to generate CIPBR signature sets. For each point $p_k = \rho$ is traced a line PC starting with ρ and ending with the central point C of the maximum circumcicle hand shape exported from Module 3. Then, it is measured the Euclidean distance from ρ to the point Q_k given by the intersection between PC and the maximum circumcicle. The set of all distances computed as this procedure is the descriptors for the first signature set. The distance ρQ_k can be calculated withthe following equation:

$$\text{Distance } \rho Q_k = (\rho_x - C_x) + (\rho_y - C_y)^2 - \text{radius},$$

where C is the hand posture contour mass center point, ρ_x, C_x are the x coordinates from points ρ and C respective, ρ_y, C_y are the y coordinates from points ρ and C respective and radius the radius calculated in the second module.

The second signature set consists of angles (A) obtained by calculating the angle (A) between a line composed for each point P of the convex hull hand shape point and radius (PC). All two signature sets are obtained in a clockwise directionalways starting with point P as Figure 2 shows.

Finally, to create the feature vector the signature sets are normalized, distances by the radius calculated in the second module as follows:

$$\text{Distance } i = \text{Distance } i \,/\, \text{radius}$$

and angles set by 360 . The final vector is created concatenating distances and angles in a single vector as follows:

$$\text{Angle } i = \text{Angle } i \,/\, 360^{\circ}$$

IV. CLASSIFIER MODULE BASED ON HMM

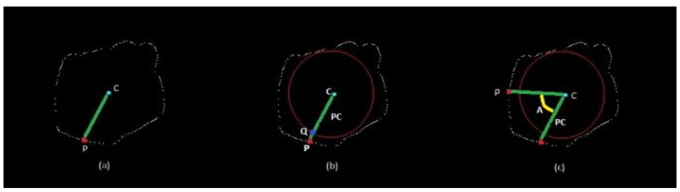

Fig. 2. (a) CIPBR second step output. The hand shape contours and central point (cyan point). The hand base middle point (red point), the radius (green line) and angle A (yellow arc). (b) and (c) CIPBR fourth step signature calculate.

For some classification techniques, such as Hidden Markov Model and Artificial Neural Networks, among others require a fixed length feature vector as input, but most of techniques return as output vectors with different sizes. CIPBR algorithm returns a set of vectors, each one containing around 200 features. To use this vectors as input in a HMM, and achieve a convergence state, it is necessary a heavy reduction of features in each vector. To solve this, two approaches is adapted in this work. At first one, Particle Swarm Optimization [23] chose the better feature vectors size and the selector algorithm proposed by Barros et al. [6] receives PSO size output and choose the candidate features for this vectors.

A. Selector Algorithm

After better feature vector size has been chose for Binary PSO, another normalization is used to choice the features which will compose the vectors. To this task the Barros-et-al. Selector Algorithm [18] is applied. First, choose S, the PSO chosen size. Then, if any vector has fewer points than S, are added in the feature vector until matches the desired length (S). The feature vectors with more points than S are redefined using a selection algorithm. This algorithm consists in calculation of a window W through the division of the current vector length by S (desired length). The current vector is parsed, and each value in W position is included to the new feature vector. If the new output vector is even smaller than the desired length, the remaining positions are randomly visited and used to compose the new output vector until the desired length is achieved.

B.HMM Training Configuration

The Hidden Markov Model technique uses a K-Means Clustering [25] to find the best initial approximation. The Baum-Welch algorithm [26] is used to train the HMM, resulting in a fast training process.

V. CLASSIFIER MODULE BASED ON DTW

The Dynamic Time Warping (DTW) was introduced to overcome the limitation in measure the distance between two time series in specific case: when there is distortion in one of then shifting some slice. To solve this a simple approach based on Euclidean distance is proposed as follows:

Given two time series X, and Y, of lengths $|X|$ and $|Y|$, construct a warp path $W = \{1, w_2, w_3, \ldots, w_k\}$ where $\max(|X|, |Y|) \leq K < |X| + |Y|$ and the k th element is $w_k = (i,j)$ where i is an index from time series X and j an index from time series Y.
The warp path must start at $w_1 = (1, 1)$ and finish at $w_k = (i, j)$ in order find the cost matrix
To find the minimum-distance warp path, every cell of the cost matrix must be filled. The value of a cell in the cost matrix is:

$$Dist(i, j) = Dist(i, j) + \min[D(i-1, j), D(i, j-1), D(i-1, j-1)],$$

The warp path to D(i, j) must pass through one of those three grid cells, and since the minimum possible warp path distance is already known for them, all that is needed is to simply add the distance of the current two points to the smallest one. Since this equation determines the value of a cell in the cost matrix by using the values in other cells, the order that they are evaluated in is very important.

A. DTW for CIPBR

The DTW algorithm has a complexity problem of $O(N^2)$ level. This has a direct effect on time rating increasing it exponentially higher order time serie which compromises a real time system. To solve this problem two approaches are used in DTW. The first one in the way that DTW calculates the cost matrix and second on in the presentation series to classifier.

The first change is made in the classifier in the following way:

$$Dist(i, j) = abs[(D_i - D_j) + (A_i - A_j)] + min[D(i-1, j), D(i, j-1), D(i-1, j-1)].$$

Where D_i, D_j are distances and A_i, A_j angles, from CIPBR feature vectors. All the rest of the DTW algorithm follows the same traditional way.

VII. CONCLUSION

This paper presents a new approach for feature extraction and classification of hand gestures called CIPBR. CIPBR + DTW combination improve the previously results for RPPDI dataset. Also is presented a new method to use the binary version of PSO to find the optimal number of features for CIPBR vector being selected by a simple selector algorithm and classified

Fig. 3. RPPDI Dynamic Gesture Dataset [18]

using the Hidden Markov Model as classifier. This selector algorithm is necessary because PSO fitness function generated in conclusive results for classification, being necessary another method to choice the feature to final vectors.

REFERENCES

[1] X. Zhang, X. Chen, Y. Li, V. Lantz, K. Wang, and J. Yang, "A framework for hand gesture recognition based on accelerometer and EMG sensors," IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans, vol. 41, no. 6, pp. 1064–1076, 2011.

[2] G. Lu, L.-K. Shark, G. Hall, and U. Zeshan, "Immersive manipulation of virtual objects through glove-based hand gesture interaction," Virtual Reality, vol. 16, no. 3, pp. 243–252, 2012.

[3] J. P. Wachs, M. Kölsch, H. Stern, and Y. Edan, "Vision-based hand- gesture applications," Communications of the ACM, vol. 54, no. 2, pp. 60–71, 2011.

[4] G. Dewaele, F. Devernay, and R. Horaud, "Hand motion from 3d point trajectories and a smooth surface model," in Computer Vision-ECCV 2004. Springer, 2004, pp. 495–507.

[5] J. Schlecht, B. Carqué, and B. Ommer, "Detecting gestures in medieval images," in 2011 18th IEEE International Conference on Image Processing (ICIP). IEEE, 2011, pp. 1285–1288.

[6] P. V. A. Barros, N. T. M. Júnior, J. M. M. Bisneto, B. J. T. Fernand dynamic gesture recognition system based on the feature vector reduction for SURF and LCS," in Artificial Neural Networks and Machine Learning–ICANN 2013. Springer, 2013, pp. 412–419.

[7] Z. Ren, J. Yuan, J. Meng, and Z. Zhang, "Robust part-based hand gesture recognition using kinect sensor," IEEE Trans. Multimedia, vol. 15, no. 5, pp. 1110–1120, 2013.

[8] H. Sakoe, S. Chiba, A. Waibel, and K. F. Lee, "Dynamic programming algorithm optimization for spoken word recognition," Readings in speech recognition, vol. 159, 1990.

[9] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," in Readings in speech recognition, A. Waibel and K.-F. Lee, Eds. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1990, pp. 267–296. [Online]. Available: http://dl.acm.org/citation.cfm?id=108235.108253

[10] S. Meena, "A study on hand gesture recognition technique," Master's thesis, National Institute of Technology, Rourkela, 2011.

[11] A. El-Sawah, C. Joslin, N. D. Georganas, and E. M. Petriu, "A framework for 3D hand tracking and gesture recognition using elements of genetic programming," in Fourth Canadian Conference on Computer and Robot Vision 2007. CRV'07. IEEE, 2007, pp. 495–502.

[12] S. Calinon and A. Billard, "Recognition and reproduction of gestures using a probabilistic framework combining PCA, ICA and HMM," in Proceedings of the 22nd international conference on Machine learning. ACM, 2005 pp. 105–112.

[13] J. Bao, A. Song, Y. Guo, and H. Tang, "Dynamic hand gesture recognition based on SURF tracking," in 2011 International Conference on Electric Information and Control Engineering (ICEICE). IEEE,2011, pp. 338–341.

[14] P. J. Burt and E. H. Adelson, "The laplacian pyramid as a compact image code," IEEE Transactions on Communications, vol. 31, no. 4, pp. 532–540, 1983.

[15] S. G. Wysoski, M. V. Lamar, S. Kuroyanagi, and A. Iwata, "A rotation invariant approach on static-gesture recognition using boundary histograms and neural networks," in Proceedings of the 9th International

Conference on Neural Information Processing, ICONIP’02, vol. 4. IEEE, 2002, pp. 2137–2141.

[16] R. Grzeszczuk, G. Bradski, M. Chu, and J.-Y. Bouguet, “Stereo based gesture recognition invariant to 3D pose and lighting,” in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, vol. 1, 2000, pp. 826–833 vol.1.

[17] E. Keogh, L. Wei, X. Xi, S.-H. Lee, and M. Vlachos, “LB Keogh supports exact indexing of shapes under rotation invariance with arbitrary representations and distance measures,” in Proceedings of the 32nd international conference on Very large data bases. VLDB Endowment, 2006, pp. 882–893.

[18] P. V. A. Barros, N. Junior, J. M. M. Bisneto, B. J. T. Fernandes, B. L. D. Bezerra, and S. M. M. Fernandes, “Convexity local contour sequences for gesture recognition,” in Proceedings of the 28th Annual ACM Symposium on Applied Computing. ACM, 2013, pp. 34–39.

[19] L. Medsker and L. C. Jain, Recurrent neural networks: design and applications. CRC press, 1999.

[20] S. Salvador and P. Chan, “Toward accurate dynamic time warping in linear time and space,” Intelligent Data Analysis, vol. 11, no. 5, pp. 561–580, 2007.

[21] N. Otsu, “A threshold selection method from gray-level histograms,” Automatica, vol. 11, no. 285-296, pp. 23–27, 1975.

[22] A. M. Day, “Planar convex hull algorithms in theory and practice,” in Computer graphics forum, vol. 7, no. 3. Wiley Online Library, 1988, pp. 177–193.

[23] J. Kennedy, R. Eberhart et al., “Particle swarm optimization,” in Proceedings of IEEE international conference on neural networks, vol. 4, no. 2. Perth, Australia, 1995, pp. 1942–1948.

[24] X. Zhang, W. Wang, Y. Li, and L. Jiao, “PSO-based automatic relevance determination and feature selection system for hyperspectral image classification,” Electronics Letters, vol. 48, no. 20, pp. 1263–1265, September 2012.

[25] J. A. Hartigan and M. A. Wong, “Algorithm AS 136: A k-means clustering algorithm,” Applied statistics, pp. 100–108, 1979.

[26] T. L. Baum, G. Peterie, and N. W. Souled, “A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains,” in Proceedings of Annals of Mathematical Statistics, vol. 41, 1995, pp. 164 –171.