Deformable Part Model Based Hand Detection against Complex Backgrounds

Chunyu Zou, Yue Liu, Jiabin Wang, Huaqi Si

Beijing Engineering Research Center of Mixed Reality and Advanced Display,
School of Optoelectronics, Beijing Institute of Technology, Beijing, China, 100081
zoucy@hotmail.com,liuyue@bit.edu.cn,
wangjiabin315@163.com,1601296077@qq.com

Abstract. Hand detection is a challenging task in hand gesture recognition system and the detection results can be easily affected by changes in hand shapes, viewpoints, lightings or complex backgrounds. In order to detect and localize the human hands in static images against complex backgrounds, a hand detection method based on a mixture of multi-scale deformable part models is proposed in this paper, which is trained discriminatively using latent SVM and consists of three components each defined by a root filter and three part filters. The hands are detected in a feature pyramid in which the features are variants of HOG descriptors. The experimental results show that the proposed method is invariant to small deformations of hand gestures and the mixture model has a good performance on NUS hand gesture dataset - II.

Keywords: hand detection ·deformable part model ·latent SVM ·HOG features ·complex backgrounds

1 Introduction

Hand gestures are important body languages in human daily communication. Traditional HCI (Human Computer Interaction) devices such as keyboard and mouse are subject to the limitations of operational distance and convenience, so it's a natural way for us to interact with the computer using hand gestures. Hand gesture recognition has various applications such as sign language recognition, remote video conference, games and VR (Virtual Reality). Glove based and vision based methods are usually used in hand gesture recognition system, in which glove based method requires user to wear special gloves which can deliver the movements of hands and fingers to the computer [1]. Such an approach can accurately recognize various hand gestures in real time, but it is an unnatural and expensive way to interact with the computer because of the adoption of the complex glove equipment. Vision based hand gesture recognition has become popular in recent years, it doesn't require the user to wear gloves and only a camera is used to capture images of hands, which is a natural and friendly way for us to interact with the computer [2]. Fig.1 shows the process of vision based hand gesture recognition.

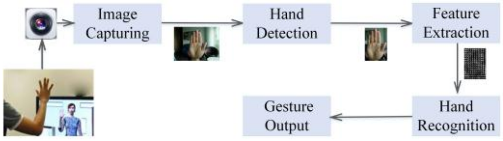


Fig. 1. Process of vision based hand gesture recognition

There are two types of hand gesture used in HCI system, i.e. static gesture and dynamic gesture, in which static gesture positions remain unchanged during a period of time and dynamic hand gesture positions are temporal and change with respect to time [3]. Static hand gesture recognition becomes popular in recent years because dynamic hand gestures can be considered as actions composed of a series of static hand gestures. The most difficult problem of vision based static hand gesture recognition is to detect hands against complex backgrounds, although depth cameras such as Kinect, LeapMotion and RealSense are robust and precise to detect hands according the depth and image information, they are not available in most existing systems. So it's important to study the hand detection approach in RGB images against complex backgrounds.

Skin color, motion information, shape or combination of these visual features are usually used to detect hands in RGB images. Hand detection in static RGB images is a challenging task for various hand shapes, viewpoints, changes in illumination, or complex backgrounds. In this paper, a hand detection method based on a mixture of multi-scale deformable part models is proposed and the mixture model is trained using latent SVM with positive examples from images which are labeled with bounding boxes around the entire hand gestures and "hard negative" examples. The mixture model is tested on NUS hand gesture dataset - II [17] and the experimental results show that the mixture model has a person independent performance.

2 Related Work

There are plenty of existing literatures about hand detection, which can be summarized into the skin color, motion information, shape, and machine learning based methods.

Moving skin pixels were detected in video streaming and Mean-shift algorithm was adopted to detect hand in [5]. The method performed well as long as non-skin objects appeared in the scene. The derived motion, skin color and morphological information were combined to detect hands in [9]. Morphological features were used to estimate the probability of a pixel belonging to the hand region in the current frame. The method can detect hands indoors in real time and the similar method was adopted in [10]. Dardas et. al first subtracted the face region using Viola and Jones method [8] and detected the remained region using a skin detector and hand gesture contour comparison algorithm [6,7]. The algorithm detected only four simply defined hand gestures in real time.

The performance of skin color or motion information based methods are restricted by strong assumptions and isn't robust in actual applications. Some methods combined the skin color with machine learning were proposed. Eng-Jon and Richard presented an unsupervised approach and detected the locations of the hands using a boosted cascade of classifiers in grey scale images, in which provided good detection accuracy [11]. Wu and Huang proposed an approach called Discriminant-EM (D-EM) [12] to help supervised learning reduce the number of labeled training samples, but the method can't address the issues of complex backgrounds. Zondag and Gritti et al. constructed a real-time hand detector using HOG features [4] in combination with two variations of the AdaBoost algorithm [13]. Liew and Yairi focused on an appearance approach and proposed a feature extraction method based on sparse pixel-pairwise intensity comparisons for hand detection [14]. The method was robust against image noises, cluttered backgrounds, and partial occlusion. Mittal and Zisserman et al. first detected possible hand gesture using a hand shape detector, a context based detector, and a skin based detector respectively. A second stage classifier was learnt to compute a final confidence score for hand detection [15]. The method was time consuming although it can achieve very good recall and precision. Pisharady and Vadakkepat et al. utilized a biologically inspired approach based on the computational model of visual cortex and a Bayesian model of visual attention to generate the saliency map by calculating the posterior probabilities of pixel locations to be part of a hand gesture. The hand gesture was extracted by segmenting input image after setting threshold value of the saliency map. The method provided a good hand detection accuracy in spite of complex backgrounds. The disadvantages were slow processing speed and high computational complexity.

Some researchers introduced general object recognition methods to detect hands. The most commonly used detection methods include rigid templates [4] or bag of features [17], which performs poorly on hand detection owing to hand's variable appearance and the wide range of hand gestures. It's obvious that an “elastic” or “deformable” model to detect hands is necessary. Felzenszwalb and Huttenlocher developed a multi-scale deformable part model in [18] and the mixture of the deformable part models in [19]. The mixture of the deformable part models can capture significant variations in appearance and was often expressive enough to represent a rich object category.

In consideration of the success of the mixture of deformable part models on human detection, we develop a hand detection method based on the mixture model. A mixture model including three components each defined by a root filter and several part filters is trained. A meaningful gesture consists of a palm, some fingers and the joint of palm and fingers, so we define three part filters corresponding to each root filter in hand detection task.

3 Overview of the Proposed Method

We propose a hand detection method based on a mixture of multi-scale deformable part models. The mixture model is trained using a discriminative procedure that only requires bounding box labels for the positive examples.

3.1 HOG Features

Skin features [5, 6, 7, 8, 15], shape context features [6, 7], Haar-like features [8, 10,13], SIFT features [6,7], morphological features [9], biologically inspired features (C2 features) [3], HOG features [4] or the combinations of these features [13,15] have been used for hand detection in recent years. HOG features are originally proposed by Dalal and Triggs in [4] for human detection. We describe a variation of HOG features of an image at a particular resolution following the construction in [4].

We compute gradients using finite difference filters, [-1, 0, +1] and its transpose. The image is first divided into 8×8 non-overlapping pixel regions called cells. For each cell, we accumulate a 1-dimensional HOG feature set over pixels in that cell. However, the feature set of a cell is a little different from that in [4] and [18]. In this paper, in order to capture information of hand gestures as much as possible, a 31-dimensional feature set is constructed which includes 18-dimensional contrast sensitive features, 9-dimensional contrast insensitive features and 4-dimensional magnitude features from the reconstruction of 36-dimensional features in [18].

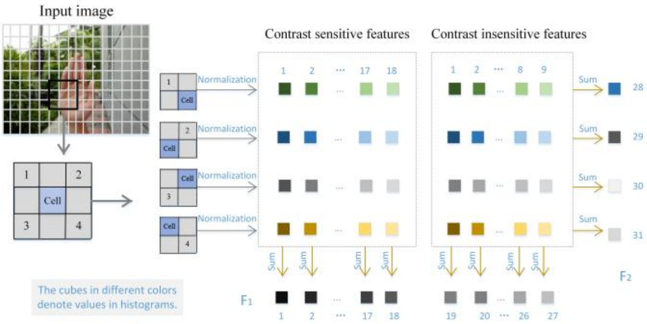

Input image
1
2
Cell
3
4
Normalization
Contrast sensitive features
Contrast insensitive features
1 2 ··· 17 18
1 2 ··· 8 9
Sum
28
29
30
31
F2
F1
1 2 ··· 17 18
19 20 ··· 26 27
The cubes in different colors denote values in histograms.

Fig. 2. Representations of 31-dimensional HOG features of a cell

For each cell in an image, the gradient orientation of each pixel is firstly discretized into eighteen orientation bins. The contribution of each pixel to gradient orientation depends on the gradient magnitude. Then the cell is normalized at four 2×2 neighborhood cells called a block with respect to the total energy of each block (see Fig.2). We sum over different normalizations and get an 18-dimensional contrast sensitive feature set . Then the same method is used but each pixel is discretized into nine orientation bins which leads to a vector of length 9×4 representing the local gradient information inside a cell. Not only the sum over different normalizations but also the sum over nine contrast insensitive orientations are computed, which reduce the 9×4 vector into a 13-dimensional feature set . The final HOG feature set is .

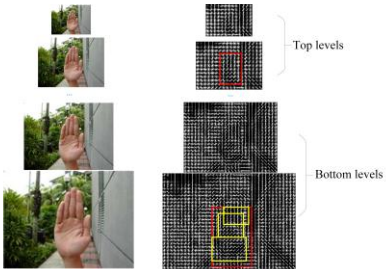


Fig. 3. HOG feature pyramid for and hand gesture hypothesis

The new features are low-dimensional and can capture local appearance which are invariant to small deformations. For a color image, the gradient of each color channel is computed and the highest gradient magnitude is picked as the final value.

A standard HOG feature pyramid is built for multi-scale hand detection. The feature pyramid consists of several couple levels, the resolution of each bottom level is twice the corresponding top resolution as shown in Fig.3. The top level HOG features represent coarse information such as hand contour while the corresponding bottom level HOG features represent finer information such as fingers or the palm in different states.

3.2 Hand Model

The mixture model involves three components each defined by a coarse root filter covering an entire hand gesture and three finer part filters covering smaller parts such as fingers or the palm. The filters in the mixture model are applied to the HOG feature pyramid to calculate the responses of an input image. We require the level of each part is such that the feature map at that level is computed at twice the resolution of the root level (see Fig.3).

The score of at a position in a feature map from the HOG feature pyramid is the "dot product" of the filter and a sub-window of the feature map with top-left corner at , while the filter is a matrix with weights.

(1)

We denote the filter as . The score of at in a feature pyramid is , written as later for convenience, where denotes the HOG features in the sub-window.

The goal is to get the best placement of root filter and part filters in a component, where specifies the position for th filter in the feature pyramid. The score of a placement is given by the scores of each filter minus a deformation cost that depends on the relative position of each part filter with respect to the root filter, plus the bias,

(2)

Where the deformation cost and are defined by [17].The formula (2) can be written as dot product, , where specifies latent information of hand's parts.

. (3)

(4)

This illustrates the connection between the model and linear classifiers. Latent SVM [17] is used to train our hand model for the partially labeled data. For the mixture model, three components are combined and the mixture model with the similar expression as the component is trained.

Learning. The process of learning model parameters using latent SVM is described in the following parts and more details can be found in [19]. The objective function is:

. (5)

is trained from labeled examples , where is an example with a binary label . Generally is not convex for a positive example and is convex for a negative example. However, becomes convex for each example if there is a single possible latent value for each positive example. Note that,

. (6)

Which means and we can minimize using coordinate descent algorithm: Optimize over by selecting the highest scoring latent value for each positive example and over using stochastic gradient descent algorithm. However, the stochastic descent algorithm is so sensitive to local minima that the mixture model is initialized as in [19].

There are very large numbers of negative examples in an image. It's reasonable to construct training data consisting of the labeled positive examples and "hard negative" examples. The hard negative examples are those that are incorrectly classified or inside the margin of the classifier defined by in the previous training.

Detection. An overall score is computed for each root location according to the best possible placement of the parts for hand detection,

. (7)

The root locations with high score define detections, and part locations with respect to root locations define a full hand hypothesis. Dynamic programming and generalized distance transforms methods [20] are used to compute the best root location. The response of each filter to the feature pyramid is first computed as follows,

(8)

Then the responses of part filters are transformed using generalized distance transform algorithm.

. (9)

The final root scores at each level can be expressed by the sum of the root filter response at that level with transformed responses at such level that the resolution of the level is twice the root detection level.

$$. \tag{10}$$

The hands are detected in a feature pyramid of an input image. There is a hand in an image if the score is higher than a threshold. The desired output is to predict the bounding boxes of a hand gesture. We eliminate the repeated detections by non-maximum suppression (NMS). The final root scores of labeled positive examples are used to learn four linear functions for predicting the bounding box by least-squares regression (LSR). Fig. 4 illustrates the detection process using a component of the mixture model.

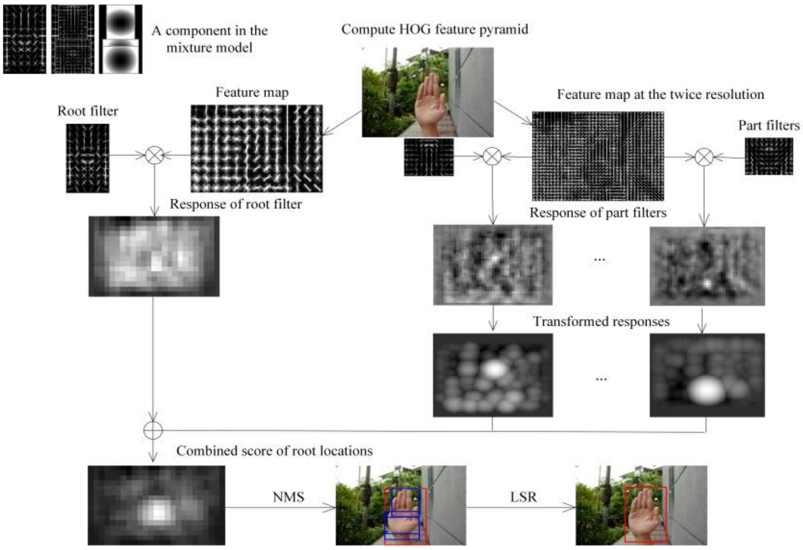

A component in the mixture model
Compute HOG feature pyramid
Feature map
Feature map at the twice resolution
Root filter
Part filters
Response of root filter
Response of part filters
...
Transformed responses
...
Combined score of root locations
NMS
LSR

Fig. 4. The detection process using a component at one scale

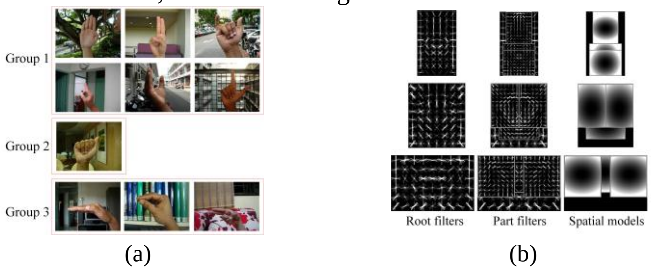
4 Experimental Results

The mixture model of hand gesture is trained and evaluated on NUS hand gesture dataset - II [16], (see Table 1).

Table 1. NUS hand gesture dataset- II

Subsets	Descriptions
A	2000 hand gesture color images
B	750 hand gesture color images with human noises
C	2000 background images without the hand gestures

In practice, all positive examples from subsets A and B are labeled with bounding boxes covering the entire hand gestures. The part locations are treated as latent variables during the training process using latent SVM. The positive example set is composed of half original positive examples from subset A and the corresponding flipped positive examples. Margin sensitive method [19] is used to mine hard negative examples from subset C. The positive example set is split into three groups according to the orientations of hand gestures (see Fig. 5(a)) which lead to a mixture model with three components (see Fig. 5(b)). The operation is different from [19] whose positive examples are split according the aspect ratio of the bounding boxes. The detection accuracy is improved by performing the operation. The proposed mixture model is symmetric along the vertical axis, so it can detect gestures of either hand.



(a) (b)

Fig. 5. The mixture model. (a) The split positive examples with three groups. (b)The mixture model with three components.

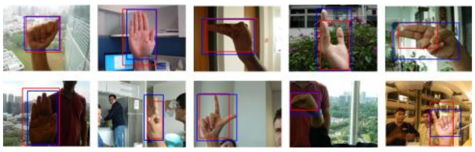
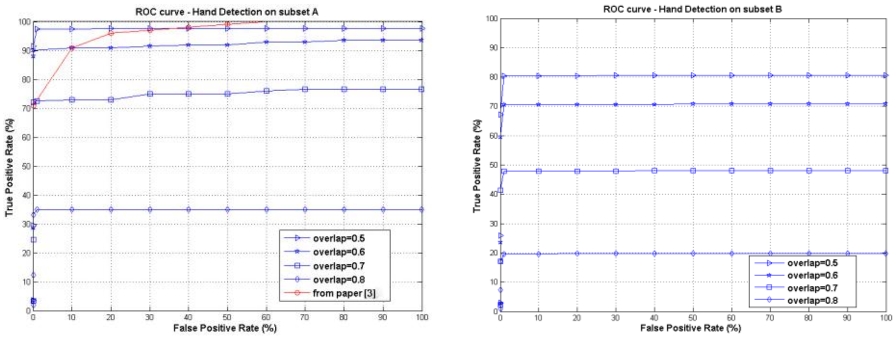


Fig. 6. Hand detection results. The first row shows some results from subset A while the second row from subset B. The prediction bounding boxes are in red while the ground-truth bounding boxes are in blue.

The mixture model is tested on the rest of images from NUS hand gesture dataset-II. Fig. 6 shows some examples of hand gesture detection using the mixture model. The mixture model has the best performance on images from the first three columns in which the prediction bounding boxes has a large overlap with the labeled bounding boxes. Although the mixture model can detect hand in the images from the last two columns, it fails to capture all regions of the hand gestures.



(a) (b)

Fig. 7. ROC curve of hand detection. (a) ROC curve of subset A without human noises (b) ROC curve of subset B with human noises.

The subsets A and C are used to test the capability of the mixture model as in [3]. The presence of hand is detected if the score of the best placement is above the threshold. Fig. 7(a) shows the ROC of the hand detection task. The effects of overlaps on detection performance are also studied as shown in Fig. 7(a). The overlap between the prediction bounding box and a ground-truth bounding box is an important index in hand detection. The proposed hand detection model is also tested on a harder subset B with human noise and subset C (see Fig. 7(b)).

It can be seen from the experimental results that the proposed mixture model performs better on subset A when the overlap is 0.5 compared to [3] and can detect hand correctly at a low false positive rate. In general, the mixture model is better than the method from [3]. Since the performance is regarded to be better if the overlap is higher in the hand detection system, the performance decreases with the increase of overlaps. However, it's sufficient for the capture of all the information of a hand gesture when the overlap is 0.6. The presence of human may reduce the detection performance as shown in Fig 7(b). The mixture model's detection time is about 0.5s, which is less compared to the biologically inspired approach in [3].

5 Conclusions

A hand detection method based on the mixture of deformable part models is proposed in this paper. More components and part filters in the mixture model will improve the performance, but may lead to time consumption and expensive detection task. The mixture model is robust to hand shapes, viewpoints, lights or complex backgrounds and invariant to the small deformations of hand gestures to an extent. The performance of the mixture model on the harder dataset including influence of human needs to be improved. The richer mixture model combining skin color or building connections among parts is helpful to improve the detection accuracy. The higher detection accuracy will produce the better recognition results in hand gesture recognition system. A well-defined hand gesture set for interaction should be designed and a robust hand gesture recognition system should be studied in the future work.

Acknowledgements

This research was supported by the National Natural Science Foundation of China under grant No. 61370134, the National High Technology Research and Development Program of China (863 Program) under grant No. 2013AA013904.

References

1. Zhu, Y., Yang, Z., Yuan, B.: Vision Based Hand Gesture Recognition. In: 2013 International Conference on Service Sciences, pp. 260-265. IEEE Press, Shen Zhen (2013)
2. Yu, C., Wang, X., Huang, H., Shen, J.: Vision-Based Hand Gesture Recognition Using Combinational Features. In: 2010 Sixth International Conference on Intelligent Information Hiding and Multimedia Signal Processing, pp. 543-546. IEEE Press, Darmstadt (2010)
3. Pisharady, P.K., Vadakkepat, P., Loh, A.P.: Attention Based Detection and Recognition of Hand Postures against Complex Backgrounds. International Journal of Computer Vision, 101, 403–419 (2013)
4. Dalal, N., Triggs, B.: Histograms of Oriented Gradients for Human Detection. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 886-893. IEEE Press, San Diego (2005)
5. Farhad, D., Abdolhossein, S., Chris, M.: Multi-layered Hand and Face Tracking for Real-Time Gesture Recognition. In: 15th International Conference on Advances in Neuro-Information Processing, pp. 587–594. Springer Press, Auckland (2009)
6. Dardas, N.H., Georganas, N.D.: Real-time Hand Gesture Detection and Recognition using Bag-of-Features and Support Vector Machine Techniques. IEEE Transactions on Instrumentation and Measurement, 60 (11), 3592–3607 (2011)
7. Dardas, N.H., Petriu, E.M.: Hand Gesture Detection and Recognition Using Principal Component Analysis. In: 2011 IEEE International Conference on Computational Intelligence for Measurement Systems and Applications, pp. 1-6. IEEE Press, Ottawa (2011)
8. Viola, P., Jones, M.J.: Robust Real-time Object Detection. International Journal of Computer Vision, 2(57), 137-154 (2004)
9. Stergiopoulou, E., Sgouropoulos, K., Nikolaou, N., Papamarkos, N.: Real Time Hand Detection in a Complex Background. Engineering Applications of Artificial Intelligence, 35, 54-70 (2014)
10. Fang, Y., Wang k., Cheng J., Lu, H., C.: A Real-Time Hand Gesture Recognition Method. In: 2007 IEEE International Conference on Multimedia and Expo, pp. 995 – 998. IEEE Press, Beijing (2007)
11. Ong, E.J., Bowden, R.: A Boosted Classifier Tree for Hand Shape Detection. In: Sixth IEEE International Conference on Automatic Face and Gesture Recognition, pp. 889-894. IEEE press, Jeju Island (2006)
12. Wu, Y., Huang, T. S.: View-independent Recognition of Hand Postures. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 88-94. IEEE Press, Hilton Head Island (2000)
13. Zondag, J. A., Gritti, T., Jeanne V.: Practical Study on Real-time Hand Detection. In: 2009 3rd International Conference on Affective Computing and Intelligent Interaction and Workshops, pp. 1-8. IEEE Press, Amsterdam (2009)
14. Liew, C.F., Yairi, T.: Generalized BRIEF: A Novel Fast Feature Extraction Method for Robust Hand Detection. In: 2014 22nd International Conference on Pattern Recognition, pp. 3014-3019. IEEE Press, Stockholm (2014)
15. Mittal, A., Zisserman, A., Torr, P.H.S.: Hand Detection using Multiple Proposals. In: 2011 British Machine Vision Conference, pp. 75.1-75.11. BMVA Press, Scotland (2011)
16. NUS Hand Posture Datasets, https://www.ece.nus.edu.sg/stfpage/elepv/NUS-HandSet
17. Zhang, J.G., Marszalek, M., Lazebnik S., Schmid, C.: Local Features and Kernels for Classification of Texture and Object Categories: A Comprehensive Study. International Journal of Computer Vision, 73(2), 213-238 (2007)

18. Felzenszwalb, P.F., McAllester, D., Ramanan, D.: A Discriminatively Trained, Multiscale, Deformable Part Model. In: 2008 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1-8. IEEE Press, Anchorage (2008)
19. Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object Detection with Discriminatively Trained Part-Based Models. IEEE Transactions on Pattern Analysis and Machine Intelligence, 32(9), 1627-1645 (2010)
20. Felzenszwalb, P.F., Huttenlocher D.: Distance Transforms of Sampled Functions. Technical Report 2004-1963, Cornell Univ. CIS (2004)