Event detection: IPG-BJTU at Trecvid 2010

Yuan Shen, Ping Guo, Shu Wang, Lin Yang, Haifeng Deng, Liang Liang, Zhenjiang Miao
Institute of Information Science, Beijing Jiaotong University
{08112074, 06120385, 09112083, 09120413, 09120465, 10120392, zjmiao}@bjtu.edu.cn

Abstract:

In trecvid 2010, our team takes part in 4 event detection competition including embracing, pointing, object put and cell to ear. We build four systems to recognize these events separately. For embracing, we use a probability accumulated method. For pointing, we use motion energy image and rules to recognize this action. For object put, we use probability rule. And for cell to ear, we use orientation of optical flow and SVM classifier to finish this work. In the experiment, two actions of our work obtain good performance.

1. Introduction

Human action recognition is one of the most challenging problems in computer vision. The focus of this problem is mainly reliability and effectiveness. However, in Trecvid dataset, it is more challenging than any other datasets, because the number of people in the scene and occlusion. Until now, many approaches have been presented for human action recognition.

One of the main approaches of recognition is dynamic models. Yamato et al. [1] used the Hidden Markov Models (HMM) as recognition model for human action recognition. Laxton et al. [2] used a Dynamic Bayesian Network to recognize human action.

Another main approach of recognition is spatio-temporal template. Bobick and Davis [3] introduced Motion-Energy-Image (MEI) and Motion-History-Image (MHI) templates for recognizing different motions. From then on, spatio-temporal templates were made famous on human action recognition. Efros et al. [4] used a motion descriptor based on optical flow measurements in a spatio-temporal volume to represent actions and used nearest-neighbor to classify actions. Blank et al. [5] defined actions as space-time shapes, and used Poisson distribution to represent the details of such shapes. Jhuang et al. [6] applied biological model of motion processing for action recognition using optical flow and space-time gradient feature.

In recent years, space-time interest points feature and “bag of words” model are widely used in human action recognition studies. Laptev et al. [7] first introduced the notion of “space-time interest points”. Piotr Dollar et al. [8] used 2-D Gauss filter and 1-D Gabor filter to extract space-time interest points for human action recognition. Popular topic models include pLSA [9], LDA [10]. Juan Carlos Niebles et al. [11] extracted space-time interest points feature and they perform unsupervised learning of action categories using pLSA model and LDA model separately. Yang Wang and Greg Mori [12] used optical flow method to extract motion feature and used latent topic models to do recognition. However, extracting space-time interest points need much computation time and topic models ignore the spatial and temporal information

In trecvid 2010, we take part in 4 event detection competition including embracing, pointing, object put and cell to ear. We build four systems to recognize these events separately. For embracing, we use a probability accumulated method. For pointing, we use motion energy image and rules to recognize this action. For

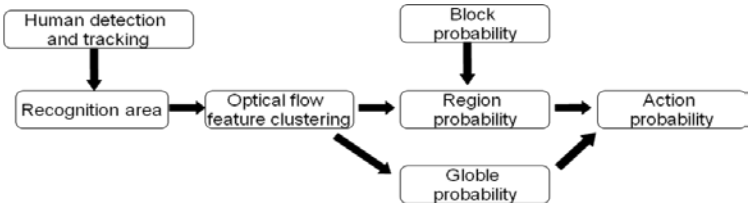

Figure 1. The flow of embracing recognition

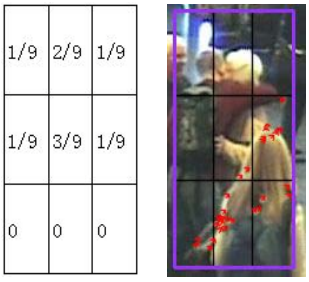

Figure 2. Block probability

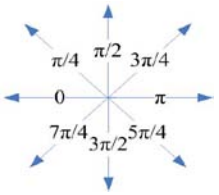

Figure 3. Eight bins for optical flow features

object put, we use probability rule. And for cell to ear, we use orientation of optical flow and SVM classifier to finish this work.

The rest parts of this paper are organized as the following: Section 2 introduces approaches of four event detection systems. Section 3 will show the performance of our method in trecvid 2010. The conclusions are given in section 4.

2. Our approaches

Before event detection, we must detect and track human. In this study, we use trecvid dataset to train a HOG-SVM [13] human detection model and use mean-shift [14] to track people.

2.1 Recognition of embracing

For action embracing, it needs several persons to interaction each other to complete this action. So we first detect and track people for each frame. When two persons are close to each other, we assume that these two persons maybe begin embracing. However, we only detect the embracing actions which are standing still, not a moving person. So, when we find the people maybe occur embracing and they stand still, we can start the process of embracing recognition. This is the first step that we need to find the space that it maybe occurs embracing action. The overall flow of this recognition is as shown in figure 1.

When we obtain the recognition area, we divide this area into nine blocks. Each block has a local probability as show in figure 2. We compute optical flow feature in this area and quantize the orientation of each feature into eight bins as shown in figure 3. Then we propose an algorithm to cluster these features based on the orientation and distance of optical flow features.

We calculate the distance of starting points of optical flow features by Euclidean distance. Before clustering, we introduce the meaning of some variables. The variable Q_1 is the set of optical flow features.

The variable Q_2 is a temporary queue of optical flow features. The variable TAG and CLASSNUM are the attribute of each optical flow vector, CLASSNUM records the class number of each optical flow vector, TAG records whether an optical flow vector is classified correctly. The variable numC records the class number in the process. The process of clustering is as follow:

Algorithm 1: Find local motion

```
Initialize numC=-1
For (B=Q1.begin; B != Q1.end;B++)
    If (Q2 is empty)
        Select B which TAG is flase, add to Q2
        numC=numC+1
        TAGB is true and CLASSNUMB is numC
    End If
    While (Q2 is not empty)
        A = Q2 get first element
      For ( C= Q1.begin; C != Q1.end; C++)
            If (distance (A, C) is near and
                orientation (A,C) is same and TAGC is false)
                CLASSNUMC = CLASSNUMA
                TAGC is true, add C to Q2
            End If
      End For
        Q2 erase first element
    End While
End For
```

After clustering, optical flow features are divided into several regions. Each region can represent a local motion. Based on the local probability in figure 2, we start to compute the region probability. For computing embracing starting, we only use the features whose orientations belong to [π/4, π/2, 3π/4]. For computing embracing ending, we only use the features whose orientations belong to [5π/4, 3π/2, 7π/4]. When an optical flow feature f_i is belong to block B_j, the probability of f_i can be represented by the probability of B_j. For a region R_k, its probability can be represented by equation (1), N_{Rk} is the feature number of the region R_k.

$$P_{Rk} = \frac{\sum_{f_i \in B_j} P_{Bj}}{N_{Rk}} \tag{1}$$

The global probability of embracing can be represented by equation (2).

$$P = \frac{N}{N_{all}} \tag{2}$$

For computing embracing starting, N can be replaced by Ns. Ns is the number of optical flow features whose orientations belong to [π/4, π/2, 3π/4]. For computing embracing ending, N can be replaced by Ne. Ne is the number of optical flow features whose orientations belong to [5π/4, 3π/2, 7π/4]. N_{all} is the total number of optical flow features.

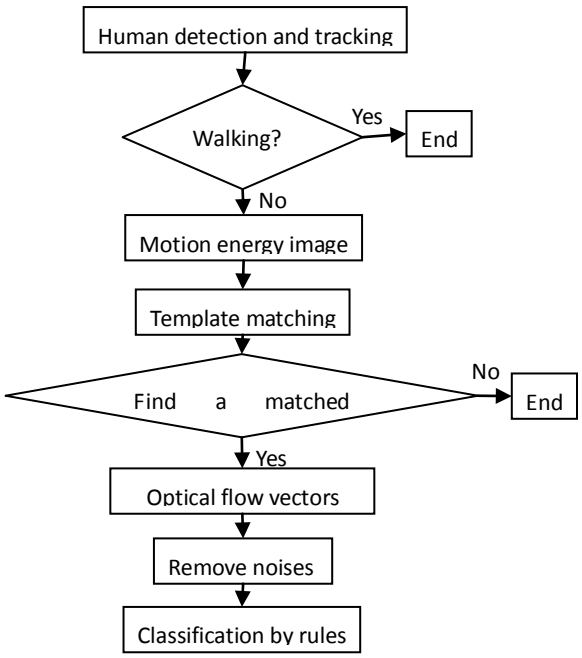

Figure 4. The flow of pointing recognition

Fighre 5. Illustration of the candidate windows.

So the probability of embracing starting or ending can be represented by equation (3).

$$P_{embrace} = P_{Rk} \times P \tag{3}$$

At last, we accumulate the starting probability for several frames. When it exceeds a threshold, we consider that the embracing action starts. After we detect the starting of embracing, we accumulate the ending probability for several frames. When it exceeds a threshold, we consider that the embracing action ends.

2.2 Recognition of pointing

The flowchart of our system is shown in figure.4. We assume that people who do the pointing event are basically standing still. Therefore, we first detect and track each human. If the person is not walking, the pointing detection system starts.

In the annotation document, it says that the pointing event does not necessarily begin when the persons raise their arm to point. However, our system considers only the raising arm to point for simplicity. Given a rectangular box for each person by human detection, we drop two candidate regions for two directions of pointing. Each candidate region is 1.5 times of the human width by the human height, as shown in figure.5.

In order to get the moving area of the raising arm, motion history image (MHI) [3] of one training video is computed. Then we manually choose three MHIs for each direction as evaluation masks, which are use to cover the candidate regions of the raising arm, as shown in fighre.6(a).

In the testing stage, MHI of the testing video is first computed, and then we choose the most matched mask for each testing frame by template matching. Optical flows are computed as low level features. Similarly to the MHI, we define a flow history image (FHI) in our system. The optical flows which are not covered by the matched mask are removed as noises, as shown in figure.6. Since we do not know the length of each event, a sliding temporal window with a const length T is used. The probability of pointing for each sliding window is computed by the following rules:

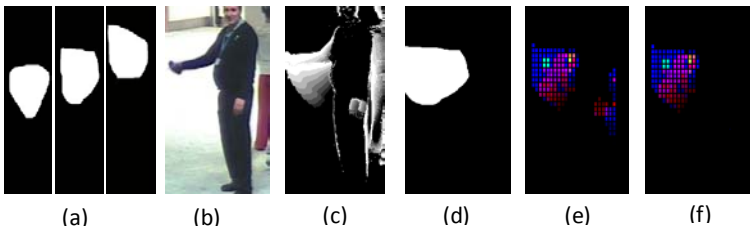

Figure 6. (a)Three masks for pointing left. (b)Input video.(c)MHI.(d)The aligned matched template. (e)FHI. Different colors represent different moving directions. Blue: up. Green:right. Red: left (f)FHI after noise

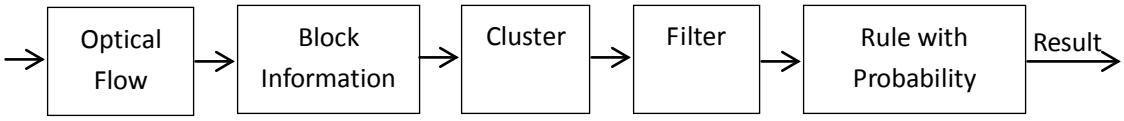

Figure 7. The flow of object put recognition

$$p1=FHI\_up/(FHI\_left+FHI\_up)$$
$$p2=(FHI\_left+FHI\_up)/FHI \quad (4)$$
$$p=(\lambda*p1+(1-\lambda)*p2)$$

If a pointing event happens at time t, there will be multiple detections at adjacent time of t. In our system, the number of detections in adjacent time is an important parameter to recognize pointing event. Suppose the number of continuous detections in adjacent time of t is n, the final probability of pointing is computed by:

$$P\_final=p3*n/\theta \quad (5)$$

where λ and θ are const parameters. In our system, λ=0.5 and θ=5.

When P_final is larger than a threshold, the time is defined to be the start time of a pointing event. After we detect the start time, if P_final is smaller than a threshold, the end time of the event is detected.

2.3 Recognition of object put

Our basic idea is detect downward motion tendency and calculate confidence according to the pattern of this tendency. Our flowchart is as shown in figure 7.

Optical flow is computed as our low-level features to describe person's motion pattern. Because the complexity of scene, some optical flow noises may be existed using state of art optical flow technology, to reduce the impact of these noises, we divide whole person rectangle into small blocks, and compute the mean velocity and direction in each block. In order to make the motion pattern description more robust and representative, we cluster similar blocks into one cluster according to its velocity, direction and location. Because the number of clusters is not known before, so an online cluster method is used to achieve this goal. We initial one block as first cluster, for the rest blocks, we calculate the distance between it and existed clusters as follows:

$$\mathrm{d}_{i,j} = (v_i - v_j)^2 + (d_i - d_j)^2 \quad (6)$$

Where v_i and d_i is the velocity and direction of block i respectively, and v_j and d_j is the velocity

How many frames was downward cluster lasted?
How the location of downward clusters changing?
Where is the location of downward clusters?
Is there any upward cluster show up after downward cluster?
What shape is the downward cluster?
What is the direction of cluster which is next to downward cluster?

Figure 8. Object put rules

and direction of cluster j respectively.

If the distance is larger than the threshold, the block is considered as a new cluster; otherwise we think it belongs to the nearest cluster. It is to be noted that this distance calculation is only between adjacent block and cluster, if they are not adjacent, the block is considered as a new cluster no matter how likely they are.

After this process, motion pattern is represented as distribution of some clusters. Because we only consider obvious Object Put event, before this distribution information is utilized, some clusters should be discarded, which contain very few blocks, to make clusters more representative and minimize the interference from subtle movement.

Now we can infer the event by analyzing the distribution of clusters. There is always downward motion tendency in Object Put event, so clusters which direction is downward are our key point. As we discussed, it is true that when Object Put event happening, there must be some downward clusters, but there may be downward clusters in many other events or even from optical flow error, so we must set some rules to distinct Object Put event from others. As we analyze the distribution of downward clusters from training date, we found some specific pattern of Object Put event. Rules are relevant to questions as shown in figure 8

According to each answer of these questions, we can get a probability based on particular rule, and then accumulate all probability into a final confidence.

2.4 Recognition of cell to ear

A cell to ear event is a small-scale action compared to other events. Because the mobile phone is very small compared to a person, it is very difficult to detect the object. So, we try to describe the action with the action of hand.

Here, we choose motion vector to represent the action. We have tried the traditional Lucas–Kanade algorithm to calculate optical flow. But we found this method will induce much noise. So I consider the block matching algorithm to calculate the motion vector. The method mainly includes two steps. First, we partition each frame of a given sequence into fixed-number blocks. Then, detect blocks displacement between the actual frame and the previous one, searching inside a given scan area. It provides a field of displacement vectors associated with. Each block defines in the previous frame, a "scan area", centered in the block center. The block is shifted pixel-by-pixel inside the scan area, calculating a match measure at each shift position. The comparison is aimed at determining the pixel set most similar to the block between its possible positions in the scan area. Among these positions, the scan area subpart defined by its center, will be the matrix with the best match measure. We use the summation of the absolute difference as the matching rule. And improve the accuracy by adding a threshold constrict and a low-pass filter and get the final motion vector, Figure 9.

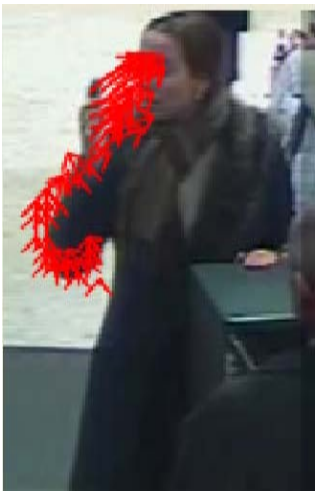

Figure 9. Optical flow for cell to ear

Analysis Report	#Ref	#Sys	#CorDet	#FA	#Miss	Act. RFA	Act. PMiss	Act. DCR	Min RFA	Min PMiss	Min DCR
ObjectPut	621	8	1	7	620	0.459	0.998	1.001	0.328	0.998	1.000
Embrace	175	64	9	55	166	3.607	0.949	0.967	3.542	0.949	0.966
Pointing	1063	113	10	26	1053	1.705	0.991	0.999	0.131	0.995	0.996
CellToEar	194	0	0	0	194	0.000	1.000	1.000	0.000	1.000	1.000

Figure 10. Event detection result table

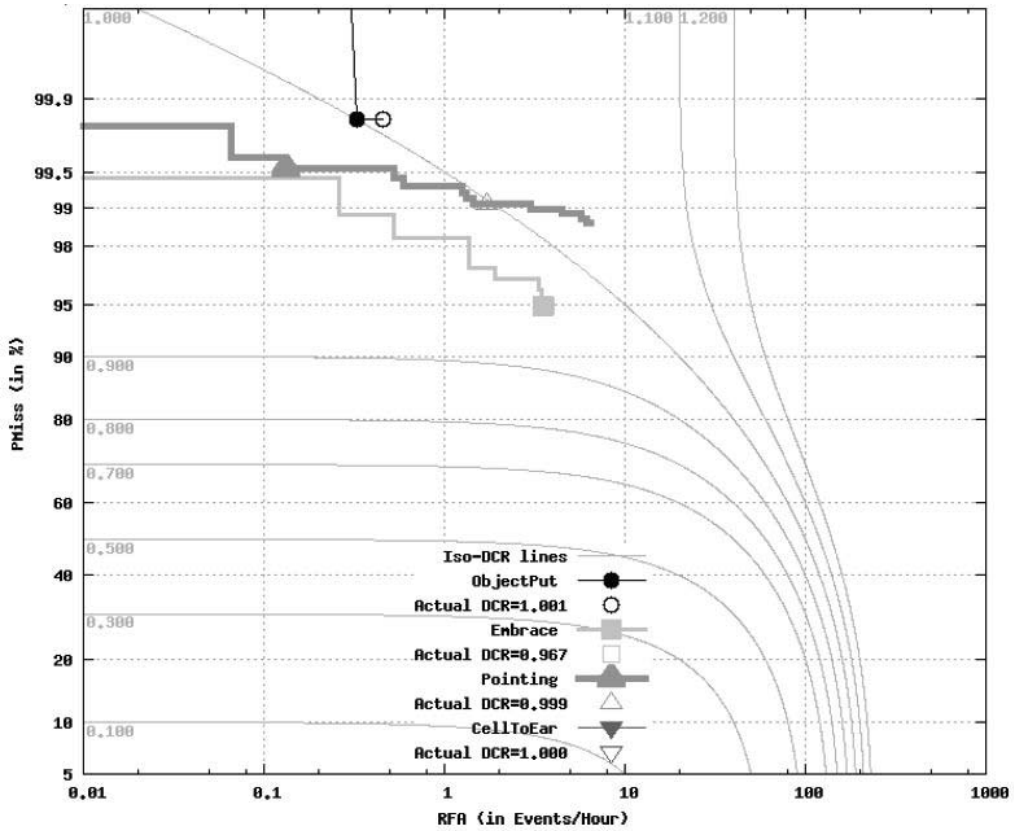

Figure 11. Event detection result graph

We do the calculate method for each frame as above. Considering the complex background, we suppose the lower half of the person who is doing the cell to ear action is still. And the motion vector of the

higher part must have a upward component. So we refine the motion vectors with the assumption. And we crop a cube, which is still on space dimensions for a certain period (7,8,or a certain frames), on the basis of the detecting results. Then, we obtain a motion vector from each frame of the cube. Finally, combine all the motion vectors to get a final feature vector for a cube. The vector will be deal with SVM and detect whether the cube includes the cell to ear event. If several cubes are all time-adjacent, and all include the event, we will link them. And the first frame of the first cube is defined the start of the event and the last frame of the last cube is defined the end of the event.

3. Experiments

We submit four action detection results. In figure 10 and 11, we show the event detection results.From the result, we can see that we have obtained a good performance in action embracing and pointing. However, the action object put and cell to ear is relatively low. We consider that in such a crowed scene, rules and SVM classifier model do not have enough discrimination ability.

4. Conclusions

In trecvid 2010, our team takes part in 4 event detection competition including embracing, pointing, object put and cell to ear. We build four systems to recognize these events separately. These four systems use different approaches to recognition actions. There are probability accumulated method, motion energy image and rules, probability rules, orientation of optical flow and SVM classifier. In the experiment, the approaches for embracing and pointing obtain a good performance. Due to the rules and SVM classifier model do not have enough discrimination ability, the action object put and cell to ear is relatively low.

Reference

[1] J.Yamato, J.Ohya, and K.Ishii. Recognizing human action in time-sequential images using hidden Markov model. In Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (Champaign IL, June 1992). CVPR '92, 379-385.

[2] B. Laxton, J. Lim, and D. Kriegman. Leveraging temporal, contextual and ordering constraints for recognizing complex activities in video. In Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (Minneapolis MN,, June 2007). CVPR'07, 1-8.

[3] A.F.Bobick and J.W.Davis. The recognition of human movement using temporal templates. IEEE Transactions on Pattern Analysis and Machine Intelligence. 23, 3 (March 2001) 257-267.

[4] A.A.Efros, A.C.Berg, G.Mori, and J.Malik. Recognizing action at a distance. In Proceedings of the IEEE 9th International Conference on Computer Vision (Nice France, 2003). ICCV'03, Vol.2, 726-733.

[5] M.Blank, L.Gorelick, E.Shechtman, M.Irani, and R.Basri. Actions as space-time shapes. In Proceedings of the IEEE 10th International Conference on Computer Vision (Beijing, 2005). ICCV'05, Vol.2, 1395-1402.

[6] H.Jhuang, T.Serre, L.Wolf, and T.Poggio. A biologically inspired system for action recognition. In Proceedings of the IEEE 11th International Conference on Computer Vision (Rio de Janeiro, October 2007). ICCV'07, 1-8.

[7] I.Laptev and T.Lindeberg. Space-time interest points. In Proceedings of the IEEE 9th International Conference on Computer Vision (Nice France, 432-439, 2003). ICCV'03, Vol.1, 432-439.

[8] P.Dollar, V.Rabaud, G.Cottrell, and S.Belongie. Behavior recognition via sparse spatio-temporal features. In Proceedings of the IEEE workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance. (October 2005). VS-PETS'05, 65-72.

[9] T.Hofmann. Probabilistic latent semantic indexing. In Proceedings of the 22nd annual international ACM SIGIR conference on research and development in information retrieval (California US, August 1999). ACM Press, New York, NY, 50-57,

[10] D.M.Blei, A.Y.Ng, and M.I.Jordan. Latent dirichlet allocation. Journal of Machine Learning Research. 3 (2003) 993-1022.

[11] Juan Carlos Niebles, Hongcheng Wang, and Fei-Fei Li. Unsupervised learning of human action categories using spatial-temporal words. International Journal of Computer Vision. 79, 3 (2008) 299-318.

[12] Yang Wang and G.Mori. Human action recognition by semilatent topic models. IEEE Transactions on Pattern Analysis and Machine Intelligence. 31, 10 (Oct. 2009) 1762-1774.

[13] N.Dalal and B.triggs. Histograms of oriented gradients for human detection, IEEE Conference on Computer Vision and Pattern Recognition, (2005),1-8.

[14] D.Comaniciu, V.Ramesh and P.Meer, “real-time tracking of non-rigid objects using mean shift”, IEEE Conference on Computer Vision and Pattern Recognition, (2000), 673-678.