Human Action Recognition by Fast Dense Trajectories

Zongbo Hao
University of Electronic Science and Technology of China
No.4 Section2, North Jianshe Road, 610054, Chengdu, Sichuan, China
Tel: +86 28 6183 0580
zbhao@uestc.edu.cn

Qianni Zhang
Queen Mary, University of London
Mile End Road
London E1 4NS, U.K.
Tel: +44 20 7882 7138
qianni.zhang@eecs.qmul.ac.uk

Ebroul Ezquierdo
Queen Mary, University of London
Mile End Road
London E1 4NS, U.K.
Tel: +44 20 7882 5354
ebroul.ezquierdo@eecs.-qmul.ac.uk

Nan Sang
University of Electronic Science and Technology of China
No.4 Section2, North Jianshe Road, 610054, Chengdu, Sichuan, China
Tel: +86 28 6183 0581
sn@uestc.edu.cn

ABSTRACT
In this paper, we propose the fast dense trajectories algorithm for human action recognition. Dense trajectories are robust to fast irregular motions and outperform other state-of-the-art descriptors such as KLT tracker or SIFT descriptors. However, the use of dense trajectories is time consuming. To improve the efficiency, we extract feature trajectories in the ROI rather than in the whole frames, and we use the temporal pyramids to achieve adaptable mechanism for different action speed. We evaluate the method on the dataset of Huawei/3DLife – 3D human reconstruction and action recognition Grand Challenge in ACM Multimedia 2013. Experimental results show a significant improvement over the dense trajectories descriptor in real-time, and adaptable to different speed.

Categories and Subject Descriptors
I.2.10 [**Vision and Scene Understanding**]: Video analysis; I.4.8 **[Scene Analysis]**: motion

Keywords
Fast dense trajectories; human action recognition; adaptable to different speed; real-time

1. INTRODUCTION
The recognition of human action and motion using computer vision has recently gained more and more interest in recent years [14][10][7]. The key point is detecting, tracking, recognizing humans from the video by the computer vision, and then understanding and characterizing their actions [16]. Action recognition has broad application prospect and potential economic value in the field such as intelligent surveillance, smart appliances, human machine interface, and content-based image retrieval. Because of individual differences, the diversity and complexity of actions, and complex backgrounds, action recognition is a challenging problem. We know the challenges in action recognition mainly are: (1) person localization in cluttered or dynamic environment; (2) lighting condition; (3) observed from different viewpoint or with the dynamic background or with the moving camera. At the same time, a robust human action recognition algorithm should be invariant to different rates of execution [14].

From the viewpoint of application, action recognition can be divided into three types: (1) intelligent surveillance; (2) human machine interface and (3) automatic video annotation. There are two stages in human action recognition by computer vision: Action representation and classification of these representations. Action representations encode the human actions in the first stage, and have great affect on the second stage. An ideal representation should not only take into account the influence of the size of human body, the complex background, different viewpoint and the speed of the action, but also comprise sufficient information for the classifier to differentiate the actions. An effective classifier is expected to be able to distinguish existing and new action types.

Action recognition can be categorized in three general approaches: the approaches based on local features, based on global features and the systematic approach. The local feature based approach represents actions using the local spatial-temporal information [13][9][6]. For instance, [4]used an unsupervised learning method similar to bag-of-words approach to learn the probability distribution of the spatial-temporal interested points. The methods proposed in [12] obtained the activities model using the data captured from the joint movement to recognize the actions. These two methods failed to combine the local features as a whole in recognition. The global methods use global features, for example the optical flow, to represent motion. In [2], Histogram of Optical Flow (HOF) was used to recognize the movement of the athletes. In [11][3], 3D spatial-temporal models were used to represent human actions. Yet these methods did not build a model for the dynamic time features. In the systematic approach, a dynamic system is built to recognize human actions according to the feature changes of the human movement. So the dynamic features are taken into account. In [1] the trajectories of the joint

MM'13, October 21–25, 2013, Barcelona, Spain.

movement and the counters of the shape were used to represent the features. In [15] a nonlinear model was built according to the trajectories. However these systems suffer from the shortcoming of inadequate local representations.

In this paper, we propose a fast dense trajectories algorithm to recognize human actions. The algorithm of dense trajectories performs well in human action recognition, yet it makes heavy calculation load and not suitable for real-time systems. We extract foreground first and track the feature points in the foreground area to reduce the calculation significantly. Another contribution of this paper is using the temporal pyramid when tracking the trajectory, which makes our approach adaptable to the action speed.

This paper is organized as follows. In section 2, we describe the fast dense trajectories and the use of spatial-temporal points to distinguish recursive actions. In section 3, we employ the temporal pyramid to make the algorithm adaptable to action speed. In section 4 we test our approach on the dataset of ACM Multimedia 2013, and compared it with the dense trajectories algorithm. At last in section 5 is the conclusion of the research.

2. FAST DENSE TRAJECTORIES

Dense trajectories contain rich information of movement [8], but the dense optical flow introduces heavy calculation load, and is not suitable for the real-time application. In this paper, we employ a fast method for dense trajectories which can significantly reduce the calculation, and thus makes dense trajectories applicable in real-time.

2.1 Fast Dense Trajectories

Dense trajectories are the sequences of the dense feature points in the video. Let the optical flow for the Image I_t be $\omega_t = (u_t, v_t)$, feature point in I_t be $p_t = (x_t, y_t)$,then

$$A = \sum \omega(x - x_t, y - y_t) \begin{pmatrix} I_x^2 & I_x I_y \\ I_y I_x & I_y^2 \end{pmatrix} \quad (1)$$

where I_x, I_y is the gradient on the x and y directions respectively. The Eigen value for matrix A is λ_1, λ_2 , then the qualified feature point p_t should satisfy

$$\max(\lambda_1, \lambda_2) > T_{eig} \quad (2)$$

T_{eig} is the threshold. Its speed should satisfy

$$\left\| \omega_t(x_t, y_t) \right\|_2 > T_v \quad (3)$$

T_v is the threshold for the moving speed. If the feature point p_t satisfies conditions in (2) and (3), it will be selected as the seed and be tracked by optical flow, which means the small movements, which are usually caused by noise, will be abandoned. We track the seeds using the method as proposed in [8] on median filtering.

$$P_{t+1} = (x_{t+1}, y_{t+1}) = (x_t, y_t) + (M * \omega)|_{(\overline{x_t}, \overline{y_t})} \quad (4)$$

where M is the median filtering kernel, and $(\overline{x_t}, \overline{y_t})$ is the rounded position of (x_t, y_t) . Because of the calculation errors, long term tracking will fail and drift away the real trajectories. Thus, we limit the trajectory length to L frames as [8]. That is, when the length of the trajectory exceeds L frames, we stop tracking that trajectory, in order to avoid tracking with the accumulated error.

Figure.1 illustrates the dense trajectories of the action knocking. The circles indicate the current positions of the feature points, and the connecting lines are the trajectories. We can see that the trajectories are continuous, by which we can calculate the gradient and predict the next position so as to reduce the calculation and improve the property of real-time.

Figure 1 Illustration of the dense trajectories

We notice that in the dataset provided by the Huawei/3DLife – 3D human reconstruction and action recognition Grand Challenge, only the corresponding limbs that perform the actions make movement while other parts of the body keep still. Based on this observation, in the first step, we extract the foreground in the video sequences. Then the trajectories are tracked only in the foreground. This approach allows reducing the calculation load significantly, and moreover, it helps improving the recognition performance by removing the noise in the videos. Here foreground extraction is achieved using a threshold-difference background subtraction method.

2.2 SPATIAL-TEMPORAL POINTS

In the video series used in our experiments, we observe that the same action is often repeated for several times, but with different speed. As we can see in the knocking video, the knocking action is performed first slowly and the speed of action increases. As a result, the same action has different trajectories length. It is necessary to distinguish real action from a mixture of several recursive trajectories as one action. So we employ the spatial-temporal points to ensure each repeating action is clearly separated from others.

Spatial-temporal interest points are the points with the local space-time features that correspond to interesting events in video data whose neighborhood are with high spatial-temporal variation or the “space-time corners”. Usually we can find spatial-temporal points at the end of an action sequence. For example, when the football player heading the ball, the ball’s trajectory will change, we can get the spatial-temporal point, as show in figure 2 [9].

Similarly, we can get spatial-temporal points at the connection spot between two recursive actions in the sequences. Here we use the approach proposed in [13]. First the frames are filtered by the Gaussian filter in spatial, and then by the 1D Gabor filter in temporal. The intensity value of each point can be calculated as:

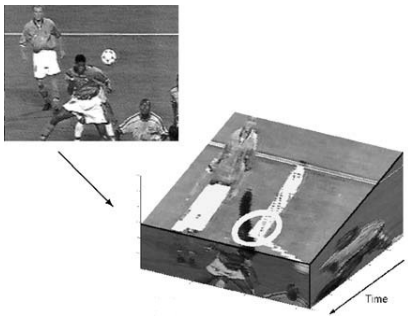

Figure 2 Spatial-temporal points

$$R = (I * g * h_{ev})^2 + (I * g * h_{od})^2 \quad (5)$$

where $g(x, y; \sigma)$ is two-dimensional spatial Gaussian smoothing kernel, h_{ev} and h_{od} are two orthogonal one-dimensional Gabor filters. Here $g(x, y; \sigma^2) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)2\sigma^2}$, $h_{ev}(t; \tau, \omega) = -\cos(2\pi t\omega) e^{-t^2/\tau^2}$, $h_{od}(t; \tau, \omega) = -\sin(2\pi t\omega) e^{-t^2/\tau^2}$.

2.3 DISTINGUISH BETEEN SIMILAR ACTIONS

Some actions are similar, for example, head scratching, eye rubbing and touching one's face. They all have the action putting the hands on the head, which makes the trajectories look alike. The biggest difference is what the hands touch with on the spatial-temporal points. So we recognize the hand and its connecting body parts using the haar cascade classifier. To some extend it helps improve the recognition accuracy.

3. TEMPORAL PYRAMID

The same action performed by different executors may take unequal time. For example, the kicking action requires the person first lift one of his leg, and then stretches it out, at last back it and stand straight. Such a series of actions may take different time. In order to ensure the proposed method is adaptable to the action speed, temporal pyramid is employed. In the pyramid, the top level $j = 0$ is a histogram over the full temporal extent of trajectories of the actions acquired as explained in section 2 and 3. The next level is the concatenation of two histograms obtained by temporally segmenting the action clip into two halves, and so on. We can get a coarse-to-fine representation by concatenating all such histograms together

$$x^j = \frac{2^{j-1}}{|T|} \sum_{t \in T^j} f^t \quad (6)$$

where T^j is the temporal extent on the j'th of the pyramid and x^j is the feature on that segment. The scale factors define an implicit correspondence based on the finest temporal resolution at which a model feature matches a video feature. This allows us to extract the trajectories in an adaptable manner. The same action with different speed can be correctly recognized.

We train a classifier with the public SVM implementation of [5]

$$x = \min(\left[x^0 \ldots x^j \ldots x^L\right]^T, 0.02) \quad (7)$$

Several histogram kernels have been considered in our experiments, but we found a simple linear kernel defined on an L1-normalized feature works well.

4. EXPERIMENTS

We tested our approach on the dataset of Huawei/3DLife – 3D human reconstruction and action recognition Grand Challenge in ACM Multimedia 2013. There are 26 kinds of gestures/movements in the dataset, among which 5 are static gestures, and the others are more dynamic. Each gesture/movement is performed by 7 individuals being recorded from six different viewpoints. We tested our algorithm on the 21 dynamic action types, and compared with the dense trajectories, as shown in Table1. Table 1shows that the clapping, pushing and walking actions have the highest recognition accuracy, because the trajectories are easy to distinguish from others. While eye movement and facial expression has the worst score, for they are somewhat static because only the eyeballs and the muscles on the face are moving, which are hard to track. And the two approaches perform basically equal for these actions. Our approach outperforms dense trajectories significantly for actions of the head scratching, eye rubbing and touching one's face.

In both our approach and the dense trajectories, the neighborhood size of the feature point has great influence on the recognition efficiency. In figure 3 the top row is the dense trajectories with different neighborhood size, and the bottom row is for our approach. The number indicates the size. We can see that when the size is small (D5 compared with F5), there are many useless feature points for fast dense trajectories. And when the size gets bigger, the feature points become less (D20 and F20). The calculation time gets smaller, as shown in figure 4. For fast dense trajectories, the size s=10 is the optimal option, and s=17 is the best for dense trajectories.

D5 D10 D15 D20

F5 F10 F15 F20

Figure 3 Influence of the neighborhood size of feature points

5. CONCLUSION

This paper has presented a fast dense trajectories algorithm for human action recognition. Dense trajectories perform well in action recognition. However it is time consuming. We extract the foreground first, and then track the feature points in the ROI instead of in the entire frame, which can improve the recognition efficiency greatly. Then we employ the temporal pyramid to make the algorithm adaptable to different action speed. For the similar actions, we use the interacting objects to distinguish. Our

approach outperforms the dense trajectories in the efficiency, and more robust to different action speed.

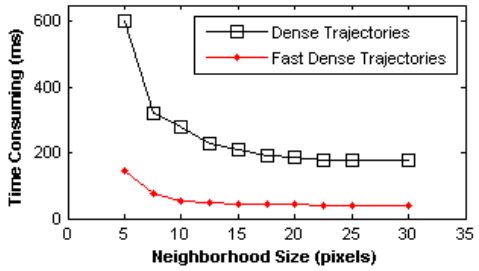

Figure 4 the neighborhood size feature

Table 1 Recognition performance

	Recognition accuracy (%)			Recognition accuracy (%)	
actions	Our approach	Dense trajectory	actions	Our approach	Dense trajectory
clapping	92.3	94.1	knocking the door	82.8	83.3
crossing the arms	87.7	88	lifting an object	88.6	88.8
crossing the legs	90.6	91.3	nodding	86.7	90.1
eye rubbing	88.4	83.7	punching/boxing	86.3	88.2
eye movement and facial expression	73.1	70.3	pushing away with both hands	91.3	93.6
fingers motion	83.5	83.2	bending the knees and standing up	89	91.1
fist clenched	77	81.1	throwing	87.6	89.3
head scratching	87.1	82.5	Touching one's face	91.1	87.9
head shaking	89.1	91.2	Walking on a treadmill	90.2	93.4
head tilting	88.7	89.1	waving one hand	84.2	84.5
kicking and punching	87.2	90.1			

6. REFERENCES

[1] Bissacco, A., Chiuso, A., and Soatto, S., 2007. Classification and recognition of dynamical models: The role of phase, independent components, kernels and optimal transport. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 29*, 11, 1958-1972.

[2] Efros, A.A., Berg, A.C., Mori, G., and Malik, J., 2003. Recognizing action at a distance. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on* IEEE, 726-733.

[3] Yilmaz, A. and Shah, M., 2005. Actions sketch: A novel action representation. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on* IEEE, 984-989.

[4] Niebles, J.C., Wang, H., and Fei-Fei, L., 2008. Unsupervised learning of human action categories using spatial-temporal words. *International Journal of Computer Vision 79*, 3, 299-318.

[5] Fan, R.-E., Chang, K.-W., Hsieh, C.-J., Wang, X.-R., and Lin, C.-J., 2008. LIBLINEAR: A library for large linear classification. *The Journal of Machine Learning Research 9*, 1871-1874.

[6] Willems, G., Tuytelaars, T., and Van Gool, L., 2008. An efficient dense and scale-invariant spatio-temporal interest point detector. In *Computer Vision–ECCV 2008* Springer, 650-663.

[7] Pirsiavash, H. and Ramanan, D., 2012. Detecting activities of daily living in first-person camera views. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on* IEEE, 2847-2854.

[8] Wang, H., Klaser, A., Schmid, C., and Liu, C.-L., 2011. Action recognition by dense trajectories. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on* IEEE, 3169-3176.

[9] Laptev, I., 2005. On space-time interest points. *International Journal of Computer Vision 64*, 2-3, 107-123.

[10] Liu, J., Kuipers, B., and Savarese, S., 2011. Recognizing human actions by attributes. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on* IEEE, 3337-3344.

[11] Blank, M., Gorelick, L., Shechtman, E., Irani, M., and Basri, R., 2005. Actions as space-time shapes. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on* IEEE, 1395-1402.

[12] İkizler, N. and Forsyth, D.A., 2008. Searching for complex human activities with no visual examples. *International Journal of Computer Vision 80*, 3, 337-357.

[13] Dollár, P., Rabaud, V., Cottrell, G., and Belongie, S., 2005. Behavior recognition via sparse spatio-temporal features. In *Visual Surveillance and Performance Evaluation of Tracking and Surveillance, 2005. 2nd Joint IEEE International Workshop on* IEEE, 65-72.

[14] Poppe, R., 2010. A survey on vision-based human action recognition. *Image and vision computing 28*, 6, 976-990.

[15] Ali, S., Basharat, A., and Shah, M., 2007. Chaotic invariants for human action recognition. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on* IEEE, 1-8.

[16] Hu, W., Tan, T., Wang, L., and Maybank, S., 2004. A survey on visual surveillance of object motion and behaviors. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on 34*, 3, 334-352