ACTIVITY RECOGNITION USING DENSE LONG-DURATION TRAJECTORIES

Ju Sun[1,2]*, Yadong Mu*[2]*, Shuicheng Yan*[2]*, Loong-Fah Cheong*[1,2]

[1]Interactive & Digital Media Institute, National University of Singapore, Singapore
[2]Department of Electrical and Computer Engineering, National University of Singapore, Singapore
Email: {idmsj, elemy, eleyans, eleclf}@nus.edu.sg

ABSTRACT

Current research on visual action/activity analysis has mostly exploited appearance-based static feature descriptions, plus statistics of short-range motion fields. The deliberate ignorance of dense, long-duration motion trajectories as features is largely due to the lack of mature mechanism for efficient extraction and quantitative representation of visual trajectories. In this paper, we propose a novel scheme for extraction and representation of dense, long-duration trajectories from video sequences, and demonstrate its ability to handle video sequences containing occlusions, camera motions, and nonrigid deformations. Moreover, we test the scheme on the KTH action recognition dataset [1], and show its promise as a scheme for general purpose long-duration motion description in realistic video sequences.

Keywords— motion trajectories, tracking, action recognition, video analysis, motion understanding, computer vision

1. INTRODUCTION

Research in machine perception of human activities has started in the computer vision community since the 1970's, stimulated by the classic moving light display (MLD) experiment of Johansson [2]. Various techniques have been proposed since then to tackle the problem (see, *e.g.*, the recent review [3]), and the major division is between the indirect school advocating reconstruction-based recognition and the direct school taking root in direct modeling and recognition [4]. Most current research work falls into the second school, as evidenced by [3].

Albeit the philosophical division between the two schools, there has been general consensus that human activities are differential in nature and hierarchy resides with the elusive term "activities". The seminal work of Bobick [5], for example, classifies human activities into "motions", "activities"[1], and "actions", in terms of increasing complexity in performing the activity, taking inspiration from Nagel's taxonomy [6].

Support of IDMPO Grant R-705-000-018-279 Singapore and CSIDM Project No. CSIDM-200803 partially funded by a grant from the National Research Foundation (NRF) (administrated by the Media Development Authority of Singapore) are gratefully acknowledged.

[1]This is of course not equivalent to the elusive definition of "activities" in general descriptions.

More critically, Bobick [5] has identified various levels of descriptive information needed, for successful representation and hence recognition of activities in the three-level categorization as described.

It is worth noting that besides "motions" for which description of motion flow fields may be sufficient, both "activities" and "actions" require more complex descriptions about the moving scenes. Specifically, they require the sequential motion information in the temporal direction (motion trajectories), and even the interactions between the spatiotemporal motion patterns (interactions between motion trajectories). The current research trend, however, shows slight regression with the prevalence of flow-based and volumetric motion features in most algorithms and the general ignorance of motion trajectories to capture long-duration motion characteristics (as evidenced by the very brief description of motion trajectories and the quick reference to the dated survey back to 1995 in [3]).

The ignorance is not incidental. In fact, the various difficulties associated with reliable visual tracking and informative trajectory representation have presented as obstacles. Visual tracking, *e.g.* keypoint-based (*e.g.* KLT [7]) and part-based (*e.g.* Meanshift tracking [8]), has individually constituted hot research topics in computer vision [9]. The variety of problems that need to be addressed by general tracking algorithms include visual noises, camera motions, occlusions. Furthermore, to use motion trajectories as features, there is still little agreement about the suitable ways for representation. The most obvious problems lie at the varying lengths of trajectories and that trajectories as real-valued 3D curves. Further difficulties could come out with the consideration for scale and viewpoint invariance. These have been the primary drives for our ongoing investigation into dense long-duration trajectory extractions and representations.

1.1. Our Investigation

Our focus in this paper is to establish an efficient scheme for trajectory extraction, to be combined with effective trajectory representations. In other words, we aim to revitalize long-duration trajectory descriptions in current research of activity analysis from video sequences, where long-duration temporal visual information has not received enough research spotlight it deserves.

Specifically, we propose to combine the initial tracking re-

978-1-4244-7493-6/10/$26.00 ©2010 IEEE

sults of the KLT [7] and the SIFT-trajectory [10], and formulate the visual matching and tracking in a unified constrained optimization problem. Through the optimization, the interior points that are neither corner points tracked by KLT nor keypoints tracked by SIFT-trajectory generation are accorded the respective flows by interpolation of the surrounding flows, subject to block-matching constraints as in normal optic flow computations [11]. The trajectory extraction mechanism can explicitly handle nonrigid deformations and occlusions, and partially eliminate the effect of camera motions with simple add-ons. A simple trajectory representation scheme is then presented to produce uniform feature vectors.

1.2. Related Work

The comprehensive review [3] has mostly covered the state-of-the-art features extraction and classification schemes used for human activity analysis. This part will link our current work to the most relevant.

The work on particle video (PV) [12] aims to combine the optic-flow and keypoint tracking techniques in order to produce dense long-duration trajectories of visual particles. The current takes inspiration from PV, and differs significantly in that: 1) we do not build our work on high-accurate optic-flow computations but resort to sparse tracking of image structural elements (corners and blobs currently); and 2) the fill-in points are based on importance sampling [13] which provides good randomness.

Scarce work has been done on trajectory representation. *Dynamic Instants* have been introduced in [14] that essentially capture the points of maximum curvature along the motion trajectory and aim at viewpoint invariance. *Star Diagrams* presented in [15] consist in overlaid trajectory segments that is closely connected to the linear dynamic motion models they have assumed. Our prior work [10] has pointed out the deficiency with the simple collection of trajectory segments, and proposed to treat the motion trajectory as a Markov dynamic process and hence employ the corresponding stationary distribution vector as a compact representation. To strike a balance towards efficiency and simplicity, we adopt a simple statistical representation in the temporal direction, with proper quantization according to magnitude and orientation as [10].

2. OVERVIEW OF DENSE LONG-DURATION TRAJECTORY EXTRACTION

We will overview technical details for various modules of the proposed dense long-duration trajectory extraction scheme. The organization is also consistent with the pipeline of our system.

2.1. KLT and SIFT Feature Tracking

Kanade-Lucas-Tomasi (KLT) tracking [7] has been a notable visual tracker that originates from early proposal for image registration. Later developments include the investigation into reliable features to track (corner points), affine verifications, and optimization speedups. We have adopted the implementation coming with [16] and initialized the tracking with Shi & Tomasi corner points [7].

By comparison, SIFT-tracking builds on matching of SIFT descriptors [17] between consecutive frames. It was designed to be a lightweight visual tracker and found its application in action recognition [10]. Uniqueness constraints are imposed on matching to ensure the validity of trajectories. Moreover, affine consistency check can also be incorporated to enhance the matching. We adopt the original proposal of Different-of-Gaussian (DOG) [17] as the feature detector, which essentially capture both corner and blob image structures. In practice it is interesting that the above two kinds of trackers serve as complementary sources for trajectory generations, as shown in Fig. 1.

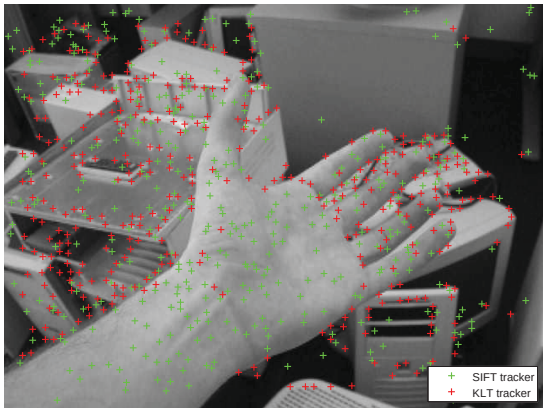

Fig. 1. Illustration of KLT and SIFT trackers on an indoor video sequence. Please refer to the electronic version for better viewing.

2.2. Random Point Generation and Triangulation

The KLT and SIFT trackers prove stable in practice, yet not dense enough to capture all salient image structures. Hence, the next step consists in generating random samples (similar to "particles" in PV) from gaps among existing trackers. Although a uniform sampling scheme is possible, it fails to account for the variation of different image regions. An intelligent way is to adapt the density of the samples with the distribution of local importance, *e.g.* in terms of the visual saliency, or the gradient map. This constitutes the notable "importance sampling" problem investigated in several research communities, *e.g.*, computer graphics. Specifically, we consider simply the magnitude of image gradients and the local variance of RGB intensity to construct the importance map, and adopt Ostromoukhov's hierarchical sampling method [13] that builds on the efficient *Penrose tiling*. Points generated using Ostromoukhov's method possess the blue noise property in the spectral domain, thus avoiding irregular point patterns and attaining better randomness. Examples on the "tree" and "car" video sequences are presented in Fig. 2.

Every frame now contains a point set that consists of all the cross-sections of trajectories through the current frame. Denote the KLT+SIFT point set as $\mathbf{P}_s$ and random tracker set as $\mathbf{P}_t$. To propagate information and mitigate outliers' bad effect, all

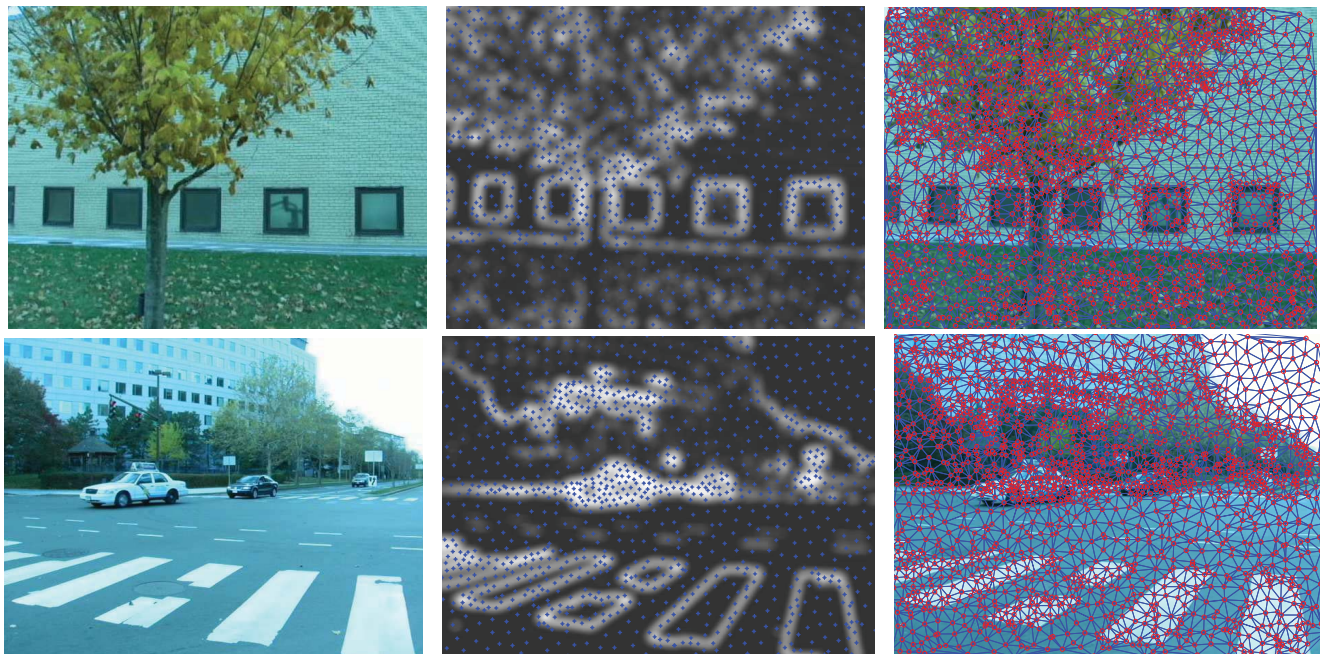

Fig. 2. Illustration of triangulation and importance sampling. **Left**: Original video frame. **Middle**: Importance map and random trackers. **Right**: Delaunay triangulation. Please refer to the electronic version for better viewing.

trackers are then linked to their locality via Delaunay triangulation to get $\mathrm{DT}(\mathbf{P}_s \cup \mathbf{P}_t)$, as shown in Figure 2.

2.3. Trajectory Optimization

There are two heterogenous sources of particles after the importance sampling as discussed above, i.e., KLT+SIFT trackers and random trackers. In the stage of motion estimation, these two kinds of trackers are treated differently. For KLT or SIFT points in $\mathbf{P}_s$, the initial optic flows can be directly obtained from either KLT or SIFT tracking, collectively denoted as $\widetilde{\mathbf{V}}_s$[2]. While for particles in $\mathbf{P}_t$, we estimate their motion using the classic Lucas-Kanade (LK) method [11], which is an advisable choice since LK is local method with closed-form solution and avoid oversmoothness. We denote these collectively as $\widetilde{\mathbf{V}}_t$ and let $\widetilde{\mathbf{V}} = \left[\widetilde{\mathbf{V}}_s, \widetilde{\mathbf{V}}_t\right]^\top$.

The triangulation network in general provides us with the structural information about the frame. For most object motions, we expect the optic flows for connected points to be similar. There are exceptions to this, at connections that cross strong edges. In such cases, the connections are likely to be created due to the topological constraint of the triangulation process, indicating spurious relations. Hence we propose to minimize the weighted difference of optic flows for connected particles. Specifically to

$$\min. \sum_{(\mathbf{v}_i, \mathbf{v}_j) \in \mathcal{C}} \omega_{ij} \|\mathbf{v}_i - \mathbf{v}_j\|^2, \tag{1}$$

where $\mathcal{C}$ collects all the connections in the triangulation network, and ω_{ij} is the specific penalty weight for the edge for the pair of flows $(\mathbf{v}_i, \mathbf{v}_j)$. We simply set ω_{ij} based on maximal gradient magnitude along the line connecting points $\mathbf{v}_i$ and $\mathbf{v}_j$. Stronger response tends to produce smaller ω_{ij}. On the other hand, we do not expect the refined optic flow to drastically deviate from the initial guess $\widetilde{\mathbf{V}}$, thus introducing a set of box constraints as below (all η_i are set to 0.4 in our implementation):

$$|\mathbf{v}_i - \widetilde{\mathbf{v}}_i| \leq \eta_i \|\widetilde{\mathbf{v}}_i\|_2, \ \forall i. \tag{2}$$

The problem in Eq. (1) with constraint (2) falls into standard quadratic programming (QP) with box constraints, and can be efficiently solved by off-the-shelf convex QP solvers. Upon performing optimization over all frames, motion trajectories of varying lengths are generated, from which various trajectory features are extracted.

2.4. Tracker Pruning and Additions

Our aim is to pursue long-range trajectories. Unfortunately, part of trajectories need be terminated due to occlusion, noises, or over-nearness to others. Moreover, new image regions emerge along with camera and objection motions, thus new tracker need be seeded. First we list four rules used to prune redundant trackers in current frame:

Rule 1. ***(for KLT trackers)*** *KLT trackers that fail to find matching points under affine transform in subsequent frame will be removed.*

[2]Here and henceforth, the capital $\mathbf{V}$ does not denote matrix; rather, it means we organize the optic flows in such a way that x-components are collected together, followed by the collection of the y-components in corresponding order. By comparison, $\mathbf{v}$ means the 2-dimensional optic flow vector for a particular particle point.

Rule 2. ***(for SIFT trackers)*** *Each SIFT tracker is associated with a counter (initialized to be 0). When a tracker finds no match in the next frame, the counter will be increased by 1 and the search is continued. Any SIFT trackers with counter larger than a threshold (3 in our implementation) will be abandoned.*

Rule 3. ***(for random trackers)*** *After motion estimation, each random tracker will get a matching score by comparing the neighboring patches around it and its "matching" point in the next frame.*

Rule 4. ***(for all trackers)*** *The trackers beyond image borders will be discarded. Moreover, for too-close tracker pair (e.g., closer than 3 pixels), an element of the two will be selected to remove.*

Consider two consequent frames F_i and F_{i+1}, for tracker addition, we discuss it in the following two cases:

1. For KLT and SIFT trackers, we pre-define a minimal tracker number (200 in our experiments) for each. Once the real tracker numbers are below this threshold, new KLT (or SIFT) trackers will be randomly sampled from the new frames.
2. The addition of random trackers is based on availability map and importance map. It consists three steps: 1) calculate the *availability map* in F_i based on existing trackers in F_i (details are ignored for space limit). 2) sample candidate random trackers based on importance map in F_{i+1}. 3) keep only trackers lying in the feasible regions in the availability map. See Fig. 3 for an illustration.

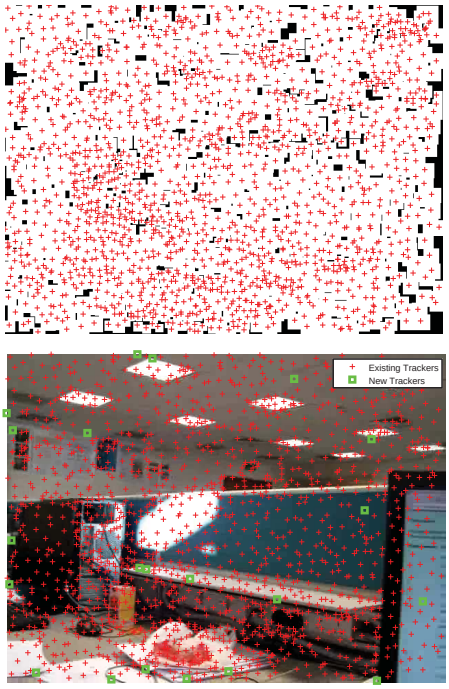

Fig. 3. Illustration of tracker addition based on importance map and availability map. **Top**: availability map. Image regions in black are allowed to insert new points. **Bottom**: newly-added random trackers (displayed in green squares). Please refer to the electronic version for better viewing.

2.5. Statistics of Flows as Feature Representation

Trajectory representation is an integral part in a motion trajectory-based recognition system. Ideal trajectory representation schemes should be compact and informative, general enough to represent trajectories of varying lengths, and simple enough to enable fast processing towards decision-making, and powerful enough to tolerate certain degrees of trajectory variations, such as scale changes, tracking noises, viewpoint alternations. Of course descriptions with high invariance are always being sought, "ambitious but not totally unrealistic" [6].

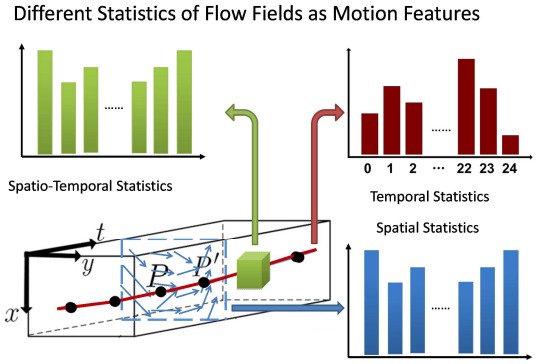

Fig. 4. Illustration of the spatial statistics, temporal statistics, and spatio-temporal statistics.

We adopt a simplified version of the representation scheme proposed in [10]. We quantize each valid motion trajectory (short trajectories are discarded) according to the magnitude (3 bins) and orientation (8 bins) following [10]. Instead of adopting the full representation of employing Markov stationary vector, we just form temporal and spatial statistics of flows based on the discrete states defined. In processing afterwards, the 24-dimensional feature vectors are treated as the building blocks.

Distinction should be noted about the difference between temporal and spatial statistics of flow. The information conveyed by spatial statistics is confined in the spatial domain, useful for identifying movements that are short-ranged and transient (some motions can be recognized by very short observations); by comparison, temporal statistics keep track of the temporal variations of the motions, information of long-duration, suggesting its usefulness for representing motions beyond the most simple and primitive patterns. By comparison, spatio-temporal statistics (HOF in STIP features [1]) of flows around interest points provide more compression, but less discrimination. See Fig. 4 for an illustration.

3. EVALUATION AND DISCUSSION

We evaluate the proposed trajectory extraction and representation scheme on the popular action recognition benchmark KTH dataset, and provide detailed analysis below.

3.1. Data Set Description and Evaluation Methodology

The KTH database [1] contains six types of human actions: walking, jogging, running, boxing, hand waving, and hand clapping. These actions are performed by 25 subjects in four different scenarios: outdoors, outdoors with scale variation, outdoors with different clothes, and indoors. These sequences

are taken in constrained environments with homogeneous background and static cameras. In each round, action sequences of one person are kept for testing while the rest are for training. The reported results are averaged over all valid training/testing splitting combinations.

3.2. Experiment Setup and Classifier Design

Typically, a video clip in KTH database comprises roughly 400 frames, from which thousands of long-range trajectories can be extracted using the proposed method. Most of these trajectories originate from image background rather than object actions, unreliable to discriminate distinct actions, which brings the challenge to filter trajectories unrelated to action recognition. Designing such filters is not a trivial task under simultaneous object and camera motions. However, for KTH database, we adopt a simple frame-differencing strategy. For each frame, the *potential region* refers to the set of pixels whose inter-frame direct differences are beyond a pre-defined threshold (we use the grayscale intensity and set the threshold to be 15). Only trajectories that stay in the *potential regions* during at least 30% of its life cycle are regarded as valid, otherwise discarded. In our experiments, roughly 10% trajectories are finally selected.

Temporal and spatial statistics of flows are then made. After that, we ignore the spatio-temporal configuration and adopt the popular "bag-of-features" framework, which introduces information loss yet being more compact. Specifically, hierarchical K-means algorithm is used to constructed the feature codebook. For the parameter selection in K-means, we constrain the tree depth to be 3, and splitting factor to be 10, which results in exactly 1000 leaf nodes. Based on this, we obtain the histogram representations for both spatial and temporal features and denote them as f_s, f_t respectively. Fig. 5 visualizes the histograms for all video clips in KTH database, where the difference between different actions can be visually observed, especially between the first three actions ("boxing", "handclapping", "handwaving") and other actions ("jogging", "running", "walking").

We adopt a multi-class SVM to solve the final classification problem. To gauge the pairwise similarity between two features, a widely used measure is the *Histogram Intersection Kernel* (HIK) [18] which is defined as follows:

$$\kappa\left(f(x_i), f(x_j)\right) = \sum_{k=1}^{d} \min\left(f^k(x_i), f^k(x_j)\right), \qquad (3)$$

where $f^k(x_i)$ denotes the k-th entry of $f(x_i) \in R^d$.

3.3. Experimental Results

As stated above, temporal and spatial trajectory features encodes different information for actions. In this part we conducts three experiments, utilizing f_t, f_s and f_{st} respectively. Recall that although both the kernel values of f_t and f_s can be computed via Eq. 3, however, the combination f_{st} has no explicit representation. To overcome this difficulty, we further assume f_{st} is a linear concatenation of f_t and f_s in the *reproducing kernel Hilbert space* (RKHS). In other words, $\kappa\left(f_{st}(x_i), f_{st}(x_j)\right) = \alpha_s \kappa\left(f_t(x_i), f_t(x_j)\right) + \alpha_t \kappa\left(f_s(x_i), f_s(x_j)\right)$, where α_s, α_t are positive combining coefficients that can be optimally estimated via *multiple kernel learning* (MKL) [19]. Here we adopt a simple kernel smoothing trick, i.e., uniformly averaging Gram matrix of f_s, f_t after normalizing the matrix traces to unity.

Table 1. Performance of action recognition over all valid training/testing configurations. The first three methods are based on our proposed features. (accuracy±deviation)

METHOD	AVERAGE ACCURACY (%)
SPATIAL FEATURE	81.3 ± 9.51
TEMPORAL FEATURE	83.1 ± 7.22
SPATIAL&TEMPORAL FEATURE	86.8 ± 10.7
METHOD IN [1]	71.7 ± 16.7

On our quad-core and 8G bytes memory computer, trajectory extraction proves efficient. It takes roughly 0.2 seconds to process a single frame with 320×240 pixels. Table 1 and Fig. 6 present the averaged recognition accuracies and confusion matrices respectively. The quantitative study validates the complementarity between spatial- and temporal trajectory features. Compared with existing video features such as the work in [1], our proposed features provides comparable performance yet more compact and efficient.

We stress that it is less meaningful to compare our results with the state-of-the-art on KTH reported in [20] and latest work (e.g., [21]). They have designed much more delicate action recognition/detection system, by exploiting further tricks such as spatio-temporal gridding, feature selection, branch-and-bound action detection, etc. By comparison, our focus here, as emphasized from the start, is to promote long-range trajectories as useful motion features. Moreover, those refining tricks applied to other types of motion features readily apply here with slight adaptations.

4. CONCLUSION AND FUTURE PERSPECTIVES

In this paper we describe a novel trajectory-based motion feature used for action recognition. The philosophy behind the method is to refine motion estimation by propagating reliable information. We provide empirical study on the widely used KTH database, which validates its effectiveness. For future work, we argues that the importance of trajectory-based features has been underestimated and more detailed comparative study together with other commonly employed features will be provided in consequent work. Moreover, several issues mentioned here need further exploration, including global motion estimation, occlusion handling.

5. REFERENCES

[1] C. Schuldt, I. Laptev, and B. Caputo, "Recognizing human actions: a local SVM approach," in *ICPR*, 2004.

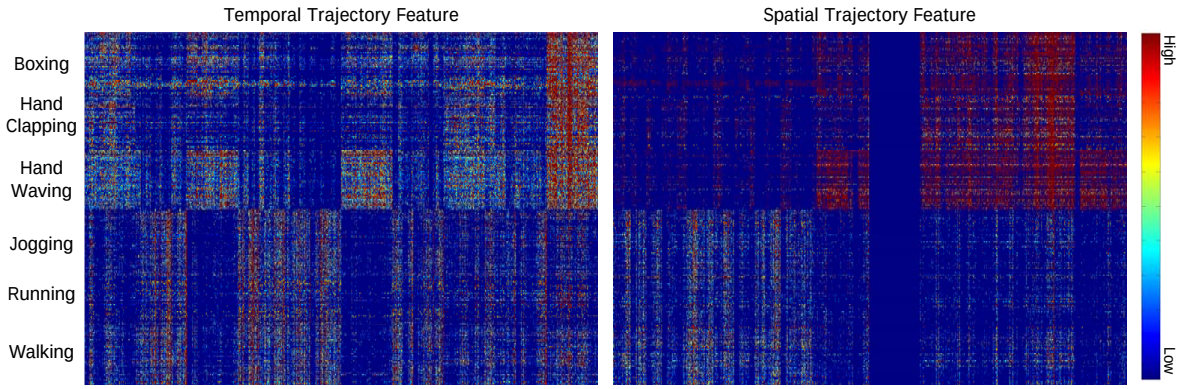

Fig. 5. Visualization of codewords for all KTH video clips. Each row corresponds to a video clip, and those related to the same actions are stacked adjacently. Please refer to the electronic version for better viewing.

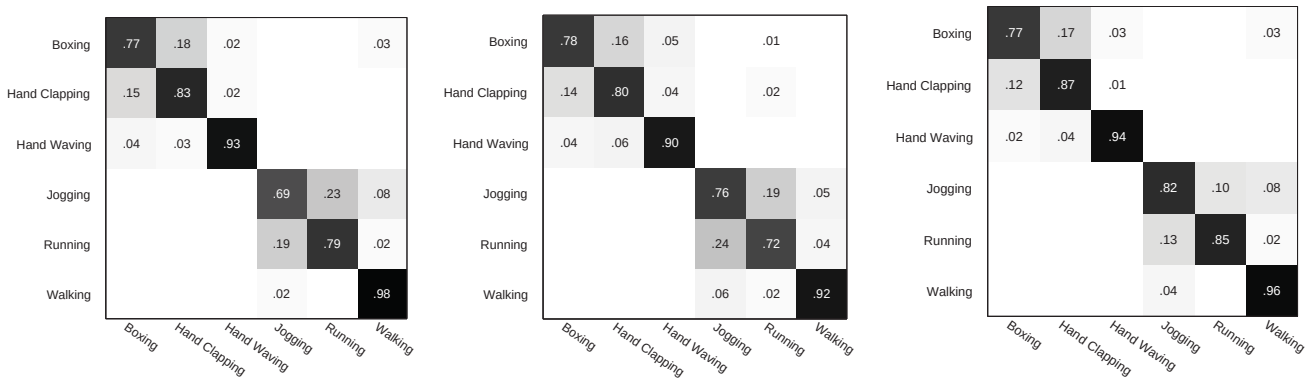

Fig. 6. Confusion matrices. **Left**: temporal trajectory feature. **Middle**: spatial trajectory feature. **Right**: temporal+saptial trajectory feature.

[2] G. Johansson, "Visual perception of biological motion and a model for its analysis," *Perception and Psychophysics*, vol. 14, pp. 201–211, 1973.

[3] P. Turaga, R. Chellappa, V.S. Subrahmanian, and O. Udrea, "Machine recognition of human activities: A survey," *IEEE Trans. CSVT*, vol. 18, no. 11, pp. 1473–1488, 2008.

[4] C. Cedras and M. Shah, "Motion-based recognition a survey," *Image. Vis. Comp.*, vol. 13, no. 2, pp. 129–155, 1995.

[5] A.F. Bobick, "Movement, activity and action: the role of knowledge in the perception of motion.," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 352, no. 1358, pp. 1257–1265, 1997.

[6] H.H. Nagel, "From image sequences towards conceptual descriptions.," *Image. Vis. Comp.*, vol. 6, no. 2, pp. 59–74, 1988.

[7] C. Tomasi and J. Shi, "Good features to track," in *CVPR*, 1994.

[8] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Trans. PAMI*, vol. 25, no. 5, pp. 564–577, 2003.

[9] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *ACM Computing Surveys (CSUR)*, vol. 38, no. 4, 2006.

[10] J. Sun, X. Wu, S. Yan, L.F. Cheong, T.S. Chua, and J. Li, "Hierarchical spatio-temporal context modeling for action recognition," in *CVPR*, 2009.

[11] B.D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *IJCAI*, 1981, vol. 3, pp. 121–130.

[12] P. Sand and S. Teller, "Particle video: Long-range motion estimation using point trajectories," *IJCV*, vol. 80, no. 1, pp. 72–91, 2008.

[13] V. Ostromoukhov, C. Donohue, and P.M. Jodoin, "Fast hierarchical importance sampling with blue noise properties," in *ACM SIGGRAPH*, 2004.

[14] C. Rao, A. Yilmaz, and M. Shah, "View-invariant representation and recognition of actions," *IJCV*, vol. 50, no. 2, pp. 203–226, 2002.

[15] N.P. Cuntoor and R. Chellappa, "Epitomic representation of human activities," in *CVPR*, 2007.

[16] S. Baker and I. Matthews, "Lucas-kanade 20 years on: A unifying framework," *IJCV*, vol. 56, no. 3, pp. 221–255, 2004.

[17] D.G. Lowe, "Distinctive image features from scale-invariant keypoints," *IJCV*, vol. 60, no. 2, pp. 91–110, 2004.

[18] S. Maji, A.C. Berg, and J. Malik, "Classification using intersection kernel support vector machines is efficient," in *CVPR*, 2008.

[19] Peter Gehler and Sebastian Nowozin, "On feature combination for multiclass object classification," in *ICCV*, 2009.

[20] I. Laptev, M. Marszałek, C. Schmid, and B. Rozenfeld, "Learning realistic human actions from movies," in *CVPR*, 2008.

[21] J. Yuan, Z. Liu, and Y. Wu, "Discriminative subvolume search for efficient action detection," in *CVPR*, 2009.