Unsupervised Discovery of Action Hierarchies in Large Collections of Activity Videos

Parvez Ahammad, Chuohao Yeo, Kannan Ramchandran and S. Shankar Sastry
Department of Electrical Engineering and Computer Sciences
University of California, Berkeley
Berkeley, CA 94720–1770
Email: {parvez,zuohao,kannanr,sastry}@eecs.berkeley.edu

***Abstract*— Given a large collection of videos containing activities, we investigate the problem of organizing it in an unsupervised fashion into a hierarchy based on the *similarity of actions* embedded in the videos. We use spatio-temporal volumes of filtered motion vectors to compute appearance-invariant action similarity measures efficiently - and use these similarity measures in hierarchical agglomerative clustering to organize videos into a hierarchy such that neighboring nodes contain similar actions. This naturally leads to a simple automatic scheme for selecting videos of representative actions (exemplars) from the database and for efficiently indexing the whole database. We compute a performance metric on the hierarchical structure to evaluate goodness of the estimated hierarchy, and show that this metric has potential for predicting the clustering performance of various joining criteria used in building hierarchies. Our results show that perceptually meaningful hierarchies can be constructed based on action similarities with minimal user supervision, while providing favorable clustering performance and retrieval performance.**

I. Introduction

Given the growing popularity of online video databases and the ease of recording and storing videos, access to video data will only increase. Even in special scenarios such as surveillance or environmental habitat monitoring where networks of cameras are deployed, it is typical to record large amounts of video data. We are interested in video data that contains actions or movements of human beings or objects. The notion of action similarity induces a perceptual hierarchy on the database of videos (see Figure 1 for example). A system that can efficiently generate such a hierarchy of the videos based on action similarity would be very useful in facilitating efficient navigation of the database thus improving its utility. Building such a system is very challenging if we consider videos containing actions of articulated structures like humans and animals moving in the visual scenes. It is preferable to assume *no metadata* (e.g. labels), *no segmentation* and *no prior alignment* for the video collections.

It is very useful to have a measure of clustering performance given a set of videos in an unsupervised setting. While this is straight-forward in the supervised scenario, it is not clear how the quality of the database organization can be judged in the absence of labels. We propose a solution to this problem by computing a performance measure on the estimated hierarchy.

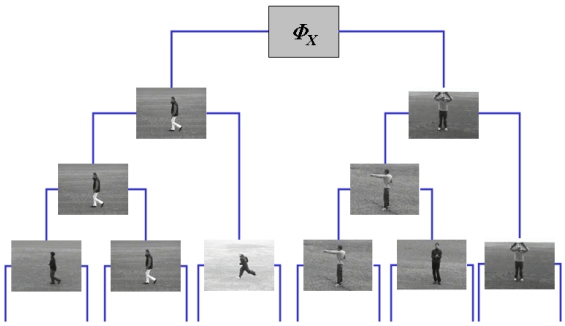

Fig. 1. A qualitative example of an action hierarchy for the activity video collection Φ_X, with associated exemplars for the subtree under each node, shown up to 6 clusters. This was generated using our proposed approach with NCNC as the action similarity measure and Ward linkage as the neighbor-joining criterion. The 6 clusters from left to right: Jogging, Walking, Running, Boxing, Handclapping, Handwaving. See Section III for further discussion.

A. Problem Statement

Given a set of videos and a user-defined *space-time scale* of actions, we would like the system to: *(a) automatically* and *efficiently* organize the videos into a hierarchy based on action similarity; *(b)* estimate clusters; *(c)* compute a performance measure on the estimated hierarchy (even without access to the labels); and *(d)* select one representative exemplar for each cluster.

B. Related Work

One of the key components in efficient grouping of actions is the ability to quickly localize and recognize actions. In our previous work [1], we used the motion vector information to compute frame-to-frame motion similarity between a query video and a target video with a similarity measure that takes into account differences in both orientation and magnitude of motion vectors. Shechtman et al.'s approach for estimating action similarity [2] is computationally complex compared to our method and may be unsuitable for use in organizing large video databases. Babu et al. [3] use codewords based on the histogram of motion vector components of the whole frame; this approach cannot localize actions very well in the video.

For large databases of videos, techniques that operate directly on compressed domain features offer a significant speed-up in processing time. Dimitrova et al. [4] assume that motion vectors are coarse approximations of optical flow but unlike our approach, they estimate object trajectories explicitly using

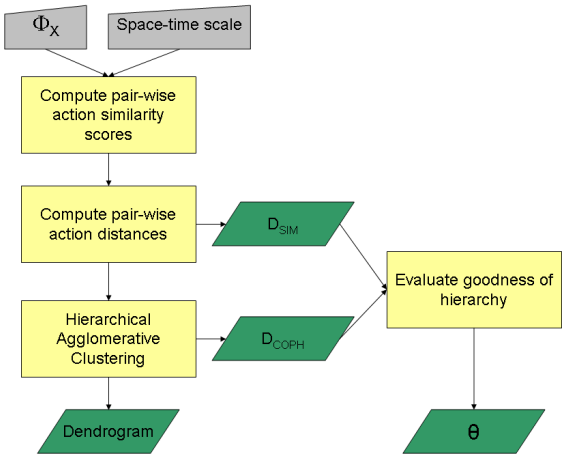

Fig. 2. Data flow for our proposed approach. Given a set of videos Φ_X and a user-defined space-time scale for actions, we compute pair-wise action similarity scores between all pairs of videos, and then convert them to symmetric action distances, D_{SIM}. We use D_{SIM} in hierarchical agglomerative clustering to produce a dendrogram, which is a binary hierarchical tree representing the videos, and the pair-wise cophenetic distances D_{COPH}, which are distances computed from the constructed dendrogram. The cophenetic correlation coefficient, Θ, is the correlation coefficient between D_{SIM} and D_{COPH}, and can be used to evaluate the goodness of the hierarchy.

motion vectors. Chang et al. assume that objects can be segmented and tracked easily in order to compute features [5]. Some approaches segment a single video into shots and organize neighboring shots into a hierarchy for browsing the video but they do not build action based hierarchies across a large collection of videos [6], [7].

II. Proposed Approach

Let $\Phi_X \doteq \{X^p\}_{p=1}^P$ be the given set of videos, where $P \in \mathbb{Z}_+$ is the cardinality of the set, and let $\tilde{N} \times \tilde{M} \times \tilde{T}$ be the user-specified space-time scale of interest. Each video X^p has an action label $y^p \in \{1, .., K\}$, where K is the number of actions in the collection. Assume that X^p is a video with T^p frames, with each frame containing $N^p \times M^p$ macroblocks. We assume that an *action* induces a motion field that can be observed as a spatio-temporal pattern; let $\vec{V}^p$ be the spatio-temporal pattern (motion field) associated with video X^p. Furthermore, $\vec{V}^p_{n,m}(i) = [V^{p,u}_{n,m}(i) \quad V^{p,v}_{n,m}(i)]$ denotes the motion vector at location (n, m) in frame i of X^p. We assume that similar actions will induce similar motion fields - i.e., $y^p = y^q \iff D(X^p, X^q) < \gamma$ for some acceptance threshold γ, where $D(.,.)$ is the distance metric defined between the videos based on their motion fields (defined in Section II-B). We will use $(\boldsymbol{u})_+$ as a shorthand for $\max(0, \boldsymbol{u})$.

Figure 2 shows the flow of our algorithm for organizing the videos (Φ_X) with minimal user input. For an extensive discussion on the intuition behind the steps involved in computation of action similarity, please refer to [1].

A. Computation of efficient pair-wise action similarity scores

In order to compute non-symmetric pair-wise action similarity scores between X^{test} and X^{query}, we carry out the following steps [1].

1) Obtain the motion field estimate, $\vec{V}$, for a video X from its compressed-domain motion vectors, keeping only the reliable estimates as indicated by a confidence map computed from DCT AC coefficients [8]. Motion vectors have been found to be a coarse but reasonable estimate of the motion field, and using them allows our approach to be computationally efficient.
2) At a particular macroblock location (n, m) of the test video, compute the frame-to-frame motion similarity measure, $\tilde{S}_{n,m}(i, j)$, between the i^{th} test video frame and the j^{th}th query video frame (cropped to $\tilde{N} \times \tilde{M}$ macroblocks). In our experiments, we used two methods to compute $\tilde{S}_{n,m}(i, j)$: *Normalized Correlation between Non-negative motion Channels (NCNC)*, and *Non-Zero Motion block Similarity (NZMS)* (discussion follows).
3) To enforce temporal consistency of the similarity between X^{test} and X^{query}, we convolve $\tilde{S}_{n,m}(i, j)$ with a smoothing kernel $H_\alpha \in \mathbb{R}^{T \times T}$. The resultant aggregated similarity matrix is $S_{n,m}(i, j) = (\tilde{S}_{n,m} \star H_\alpha)(i, j)$[1]. α is a parameter that allows us to control how tolerant we are to different action rates [1].
4) After repeating the above two steps over space and time we compute a confidence score at the (n, m) macroblock of test video frame i by taking the maximum of the aggregated similarity matrix over a space-time window:
$$C(n, m, i) = \max_{\substack{\max(i-\frac{T}{2},0) \le k \le \min(i+\frac{T}{2}, T^{\text{test}}-1) \\ 0 \le j \le \tilde{T}-1}} S_{n,m}(k, j) \tag{1}$$
5) Compute the similarity, $\rho(X^{\text{test}}, X^{\text{query}})$, of the test video to the query video by:
$$\rho(X^{\text{test}}, X^{\text{query}}) = \frac{\sum_{i=0}^{T^{\text{test}}-1} \eta(i) \left(\max_{n,m} C(n, m, i)\right)}{\sum_{i=0}^{T^{\text{test}}-1} \eta(i)} \tag{2}$$
where $\eta(i)$ is an indicator function which returns one if at least T frames in the $2T$-length temporal neighborhood centered at frame i have significant motion and returns zero otherwise. A frame is asserted to have significant motion if at least δ proportion of the macroblocks have reliable motion vectors of magnitude greater than ϵ.

In our experiments, we used $\alpha = 2.0$, $\tilde{N} = \tilde{M} = 6$, $T = 17$, $\tilde{T} = 2T + 1 = 35$, $\delta = \frac{1}{30}$ and $\epsilon = 0.5$ pixels/frame.

Let us elaborate on the methods for computing $\tilde{S}_{n,m}(i, j)$:

1) Normalized Correlation between Non-negative motion Channels (NCNC): Each $\vec{V}_{n,m}$ is first split into non-negative motion channels (e.g. left, right, up and down) [1], [9]. An $\tilde{N} \times \tilde{M}$ patch of these motion channels with top-left corner at (n, m) is stacked into a single vector $\vec{U}_{n,m} \in \mathbb{R}^{4\tilde{N}\tilde{M}}$. $\tilde{S}^{\text{NCNC}}_{n,m}(i, j)$ is then computed as follows:
$$\tilde{S}^{\text{NCNC}}_{n,m}(i, j) = \frac{\langle \vec{U}^{\text{test}}_{n,m}(i), \vec{U}^{\text{query}}(j) \rangle}{\|\vec{U}^{\text{test}}_{n,m}(i)\| \|\vec{U}^{\text{query}}(j)\|} \tag{3}$$

[1]Note that the convolution is performed separately for each (n, m), and is only over the (i, j) frame indices.

2) Non-Zero Motion block Similarity (NZMS): $\tilde{S}^{\text{NZMS}}_{n,m}(i,j)$ is computed as follows [1]:

$$\tilde{S}^{\text{NZMS}}_{n,m}(i,j) = \frac{1}{Z_{n,m}(i,j)} \sum_{k=0}^{\tilde{N}-1} \sum_{l=0}^{\tilde{M}-1} f(\vec{V}^{\text{test}}_{k+n,l+m}(i), \vec{V}^{\text{query}}_{k,l}(j)) \tag{4}$$

$$f(\vec{V_1}, \vec{V_2}) = \begin{cases} \frac{(\langle \vec{V_1}, \vec{V_2} \rangle)_+}{\max(\|\vec{V_2}\|^2, \|\vec{V_1}\|^2)} & \text{if } \|\vec{V_1}\| > 0 \text{ and } \|\vec{V_2}\| > 0 \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

The normalizing factor, $Z_{n,m}(i,j)$, in (4) is:

$$Z_{n,m}(i,j) = \sum_{k=0}^{\tilde{N}-1} \sum_{l=0}^{\tilde{M}-1} \mathbb{I}\Big[\|\vec{V}^{\text{test}}_{k+n,l+m}(i)\| > 0 \text{ or } \|\vec{V}^{\text{query}}_{k,l}(j)\| > 0\Big] \tag{6}$$

B. Computation of pair-wise action distances

Using the similarity scores computed from Section II-A, we compute the pair-wise symmetric action distances for videos X^p and X^q as follows:

$$D_{\text{SIM}}(X^p, X^q) = \frac{1}{\max\left(\frac{1}{2}\left(\rho(X^p, X^q) + \rho(X^q, X^p)\right), \beta\right)} \tag{7}$$

where β represents the smallest value of $\rho(.,.)$ admissible. In our experiments, we choose $\beta = 0.01$.

C. Hierarchical agglomerative clustering of actions

We apply hierarchical agglomerative clustering (HAC) [10] to construct a binary tree (also called *dendrogram*) containing all the elements of Φ_X as leaf nodes. Divisive methods (e.g. K-means, K-medoids) for constructing dendrogram are usually sensitive to initialization [10]. To address this sensitivity with divisive methods, typically one needs to perform many randomly initialized trials in order to obtain a good clustering solution, thus resulting in loss of computational efficiency. In contrast, HAC constructs the dendrogram in a sequential and *deterministic* fashion using a neighbor-joining (also called linkage) criterion. We use four different linkage criteria in our experiments: *Single linkage*, *Complete linkage*, *Average linkage* and *Ward linkage* [11].

The user defines a stopping condition for the agglomeration, L^{STOP}, which is the farthest allowable merging distance between clusters. L^{STOP} is used to cut the dendrogram at an appropriate level and obtain the clusters. After computing the matrix of pair-wise action distances $D_{\text{SIM}} \in \mathbb{R}^{P \times P}$ as described in Section II-B, we apply HAC to obtain the hierarchy. The *cophenetic distance* between videos X^p and X^q, $D_{\text{COPH}}(X^p, X^q)$, computed in the HAC procedure, is their linkage distance when first merged into the same cluster [10].

D. Measuring the goodness of the estimated hierarchy

Different choices in clustering parameters, such as distance metric or linkage criteria, lead to different hierarchies (dendrograms). For a good hierarchy, *the cophenetic distances*, D_{COPH}, *should obey the input pair-wise distance relationships* specified by D_{SIM} [12]. The *Cophenetic Correlation Coefficient*, $\Theta \in [0,1]$, for a dendrogram is defined as the correlation coefficient between D_{COPH} obtained from the dendrogram, and D_{SIM} used to construct the dendrogram [13]. Thus Θ is a measure of *how faithfully the dendrogram represents the dissimilarities* among videos in the given set Φ_X; its magnitude should be close to 1 for a high-quality solution. Θ is useful in comparing alternative dendrograms obtained by using different neighbor joining strategies.

III. Experimental Results and Discussion

We use a publicly available[2] comprehensive dataset compiled by [14] to perform our evaluations. This dataset consists of different actions (boxing, handclapping, handwaving, running, jogging and walking) performed by 25 different people over 4 different environments (outdoors [d1], outdoors with scale variations [d2], outdoors with different clothes [d3] and indoors [d4]). Since the two similarity measures we used are not designed for scale-varying actions, we considered only the three non-scale-varying environments.

From each action video, we create a query video by cropping out a space-time volume in an automatic fashion. Since automatic determination of space-time scale is very hard, we let the user specify the size of an approximate space-time bounding box, $\tilde{N} \times \tilde{M}$ macroblocks by $\tilde{T}$ frames, for the entire collection of videos[3]. The system then looks in each action video for a $\tilde{M} \times \tilde{N} \times \tilde{T}$ space-time volume that contains the most number of significant motion vectors, where $\vec{V}$ is significant if $\|\vec{V}\| > \epsilon$ (as defined in Section II-A).

We adopt two different criteria for evaluating the performance of our organization scheme. The first is based on the ability of the hierarchy to infer meaningful exemplars from the dataset and the second is based on the *F-score* [15] used in information-retrieval literature.

A. Inferring action exemplars

In each cluster, an exemplar is defined as the element that has the minimum pair-wise distance with respect to all the other elements in the cluster. *A meaningful hierarchy would organize the videos in such a way that exemplars from each cluster would represent a distinct action from the dataset*. In Figure 1, we show the estimated action hierarchy constructed using NCNC action similarity measure with Ward linkage neighbor-joining criterion. Notice that the actions such as running, walking and jogging were grouped separately compared to actions such as boxing, handwaving or handclapping. Intuitively, this fits well with what a human operator would do given the same task. Among the 4 linkage criteria we used, we found qualitatively that the combination of NCNC and Ward linkage gives the best inference for exemplars of actions in the database.

B. Evaluating clustering performance

Treating the action video X^p (with label y^p and in cluster $C^p \in \{1, \cdots, K\}$) as a query video, we define the following:

[2] http://www.nada.kth.se/cvap/actions/

[3] This implicitly constrains the system to consider actions of approximately similar space-time scale.

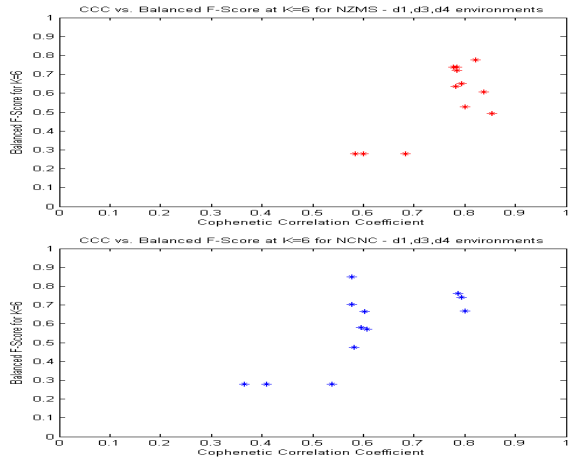

Fig. 3. Plots (top: NZMS, bottom: NCNC) showing positive correlation between Cophenetic Correlation Coefficient (Θ) and the Balanced F-score (F_1), suggesting that the goodness of hierarchy correlates well with clustering performance.

1) N_1^p is the number of videos in cluster C^p with label y^p,
2) N_2^p is the number of videos in cluster C^p,
3) N_3^p is the number of videos in Φ_X with label y^p.

For the query video X^p, we compute *precision* as $Pr^p = N_1^p/N_2^p$ and its *recall* as $Rc^p = N_1^p/N_3^p$. The *Balanced F-score* [15], F_1^p, for this query is the harmonic mean of its *precision* and *recall*: $F_1^p = \frac{2 \cdot Pr^p \cdot Rc^p}{Pr^p + Rc^p}$. We average F_1^p to get $F_1 = \frac{\sum_{p=1}^{P} F_1^p}{P}$. Since the labels in our dataset are for six actions, for the purpose of making comparisons, we only consider F_1 using a value of L^{STOP} such that the number of estimated clusters is 6. We also compute Θ as described in Section II-D. Figure 3 shows the variation of F_1 with Θ for different neighbor joining criteria and action environments. The correlation coefficient between F_1 and Θ is 0.77 for NZMS and 0.73 for NCNC respectively - suggesting that Θ can be used to predict clustering performance across various linkage criteria even in the absence of labels.

We also compare F_1 scores of our proposed approach with those of a baseline clustering scheme, K-medoids [10], with $K = 6$. We run K-medoids with 200 different random initializations and pick the best F_1 score over all the runs. Due to space constraints, we show only results for HAC using Ward linkage and K-medoids in Table I. It is clear from the results that HAC almost always gives favorable clustering results, without any initialization issues while efficiently producing a useful hierarchy.

IV. Conclusions and Future Directions

We have demonstrated an efficient unsupervised approach for organizing large collections of videos into a meaningful hierarchy based on the similarity of *actions* embedded in the videos. The database can be quickly indexed by assigning a unique action tag to each cluster[4] and these derived action tags can then be combined with other features (such as color, texture etc.) to build more complex queries or to develop organizational principles for managing video databases. Based on the evidence of high correlation between Θ and F_1 for a given D_{SIM}, we conjecture that *the unsupervised hierarchical solution for actions that has high Θ would also be a solution with high F_1*, thus hinting at good clustering performance. We plan to extend this framework to include features from raw video and investigate other clustering criteria in the future.

[4]User can easily label a cluster simply by identifying the cluster exemplar.

TABLE I

F_1 scores for different environments and clustering methods

Environment	NZMS / HAC Ward	NCNC / HAC Ward	NZMS / K-medoids	NCNC / K-medoids
d1	0.7384	0.8496	0.8089	0.7514
d3	0.7220	0.6659	0.7122	0.6480
d4	0.7774	0.7614	0.7515	0.6601

Acknowledgments

This work is supported in part by the National Science Foundation (NSF) Grant CCR-0330514 and Army Research Office (ARO) Award W911NF-06-1-0076. CY is funded by the Agency for Science, Technology and Research, Singapore (A*STAR). Any opinions, findings, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF, ARO or A*STAR.

References

[1] C. Yeo, P. Ahammad, K. Ramchandran, and S. S. Sastry, "Compressed domain real-time action recognition," in *Proceedings of IEEE International Workshop on Multimedia Signal Processing (MMSP)*, 2006.
[2] E. Shechtman and M. Irani, "Space-time behavior based correlation," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, San Diego, USA, June 2005, pp. 405–412.
[3] R. V. Babu, B. Anantharaman, K. Ramakrishnan, and S. Srinivasan, "Compressed domain action classification using HMM," *Pattern Recognition Letters*, vol. 23, no. 10, pp. 1203–1213, Aug. 2002.
[4] N. Dimitrova and F. Golshani, "Rx for semantic video database retrieval," in *MULTIMEDIA '94: Proceedings of the second ACM international conference on Multimedia*. New York, NY, USA: ACM Press, 1994, pp. 219–226.
[5] S.-F. Chang, W. Chen, H. Meng, H. Sundaram, and D. Zhong, "A fully automated content-based video search engine supporting spatiotemporal queries," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 8, no. 5, pp. 602 – 615, Sept. 1998.
[6] M. M. Yeung and B. Liu, "Efficient matching and clustering of video shots," in *IEEE International Conference on Image Processing*, vol. 1, 1995, pp. 338–341.
[7] C. Ngo, T. Pong, and H. Zhang, "On clustering and retrieval of video shots through temporal slices analysis," *IEEE Transactions on Multimedia*, vol. 4, no. 4, pp. 446–458, 2002.
[8] M. T. Coimbra and M. Davies, "Approximating optical flow within the mpeg-2 compressed domain." *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 1, pp. 103–107, 2005.
[9] A. Efros, A. Berg, G. Mori, and J. Malik, "Recognizing action at a distance," in *Proc. IEEE International Conference on Computer Vision*, Nice, France, Oct. 2003.
[10] A. Webb, *Statistical Pattern Recognition*. Oxford: Oxford University Press, 1999.
[11] J. Hair, R. Anderson, R. Tatham, and W. Black, *Multivariate Data Analysis*, 4th ed. New York, NY: Prentice Hall, 1995.
[12] J. F. Rohlf and D. R. Fisher, "Tests for hierarchical structure in random data sets," *Systematic Zoology*, vol. 17, no. 4, pp. 407–412, Dec 1968.
[13] J. S.Farris, "On the cophenetic correlation coefficient," *Systematic Zoology*, vol. 18, no. 3, pp. 279–285, Sep 1969.
[14] C. Schüldt, I. Laptev, and B. Caputo, "Recognizing human actions: A local SVM approach." in *Proc. International Conference on Pattern Recognition*, Cambridge, UK, Aug. 2004, pp. 32–36.
[15] B. Larsen and C. Aone, "Fast and effective text mining using linear-time document clustering," in *KDD '99: Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*. New York, NY, USA: ACM Press, 1999, pp. 16–22.