Searching Video for Complex Activities with Finite State Models

Nazlı İkizler
Department of Computer Engineering
Bilkent University
Ankara, 06800, Turkey
inazli@cs.bilkent.edu.tr

David Forsyth
Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana, IL, 61801
daf@cs.uiuc.edu

Abstract

We describe a method of representing human activities that allows a collection of motions to be queried without examples, using a simple and effective query language. Our approach is based on units of activity at segments of the body, that can be composed across space and across the body to produce complex queries. The presence of search units is inferred automatically by tracking the body, lifting the tracks to 3D and comparing to models trained using motion capture data. We show results for a large range of queries applied to a collection of complex motion and activity. Our models of short time scale limb behaviour are built using labelled motion capture set. We compare with discriminative methods applied to tracker data; our method offers significantly improved performance. We show experimental evidence that our method is robust to view direction and is unaffected by the changes of clothing.

1. Introduction

Understanding what people are doing is one of the great unsolved problems of computer vision. A fair solution opens tremendous application possibilities. The major difficulties have been that (a) good kinematic tracking is hard; (b) models typically have too many parameters to be learned directly from data; and (c) for much everyday behaviour, there isn't a taxonomy. Tracking is now a usable, if not perfect technology (section 2.1). Building extremely complex dynamical models from heterogenous data is now well understood by the speech community, and we borrow some speech tricks to build models from motion capture data (section 2.2).

Our particular interest is everyday activity. In this case, a fixed vocabulary either doesn't exist, or isn't appropriate. For example, one often does not know words for behaviours that appear familiar. One way to deal with this is to work with a notation (for example, laban notation); but such notations typically work in terms that are difficult to map to visual observables (for example, the weight of a motion). We must either develop a vocabulary or develop expressive tools for authoring models. We favour this third approach (section 3).

Timescale: Space does not allow a complete review; the literature is complex, because there are many quite different cases in activity recognition. Motions could be sustained (walking, running) or have a localizable character (catch, kick). The information available to represent what a person is doing depends on timescale. We distinguish between short-timescale representations (**acts**); medium timescale **actions**, like walking, running, jumping, standing, waving, whose temporal extent can be short (but may be long) and are typically composites of multiple acts; and long timescale **activities**, which are complex composites of actions.

1.1. Review

There is a long tradition of research on interpreting activities in the vision community (see, for example, the extensive surveys in [20, 14]). There are three major threads. First, one can use temporal logics to represent crucial order relations between states that constrain activities; these methods are currently unpopular, and we do not review them here. Second, one can use spatio-temporal templates to identify instances of activities. Third, one can use (typically, hidden Markov) models of dynamics.

Timescale: A wide range of helpful distinctions is available. Bobick [5] distinguishes between movements, activity and actions, corresponding to longer timescales and increasing complexity of representation; some variants are described in two useful review papers [1, 15].

Methods based on Templates: The notion that a motion produces a characteristic spatio-temporal pattern dates at least to Polana and Nelson [23]. Spatio-temporal patterns are used to recognize actions in [6]. Ben-Arie *et al.* [3] recognize actions by first finding and tracking body parts using a form of template matcher and voting on lifted tracks.

1-4244-1180-7/07/$25.00 ©2007 IEEE

Bobick and Wilson [7] use a state-based method that encodes gestures as a string of vector-quantized observation segments; this preserves order, but drops dynamical information. Efros *et al.* [11] use a motion descriptor based on optical flow of a spatio-temporal volume, but their evaluation is limited to matching videos only. Blank *et al.* [4] define actions as space-time volumes. An important disadvantage of methods that match video templates directly is that one needs to have a template of the desired action to perform a search.

HMM's: Hidden Markov models have been very widely adopted in activity recognition, but the models used have tended to be small (e.g, three and five state models in [10]). Such models have been used to recognize: tennis strokes [30]; pushes [28]; and handwriting gestures [31]. Feng and Perona [13] call actions "movelets", and build a vocabulary by vector quantizing a representation of image shape. These codewords are then strung together by an HMM, representing activities; there is one HMM per activity, and discrimination is by maximum likelihood. The method is not view invariant, depending on an image centered representation. There has been a great deal of interest in models obtained by modifying the HMM structure, to improve the expressive power of the model without complicating the processes of learning or inference. Methods include: coupled HMM's ([10]; to classify T'ai Chi moves); layered HMM's ([22]; to represent office activity); hierachies ([21]; to recognize everyday gesture); HMM's with a global free parameter ([29]; to model gestures); and entropic HMM's ([9]; for video puppetry). Building variant HMM's is a way to simplify learning the state transition process from data (if the state space is large, the number of parameters is a problem). But there is an alternative — one could author the state transition process in such a way that it has relatively few free parameters, despite a very large state space, and then learn those parameters; this is the lifeblood of the speech community.

Stochastic grammars have been applied to find hand gestures and location tracks as composites of primitives [8]. However, difficulties with tracking mean that there is currently no method that can exploit the potential view-invariance of lifted tracks, or can search for models of activity that compose across the body and across time.

Finite state methods have been used directly. Hongeng *et al.* demonstrate recognition of multiperson activities from video of people at coarse scales (few kinematic details are available); activities include conversing and blocking [17]. Zhao and Nevatia use a finite-state model of walking, running and standing, built from motion capture [32]. Hong *et al.* use finite state machines to model gesture [16].

2. Representing Acts and Activities

Since we want our complex, composite motions to share a vocabulary of base units, we use the kinematic configuration of the body as distinctive feature. We have ignored limb velocities and accelerations because actions like reach/wave can be performed at varying speeds. However, velocity and acceleration is a useful clue when differentiating run and walk motions.

We want our representation to be as robust as possible to view effects and to details of appearance of the body. Furthermore, we wish to search for motions without possessing an example. All this suggests working with an inferred representation of the body's configuration (rather than, say, image flow templates). An advantage of this approach is that models of activity, etc. can be built using motion capture data, then transferred to use on image observations, and this is what we do.

2.1. Transducing the body

Tracking: We track motion sequences with the tracker of [25]; this tracker obtains an appearance model by detecting a lateral walk pose, then detects instances in each frame using the pictorial structure method of [12]. Kinematic tracking is known to be hard (see the review in [14]) and, while the tracker is usable, it has some pronounced eccentricities (Figure 2).

Lifting: The tracker reports a 2D configuration of a puppet figure in the image (Figure 1), but we require 3D information. Several authors have successfully obtained 3D reconstructions by matching projected motion capture data to image data by matching **snippets** of multiple motion frames [18, 19, 24]. A complete sequence incurs a per-frame cost of matching the snippet centered at the frame, and a frame-frame transition cost which reflects (a) the extent of the movement and (b) the extent of camera motion. The best sequence is obtained with dynamic programming. The smoothing effect of matching snippets — rather than frames — appears to significantly reduce reconstruction ambiguity (see also the review in [14]).

The disadvantage of the method is that one may not have motion capture that matches the image well, particularly if one has a rich collection of activities to deal with. We use a variant of the method. In particular, we decompose the body into four quarters (two arms, two legs). We then match the legs using the snippet method, but allowing the left and right legs to come from different snippets of motion capture. The per-frame cost must now also reflect the difference in camera position in the root coordinate system of the motion capture; for simplicity, we follow [24] in assuming an orthographic camera with a vertical image plane. We choose arms in a similar manner conditioned on the choice of legs, requiring the camera to be close to the camera of the legs.

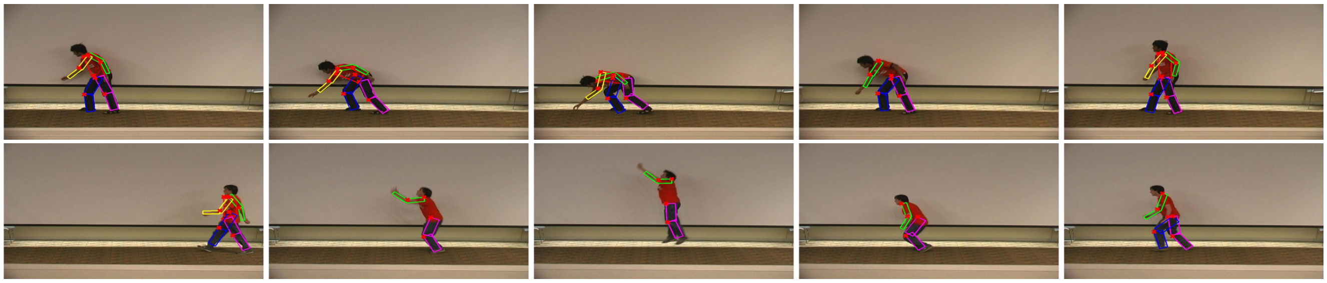


Figure 1. *Here are some example tracks from our video collection. These are two sequences performed by two different actors.* **Top:** `stand-pickup` *sequence.* **Bottom:** `walk-jump-reach-walk` *sequence. The tracker is able to spot most of the body parts in these sequences. However, in most of the sequences, especially in lateral views, only two out of four limbs are tracked.*

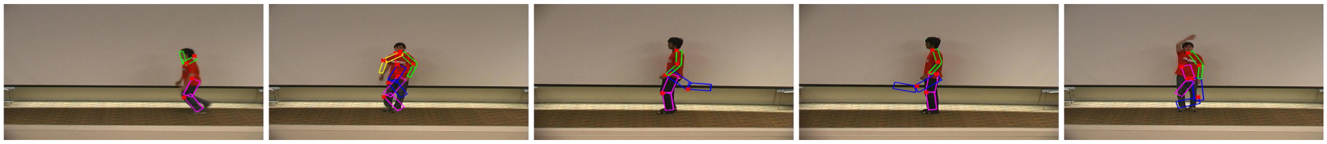


Figure 2. *Due to motion blur and similarities in appearance, some frames are out of track.* **first:** *appearance and motion blur error* **second:** *legs mixed up because of rectangle search failure on legs.* **third and fourth:** *one leg is occluded by the other leg, the tracker tries to find second leg, mistaken by the solid dark line* **fifth:** *motion blur causes tracker to miss the waving arm, legs scrambled. Note that all such bad tracks are a part of our test collection and non-perfect tracking introduces considerable amount of noise to our motion understanding procedure.*

In practice, this method is able to obtain lifts to quite rich sequences of motion from a relatively small motion capture collection.

2.2. Acts and Activities

Acts in short timescales: Individual frames are a poor guide to what the body is up to, not least because transduction is quite noisy and the frame rate is relatively high (15-30Hz). We expect better behaviour from short runs of frames. At the short timescale, we represent motion with three frame long snippets of the lifted 3D representation. We form one snippet for each leg and one for each arm; we omit the torso, because torso motions appear not to be particularly informative in practice. Each limb in each frame is represented with the vector quantized value of the snippet centered on that frame. We use 40 as the number of clusters in vector quantization, for each limb.

Limb models: Using a vague analogy with speech, we wish to build a large dynamical model with the minimum of parameter estimation. We first build a model of the activity of each limb (arms, legs) for a range of actions, using HMM's that emit vector quantized snippets. We choose a set of 9 activities by hand, with the intention of modelling our motion capture collection reasonably well; the collection is the research collection of motion capture data released by Electronic Arts in 2002, and consists of assorted football movements. Motion sequences from this collection are sorted into activities using the labelling of [2]. This labelling is adapted to have separate action marks for each limb. Since actions like `wave` cannot be definable for legs, we only used a subset of 6 activities for labelling legs. For each activity, we fit to the examples using maximum likelihood, and searching over 3-10 state models. Experimentation with the structures shows that 3-state models represent the data well enough. Thus, we take 3-state HMMs as our smallest unit for activity representation.

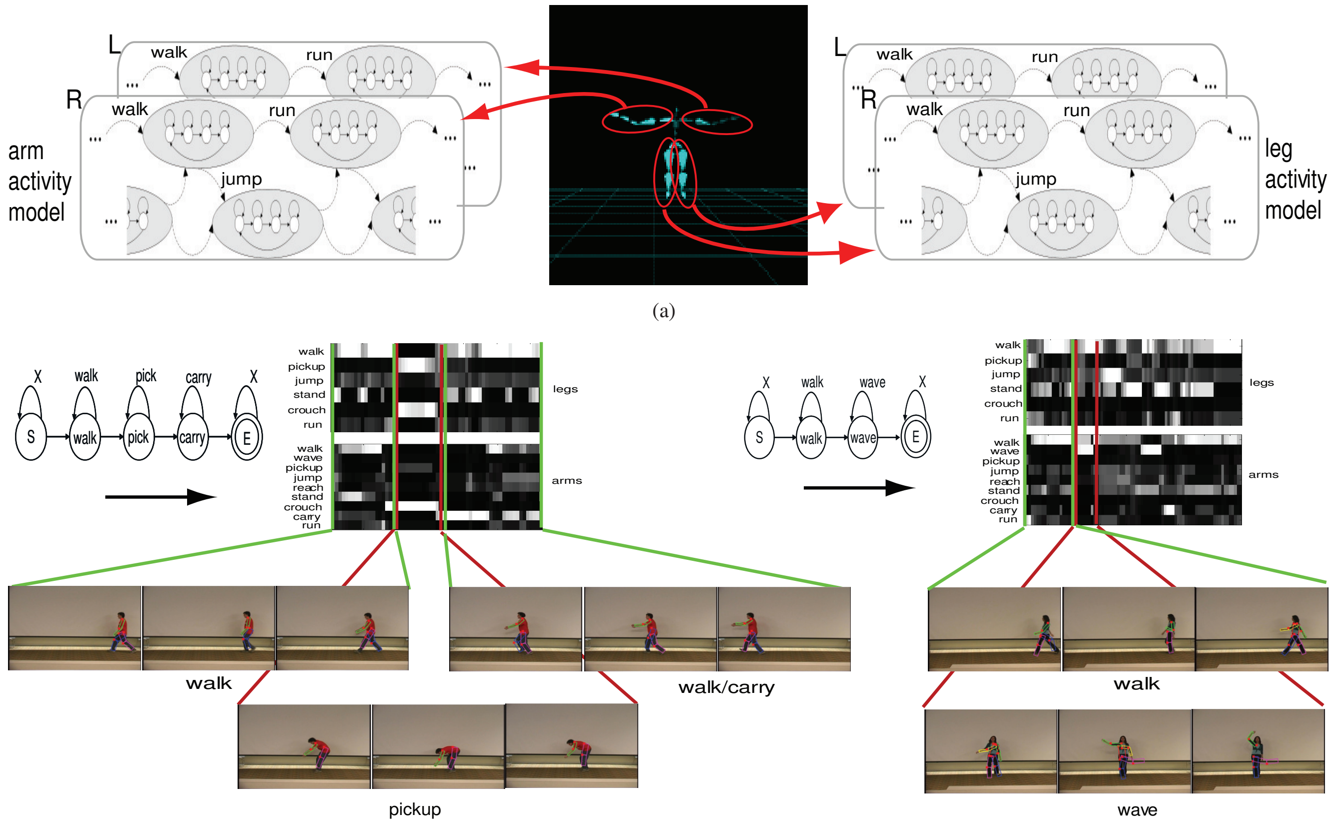
Activity models: We now string the limb models into a larger HMM by linking states that have similar emission probabilities. That is, we put a link between states m and n of the different action models A and B if the distance

$$dist(A_m, B_n) = \sum_{o_m=1}^{N} \sum_{o_n=1}^{N} p(o_m)p(o_n)C(o_m, o_n) \quad (1)$$

is minimal. Here, $p(o_m)$ and $p(o_n)$ are the emission probabilities of respective action model states A_m and B_n, N is the number of observations and $C(o_m, o_n)$ is the Euclidean distance between the emissions centers, which are the cluster centers of the vector-quantized 3D joint points.

The result of this linkage is a dynamical model for each limb that has a rich variety of states, but is relatively easily learned. States in this model are grouped by limb model, and we call a group of states corresponding to a particular limb model a **limb activity model** (Figure 3).

Representing the body: We can now represent the body's behaviour for any sequence of frames as $P(\text{limb activity model}|\text{frames})$. The model has been built entirely on motion capture data. As Figure 3 indicates, this representation is quite competent at discriminating between different labellings for motion capture data. In addition, we



(a)

(b)

Figure 3. **(a)** *First, single action HMMs for left leg, right leg, left arm, right arm are formed using motion capture dataset. Actions are chosen by hand to conform with the available actions in this largely synthesized motion capture set (provided by Electronic Arts, consisting of American Football movements). Second, single action HMMs are joint together by linking the states that have similar emission probabilities. This is analogous to joining phoneme models to recognize words in speech recognition. This is loosely a generative model, we compute the probability that each sequence is generated by a certain set of action HMMs. Using a joint HMM like this for each body part, we compute posteriors of sequences. After that, HMM posteriors for right and left parts of the body are queried together.* **(b) Top:** *Average HMM trellises for the legs and arms of sequences* `walk-pickup-carry` *and* `walk-wave-walk` *(which are performed by different actors - male and female) are shown. As it can be seen, maximum likelihood goes from one action HMM to the other as the action in the video changes. This way, we achieve automatic segmentation of activities and there is no need to use other motion segmentation procedures.* **(b) Bottom:** *Corresponding frames from the subsequences are shown. These sequences are correctly labeled and segmented as* `walk-pickup-carry` *and* `walk-wave` *as the corresponding queries are evaluated.*

achieve automatic segmentation of motions using this representation. There is no need for explicit motion segmentation, since transitions between action HMMs simply provide this information.

3. Querying for Activities

We can compute a representation of what the body is doing from a sequence of video. We would like to be able to build complex queries of composite actions, such as carrying while standing, or waving while running. We can address composition across the body because we can represent different limbs doing different things; and composition in time is straightforward with our representation.

This suggests thinking of querying as looking for strings, where the alphabet is a product of possible activities at limbs and locations in the string represent locations in time. Generally, we do not wish to be precise about the temporal location of particular activities, but would rather find sequences where there is strong evidence for one activity, followed by strong evidence for another, and with a little noise scattered about. In turn, it is natural to start by using regular expressions for motion queries (we see no need for a more expressive string model at this point).

An advantage of using regular expressions is that it is relatively straightforward to compute

$$\sum_{\text{strings matching RE}} P(\text{string}|\text{frames}) \qquad (2)$$

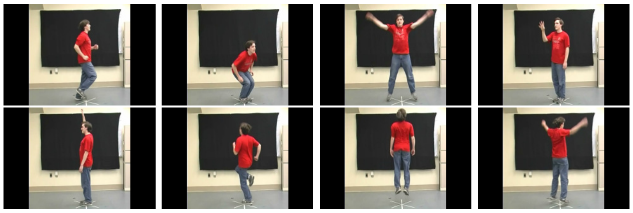


Figure 4. *Example frames from our dataset of single activities with different views.* **Top row:** *Jogging 0 degrees, Jump 45 degrees, jumpjack 90 degrees, reach 135 degrees.* **Bottom row:** *wave 180 degrees, jog 225 degrees, jump 270 degrees, jumpjack 315 degrees.*

which we do by reducing the regular expression to a finite state automaton and then computing the probability this reaches its final state using a straightforward sum-product algorithm.

A tremendous attraction of this approach is that no visual example of a motion is required to query; once one has grasped the semantics of the query language, it is easy to write very complex queries which are relatively successful. The alphabet from which queries are formed consists in principle of $6^2 \times 9^2$ terms (one has one choice each for each leg and each arm). We have found that the tracker is not sufficiently reliable to give sensible representations of both legs (resp. arms). It is often the case that one leg is tracked well and the other poorly. We therefore do not attempt to distinguish between legs (resp. arms), and reduce the alphabet to terms where either leg (resp. either arm) is performing an action; this gives an alphabet of 6×9 terms (one choice at the leg and one at the arm). Using this alphabet, we can write complex composite queries, for example, searching for strings that have several `(l-walk; a-walk)`'s followed by several `(l-stand; a-wave)` followed by several `(l-walk; a-walk)` yields sequences where a person walks into view, stands and waves, then walks out of view.

4. Experimental Results

Using limb activity models, we can do complex activity search with fair accuracy. Our method is insensitive to the clothing or the viewing direction of the subject.

Datasets: We collected our own set of motions, involving three subjects wearing a total of five different outfits in a total of 73 movies (15Hz). Each video shows a subject instructed to produce a complex activity. The sequences differ in length. The complete list of activities collected is given in Table 1.

For viewpoint evaluation, we collected videos of 5 actions: jog, jump, jumpjack, wave and reach. Each action is performed in 8 different directions to the camera, making a total dataset of 40 videos (30Hz). Figure 4 shows example frames of this dataset.

Performance over a set of queries is evaluated using mean average precision (MAP) of the queries. Average precision of a query is defined as the area under the precision-recall curve for that query and a higher average precision value means that more relevant items are returned earlier.

Context	# videos	Context	# videos
crouch-run	2	run-backwards-wave	2
jump-jack	2	run-jump-reach	5
run-carry	2	run-pickup-run	5
run-jump	2	walk-jump-carry	2
run-wave	2	walk-jump-walk	2
stand-pickup	5	walk-pickup-walk	2
stand-reach	5	walk-stand-wave-walk	5
stand-wave	2	crouch-jump-run	3
walk-carry	2	walk-crouch-walk	3
walk-run	3	walk-pickup-carry	3
run-stand-run	3	walk-jump-reach-walk	3
run-backwards	2	walk-stand-run	3
walk-stand-walk	3		

Table 1. *Our collection of video sequences, named by the instructions given to actors.*

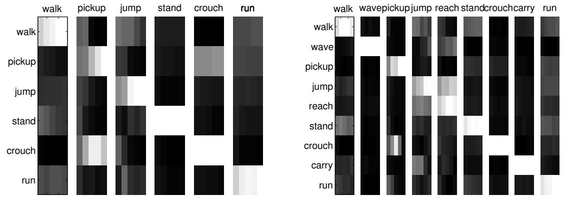


Figure 5. *Local dynamics is quite a good guide to a motion in the motion capture data set. Here we show HMM interpretation of these dynamics. Each column represents 5 frame average HMM trellises for the motion capture sequences (***left:***legs* **right:***arms). This image can also be interpreted as a confusion matrix between actions. Most of the confusion occurs between dynamically similar actions. For example, for* `pickup` *motion, the leg HMMs may fire* `pickup` *or* `crouch` *motions. These two actions are in fact very similar in dynamics. Likewise, for* `reach` *motion, arm HMMs show higher posteriors for* `reach`, `wave` *or* `jump` *motions.*

Limb activity models were fit using a collection of 10938 frames of motion capture data released by Electronic Arts in 2002, consisting of assorted football movements. We choose a set of 9 activities by hand, with the intention of modelling our motion capture collection reasonably well. While these activities are abstract building blocks, the leg models correspond reasonably well to: run, walk, stand, crouch, jump, pickup (total of 6 activities). Similarly, the arm models correspond reasonably well to: run, walk, stand, reach, crouch, carry, wave, pickup, jump motions (total of 9 activities). Figure 5 shows the posterior for each model applied to labelled motion capture data; this can be interpreted as a class confusion matrix.

Searching: We have evaluated several different types of search. In the first type, we employed queries where legs and arms are doing different motions simultaneously.

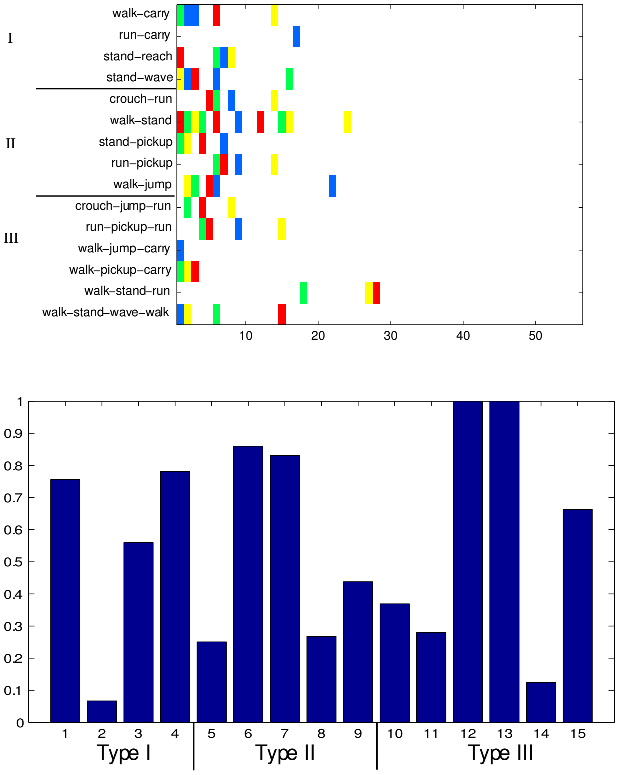


Figure 6. *Our representation can give quite accurate results for complex motion queries, regardless of the clothing worn by the subject.* **Top:** *The results of ranking for 15 queries over our video collection (using k=40 in k-means). In this image, a colored pixel indicates a relevant video. An ideal search would result in an image where all the colored pixels are on the left of the image. Each color represents a different outfit. Note that the choice of the outfit doesn't affect the performance. We have three types of query here. Type I: single activities where there is a different action for legs and arms (ex: walk-carry). Type II: two consecutive actions like* `crouch` *followed by a* `run`. *Type III: activities that are more complex, consisting of three consecutive actions where different body parts may be doing different things (ex:* `walk-stand-walk` *for legs;* `walk-wave-walk` *for arms.* **Bottom:** *Average precision values for each query. Overall, mean average precision for HMM models using 40 clusters is 0.5636.*

In the second type, we evaluated queries where there are two consecutive actions same for legs and arms. In the third type, we searched for motions that are more complex, three consecutive actions where different limbs may be doing different things. We evaluate our searches by first identifying an activity to search for, then marking relevant videos, then writing a regular expression, and finally determining the recall and precision of the results ranked by $P(\text{FSA in end state}|\text{sequence})$. On the traditional simple queries (walk, run, stand), MAP value is 0.9365, only a short sequence of run action is confused with walk action.

Figure 6 shows search results for more complex queries. Our method is able to respond to complex queries quite effectively. The biggest difficulty we faced was to find the perfect track for each limb, due to the discontinuity in track paths and left/right ambiguity of the limbs. That's why some sequences are identified poorly.

SVM classifier over 2d tracks: To evaluate the effectiveness of our approach, we also implemented an SVM-based action classifier over the raw 2D tracks. Using the tracker outputs for 17 videos as training set (chosen such that 2 different video sequences are available for each action), we built action SVMs for each limb separately. We used rbf kernel and 7 frame snippets of tracks to build the classifiers. A grid search over parameter space of the SVM is done and best classifiers are selected using 10-fold cross-validation. The performance of these SVMs are then evaluated over the remaining 56 videos. Figure 7 shows the results. Note that for some queries, SVMs are quite successful in marking relevant documents. However, on the overall, SVMs are penalized by the noise and variance in dynamics of the activities. Our HMM limb activity models, on the other hand, deal with this issue by the help of the dynamics introduced by synthesized motion capture data. SVMs would need a great deal of training data to discover such dynamics.

Viewpoint evaluation: To evaluate our methods invariance to viewpoint, we queried 5 single activities (`jog, jump, jumpjack, reach, wave`) over the data set that has 8 different view directions of subjects. Results are shown in Figure 8. Note that performance is not significantly affected by the change in viewpoint, while there is slight lost in some angles due to tracking difficulties in those view directions.

5. Discussions and Conclusion

There is little evidence that a fixed taxonomy for human motion is available. However, research to date has focused on multi-class discrimination of simple actions. We have demonstrated a representation of human motion that can be used to query for complex activities in a large collection of video, building queries using a regular expression. We are aware of no other method that can give comparable responses for such queries. Our representation uses a generative model, built using both motion capture and video data. Query responses are unaffected by clothing, and our representation is robust to aspect. Our representation significantly outperforms a discriminative representation built using image data alone.

One of the strengths of our method is that, when searching for a particular activity, an example of the motion requested is not required to formulate a query. We use a simple and effective query language; we simply search for activities by formulating sentences like 'Find me action X

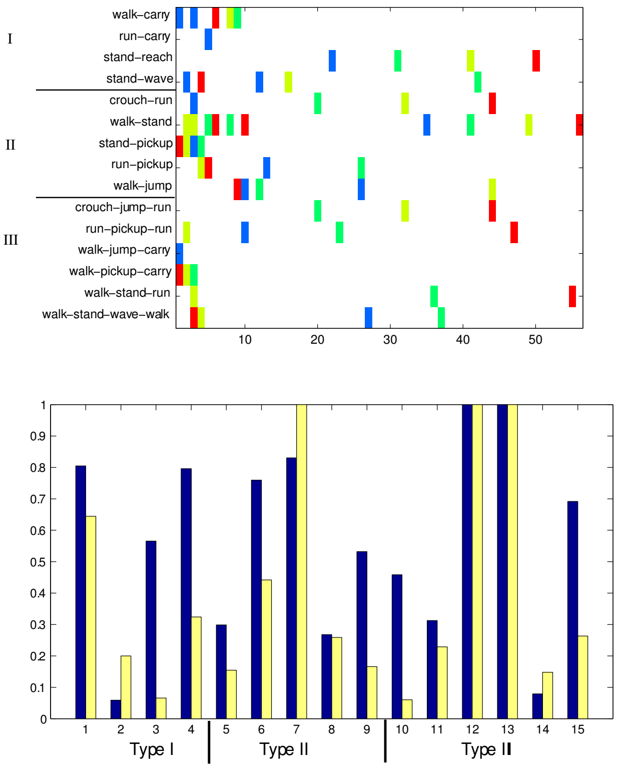
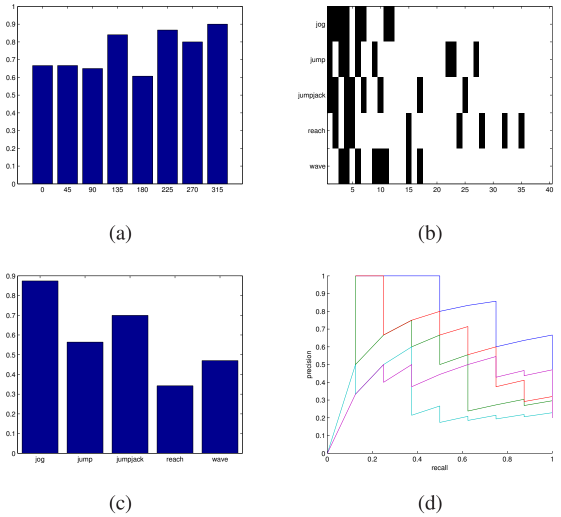


Figure 7. *Composite queries built on top of a discriminative (SVM) based representation are not as successful as queries using our representation. Again, clothing does not affect the result.* **Top:** *Ranking of the queries using SVM-based action classifier. Here, as in Figure 6, the colored pixels show relevant videos and each color represents a different outfit. For some queries, SVM performance is good, however, on the overall, precision and recall rate is low.* **Bottom:** *Average precision comparison of SVM and HMM methods. Our method is shown by blue(dark) bars and SVM method is shown by yellow(light) bars. Mean average precision of the whole query set is 0.3970 for SVM-based action classification, while it is 0.5636 for our HMM-based limb activity models.*

followed by action Y’ or ’Find me videos where legs doing action X and arms doing action Y’ via finite state automata. Matches to the query are evaluated and ranked by the posterior probability of a state representation summed over strings matching the query.

There is much room for improvement; a better tracker would give better results immediately. Further improvements would involve a richer vocabulary of actions, or some theory about how a canonical action vocabulary could be built; a front-end of discriminative features (after [27, 26]); improved lifting to 3D; and, perhaps, a richer query interface.



(a)

(b)

(c)

(d)

Figure 8. *For evaluating our methods invariance to view direction change, we have a separate dataset of single activities* `1-jog 2-jump 3-jumpjack 4-reach 5-wave`. *(a) Average precision values for each viewing direction. Some viewing directions have slightly better performance due to the occlusion of the limbs and poor tracking response to bendings of the limbs in some view directions, however, overall, performance is not significantly affected because of the difference in viewpoint. (b) is the ranking of the five queries of single actions separately. The poorest response comes from* `reach` *action, which inevitably confuses with* `wave`. *(c-d) Respective precision-recall curves and average precision graphs. Also, note that SVM's would need to be retrained for each viewing direction, while our method does not.*

Acknowledgments

We are grateful to Deva Ramanan for the tracker code and many helpful discussions. We also would like to thank Alexander Sorokin for his useful comments.

This work was supported in part by the National Science Foundation under IIS - 0534837 and in part by the Office of Naval Research under N00014-01-1-0890 as part of the MURI program. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect those of the National Science Foundation or the Office of Naval Research. Nazlı İkizler was supported by TÜBİTAK(Turkey).

References

[1] J. Aggarwal and Q. Cai. Human motion analysis: A review. *Computer Vision and Image Understanding*, 73(3):428–440, March 1999.

[2] O. Arikan, D. Forsyth, and J. O'Brien. Motion synthesis from annotations. In *Proc of SIGGRAPH*, 2003.

[3] J. Ben-Arie, Z. Wang, P. Pandit, and S. Rajaram. Human activity recognition using multidimensional indexing. *IEEE*

T. Pattern Analysis and Machine Intelligence, 24(8):1091–1104, August 2002.

[4] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. In *ICCV*, pages 1395–1402, 2005.

[5] A. Bobick. Movement, activity, and action: The role of knowledge in the perception of motion. *Proc. Roy. Soc. B*, 352:1257–1265, 1997.

[6] A. Bobick and J. Davis. The recognition of human movement using temporal templates. *IEEE T. Pattern Analysis and Machine Intelligence*, 23(3):257–267, March 2001.

[7] A. Bobick and A. Wilson. A state based approach to the representation and recognition of gesture. *IEEE T. Pattern Analysis and Machine Intelligence*, 19(12):1325–1337, December 1997.

[8] A. F. Bobick and Y. A. Ivanov. Action recognition using probabilistic parsing. In *CVPR*, page 196, 1998.

[9] M. Brand and V. Kettnaker. Discovery and segmentation of activities in video. *IEEE T. Pattern Analysis and Machine Intelligence*, 22(8):844–851, August 2000.

[10] M. Brand, N. Oliver, and A. Pentland. Coupled hidden markov models for complex action recognition. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 994–999, 1997.

[11] A. A. Efros, A. C. Berg, G. Mori, and J. Malik. Recognizing action at a distance. In *ICCV '03*, pages 726–733, 2003.

[12] P. Felzenszwalb and D. Huttenlocher. Pictorial structures for object recognition. *Int. J. Computer Vision*, 61(1):55–79, January 2005.

[13] X. Feng and P. Perona. Human action recognition by sequence of movelet codewords. In *3D Data Processing Visualization and Transmission*, pages 717–721, 2002.

[14] D. Forsyth, O.Arikan, L. Ikemoto, J. O'Brien, and D. Ramanan. Computational studies of human motion i: Tracking and animation. *Foundations and Trends in Computer Graphics and Vision*, 1(2/3), 2006.

[15] D. M. Gavrila. The visual analysis of human movement: A survey. *Computer Vision and Image Understanding: CVIU*, 73(1):82–98, 1999.

[16] P. Hong, M. Turk, and T. Huang. Gesture modeling and recognition using finite state machines. In *Int. Conf. Automatic Face and Gesture Recognition*, pages 410–415, 2000.

[17] S. Hongeng, R. Nevatia, and F. Bremond. Video-based event recognition: activity representation and probabilistic recognition methods. *Computer Vision and Image Understanding*, 96(2):129–162, November 2004.

[18] N. Howe. Silhouette lookup for automatic pose tracking. In *IEEE Workshop on Articulated and Non-Rigid Motion*, page 15, 2004.

[19] N. R. Howe, M. E. Leventon, and W. T. Freeman. Bayesian reconstruction of 3d human motion from single-camera video. In *Proc. Neural Information Processing Systems*, pages 820–26, 2000.

[20] W. Hu, T. Tan, L. Wang, and S. Maybank. A survey on visual surveillance of object motion and behaviors. *IEEE transactions on systems, man, and cyberneticspart c: applications and reviews*, 34(3), 2004.

[21] T. Mori, Y. Segawa, M. Shimosaka, and T. Sato. Hierarchical recognition of daily human actions based on continuous hidden markov models. In *Int. Conf. Automatic Face and Gesture Recognition*, pages 779–784, 2004.

[22] N. Oliver, A. Garg, and E. Horvitz. Layered representations for learning and inferring office activity from multiple sensory channels. *Computer Vision and Image Understanding*, 96(2):163–180, November 2004.

[23] R. Polana and R. Nelson. Detecting activities. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 2–7, 1993.

[24] D. Ramanan and D. Forsyth. Automatic annotation of everyday movements. In *Proc. Neural Information Processing Systems*, 2003.

[25] D. Ramanan, D. Forsyth, and A. Zisserman. Strike a pose: Tracking people by finding stylized poses. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages I: 271–278, 2005.

[26] C. Sminchisescu, A. Kanaujia, Z. Li, and D. Metaxas. Conditional random fields for contextual human motion recognition. In *ICCV*, pages 1808–1815, 2005.

[27] C. Sminchisescu, A. Kanaujia, Z. Li, and D. Metaxas. Discriminative density propagation for 3d human motion estimation. *CVPR*, 1:390–397, 2005.

[28] A. Wilson and A. Bobick. Learning visual behavior for gesture analysis. In *IEEE Symposium on Computer Vision*, pages 229–234, 1995.

[29] A. Wilson and A. Bobick. Parametric hidden markov models for gesture recognition. *IEEE T. Pattern Analysis and Machine Intelligence*, 21(9):884–900, September 1999.

[30] J. Yamato, J. Ohya, and K. Ishii. Recognising human action in time sequential images using hidden markov model. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 379–385, 1992.

[31] J. Yang, Y. Xu, and C. S. Chen. Human action learning via hidden markov model. *IEEE Transactions on Systems Man and Cybernetics*, 27:34–44, 1997.

[32] T. Zhao and R. Nevatia. Tracking multiple humans in complex situations. *IEEE T. Pattern Analysis and Machine Intelligence*, 26(9):1208–1221, September 2004.