View-invariant modeling and recognition of human actions using grammars

Abhijit S. Ogale, Alap Karapurkar, Yiannis Aloimonos

Computer Vision Laboratory, Dept. of Computer Science
University of Maryland, College Park, MD 20742 USA
Email: {ogale,karapurk,yiannis}@cs.umd.edu

Abstract

In this paper, we represent human actions as short sequences of atomic body poses. The knowledge of body pose is stored only implicitly as a set of silhouettes seen from multiple viewpoints; no explicit 3D poses or body models are used, and individual body parts are not identified. Actions and their constituent atomic poses are extracted from a set of multiview multiperson video sequences by an automatic keyframe selection process, and are used to automatically construct a probabilistic context-free grammar (PCFG). Given a new single viewpoint video, we can parse it to recognize actions and changes in viewpoint simultaneously. Experimental results are provided.

1. Introduction

The motivation for representing human activity in terms of a language lies primarily in the dual ability of linguistic mechanisms to be used for both recognitive and generative purposes. This ability is highly desirable for a representation of human action, since humans (or humanoid robots) must not only recognize actions performed by their peers, but also potentially perform (or generate) these actions themselves. Rizzolatti and Arbib [13] discuss the presence of so-called *mirror neurons* in the monkey brain, which respond when a monkey observes a grasping action, and also when the monkey performs a similar action. Such observations indicate the proximity of recognitive and generative processes in the brain at a very low level, and further add to the appeal of using structures such as grammars for modeling actions, since they too possess such a dual character.

In computer vision, although the developments in recognition and description of human activity are relatively recent, there exist a wide variety of methods, including many which implicitly or explicitly utilize the parallels with language. Due to space constraints, we mention only a few methods which are relevant to the ideas on full body action recognition presented in this paper; for a broader perspective, we refer the interested reader to recent reviews by Aggarwal et al [1] and Wang et al [15], which survey various approaches for human motion analysis including the recognition of human actions. HMM's as well as context-free grammars have previously been used in the recognition of hand and face gestures, but the literature in these areas is extensive, and we will limit ourselves to full body action recognition only. For a review of hand gesture recognition techniques, the reader is referred to [10].

Hidden Markov Models (HMMs) have often been used to express the temporal relationships inherent in human actions. Yamato et al. [16] used mesh features of human silhouettes from a single viewpoint to build one HMM for each action. Bregler et al. [4] describe a four level probabilistic framework for segmentation, tracking and classification of human dynamics. Brand et al. [3] use HMMs and infer 3D pose and orientation from silhouettes, using 3D motion capture data and 2D projections for training. Bobick et al. [2] model actions using a novel representation called temporal templates. Kojima et al. [7] build a verbs hierarchy using case frames to produce textual descriptions of activity. Sullivan et al. [14] develop a view based approach which uses manually selected keyframes to represent and find similar actions in a video using a novel matching algorithm. Rao et al. [11] represent actions using view-invariant dynamic instants found using the spatiotemporal curvature of point trajectories. Davis et al. [5] discuss a reliable inference framework for discriminating various actions. Mori et al. [8] use 3D motion data and associate each action with a distinct feature detector and HMM, followed by hierarchical recognition. Feng et al. [6] model actions using codewords extracted from movelets (spacetime poses constructed by identifying body parts), and estimate the likely movelet codeword sequence with HMMs. Park et al. [9] compute 3D pose from silhouettes for every image and kinetic parameters which are recognized with a hierarchical DFA.

In this paper, we present an approach for using multiview training videos to automatically create view-independent

representations of actions within the framework of a probabilistic context-free grammar. This grammar is then used to parse a new single-viewpoint video sequence to deduce the sequence of actions in a view-invariant fashion.

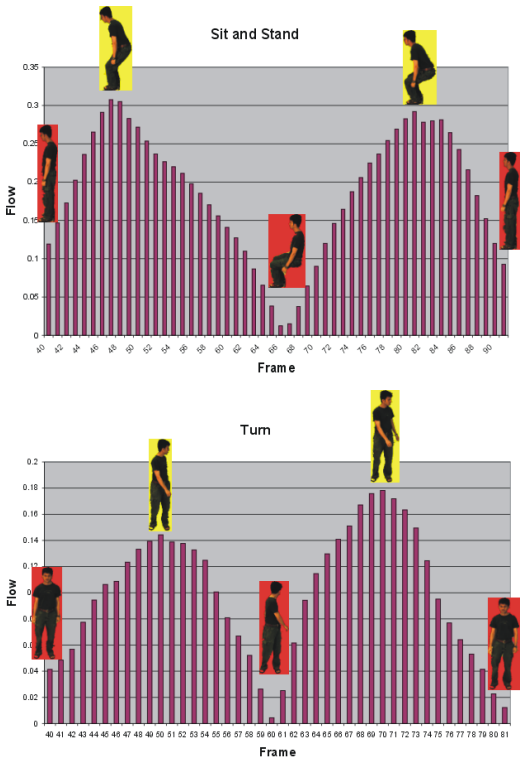

Figure 1. Keyframe extraction demonstration for two videos showing the *sit* and *turn* actions. Plots show the value found using Eq. (1), and the resulting keyframes at the extrema

2. Our approach

We believe that the right place to begin a discussion about actions and their recognition is to first ask the question: what do we really mean by actions? When humans speak of *recognizing* an action, they may be referring to a set of visually observable transitions of the human body such as 'raise right arm', or an abstract event such as 'a person entered the room'. While recognizing the former requires only visual knowledge about allowed transitions or movements of the human body, the latter requires much more than purely visual knowledge: it requires that we know about rooms and the fact that they can be 'entered into' and 'exited from', along with the relationships of these abstract linguistic verbs to lower level verbs having direct visual counterparts. In this paper, we shall deal with the automatic view-invariant recognition of low level visual verbs which only involve the human body. The visual verbs enforce the visual syntactic structure of human actions (allowed transitions of the body and viewpoint) without worrying about semantic descriptions.

In our framework, each training verb or action a is described by a short sequence of key pose pairs $a = ((p_1, p_2), (p_2, p_3), ..., p_k)$, where each pose $p_i \in P$, where P is the complete set of observed (allowed) poses. Note that for every consecutive pair, the second pose in the earlier pair is the same as the first pose in the latter pair, since they correspond to the same time instant. This is because what we really observe in a video is a sequence of poses, not pose pairs. Hence, if we observe poses (p_1, p_2, p_3, p_4) in the video, then we build the corresponding pose pairs as $((p_1, p_2), (p_2, p_3), (p_3, p_4))$.

Each pose p_i is represented implicitly by a family of silhouettes (images) observed in m different viewspoints, i.e. $p_i = (p_i^1, p_i^2, ..., p_i^m)$. The set of key poses and actions is directly obtained from multi-camera multi-person training data without manual intervention. A probabilistic context-free grammar (PCFG) is automatically constructed to encapsulate the knowledge about actions, their constituent poses, and view transitions. During recognition, the PCFG is used to find the most likely sequence of actions seen in a single viewpoint video. Let us explore these steps in detail.

2.1. Keyframe extraction

In this paper, we do not deal with background subtraction, which is a widely studied topic of research in itself. The sequences we have used were obtained using a white background which make background subtraction a straightforward task. We have also experimented with a combination of motion, depth and appearance-based background subtraction techniques to extract silhouettes from monocular or stereoscopic videos without specially created backgrounds; however, in this paper, we avoid discussing background subtraction but focus on subsequent processes for representing and recognizing actions.

Given a sequence (after detecting the human silhouette using background subtraction), the issue at hand is how to find a representative sequence of key poses to describe the action being seen. For a given sequence of frames, we define a *keyframe* to be a frame where the average of the optical flow *magnitude* of foreground pixels (pixels lying inside the human silhouette) reaches an extremum. Note that the optical flow is measured in the reference frame of the foreground, i.e. the mean optical flow of the foreground is first subtracted from the flow value at each foreground pixel. Hence, given frames $f_1, ...f_n$, and the 2D optical flow $\vec{u}_1(x, y), ..., \vec{u}_n(x, y)$ for each frame, we find extrema of the

discrete function (see Figure 1)

$$K_i = \frac{1}{N_i} \sum_{(x,y) \in foreground_i} |\vec{u}_i(x,y) - \vec{u}_i^{mean}| \quad (1)$$

where N_i is the number of foreground pixels and $\vec{u}_i^{mean}$ is the mean foreground flow in frame f_i. In other words, these are points of high average acceleration. The intuition behind this criterion is that frames where this value reaches a minimum indicate flow reversals which occur when the body reaches an extreme pose. Frames at the maxima are points where the body is exactly in between two extreme configurations, and is in the middle of a transition undergoing large overall movement.

Since our training videos consist of *synchronized* multiview data for each action, we perform keyframe extraction in each view separately, and each view v yields a set of key time instants $\{t_1^v, t_2^v, t_3^v ...\}$. For each action a, the union of these sets of key time instants from all the views gives the complete set of key time instants $\{t_1, t_2, t_3 ...\}$ for that action. Corresponding to each key time instant t_i, we obtain a pose p_i as a multiview set of silhouette images $p_i = (p_i^1, p_i^2, ..., p_i^m)$. Thus, each action is represented by a short sequence of key multiview pose pairs as described earlier. The entire process requires no human intervention. The keyframe extraction process is fairly robust and not sensitive to the accuracy of optical flow estimation, since it only uses averages of the flow.

2.2. Creating a PCFG

In this section, we discuss a method to automatically construct a PCFG using our multiview training dataset, which is separate from our single-view test dataset. Note that we are *specifying* a PCFG, and not *learning* it, hence the term *training data* is not being used in the strictest sense. In the previous step, we used multiview training videos to find a sequence of key poses for all the training actions. From this data, we wish to find out the complete set of *unique* key poses of the body. It is clear that a particular key pose (such as 'standing upright') may be common to many actions. However, since we used independent training videos for each action, we must first find identify such common poses automatically, so that we avoid redundant representations. Hence, given a set of training actions $\{a, b, c...\}$, and the recovered multiview pose sequence pairs for each action, i.e. $a \equiv ((pa_1, pa_2), (pa_2, pa_3), ...)$, $b \equiv ((pb_1, pb_2), (pb_2, pb_3), ...)$ and so on, the task is to identify the complete set $P = \{p_1, p_2, p_3 ..., p_n\}$ of unique poses, where a pose $p_i \in P$ represents (say) equivalent poses pa_1, pb_4, pc_2.

To do this, we can first create the set $PO = \{pa_1, pa_2, ..., pb_1, pb_2, ..., pc_1, pc_2, ...\}$of all observed key poses (with possible repetitions) from all actions. If the silhouettes for two poses p_i and p_j match in each of the m views, the two poses are considered to be the same. We register two silhouette images using phase correlation [12] in the cartesian and logpolar space, which is invariant to 2D translation, rotation and scaling. In the registered images, the ratio of the sizes of the intersection set (overlap) of the silhouettes to the union set must be close to 1 for the silhouettes to match, which is decided with a threshold. If the silhouettes for two poses match in all the views, the poses are considered to be the same. This procedure allows us to map the observed set of key poses PO to a smaller set of unique key poses P. After this is done, each action is relabeled using the mapping from $PO \rightarrow P$, so that we finally get representations such as $a \equiv ((p_5, p_2), (p_2, p_7), ...)$, $b \equiv ((p_3, p_5), (p_5, p_1), ...)$ and so on. Now we are ready to construct the PCFG; this process is summarized in Figure 2.

$V \rightarrow A \mid AA \mid ... \mid A^f$	$\forall i, p(A^i \mid V) = 1/f$
$A \rightarrow A_1 \mid A_2 \mid .... \mid A_g$	$\forall i, p(A_i \mid A) = 1/g$
$A_i \rightarrow q_{ab} q_{bc} q_{cd} ...$	$p(q_{ab} q_{bc} q_{cd} ... \mid A_i) = 1$
$q_{cd} \rightarrow p_c^u p_d^v$	$\sum_{\substack{allowed \\ u,v}} p(p_c^u p_d^v \mid q_{cd}) = 1$
$p_i^v \rightarrow s_k$	$p(s_k \mid p_i^v)$ *obtained at runtime*

Figure 2. Summary of PCFG construction

Let the symbol V denote a sequence of actions, and A denote a particular action. Then each action sequence may be composed of one or more actions in sequence. We allow for upto f consecutive actions with equal probability, by using the productions

$$V \rightarrow A|AA|....|A^f$$

such that each production has a probability of $1/f$. (We have used $f = 5$ in our experiments). The symbol A denotes a specific action, and if we have g actions $A_1, A_2, ..., A_g$ in our training set, then we add the following productions, each having a probability of $1/g$:

$$A \rightarrow A_1|A_2|....|A_g$$

If we denote an ordered pair of poses by $q_{ij} = (p_i, p_j)$, then an action is represented as $A_i = (q_{ab}, q_{bc}, ...,)$. Note the relationship between consecutive indices. Using this notation, we can add a production for every action which

expands it in terms of its pose pair sequence with unit probability:

$$A_i \rightarrow q_{ab}q_{bc}q_{cd}....$$

For each pose pair q_{cd}, we add all possible productions which expand it into its two constituent poses with all possible viewpoints as follows:

$$q_{cd} \rightarrow p_c^u p_d^v$$

Here, the superscripts u and v denote viewpoints, and we add only the productions in which $u = v$, or u is adjacent to v. (the probabilities are kept slightly biased towards $u = v$). In this way, we force the viewpoint to remain constant or change smoothly from one key pose to the next consecutive pose. Note that the total probability for all productions for each q_{cd} is normalized to unity.

This is the only portion of the grammar that can be prespecified. The final productions in the grammar which convert a pose-viewpoint pair p_c^u to an observed silhouette o_i in the input video (the terminal symbols) and their associated probabilities are specified at runtime.

Recall that each pose-viewpoint pair p_c^u is associated with a silhouette image. In the actual implementation, we use sequences of many persons performing each action as training data. The only modification required because of this, is that we average silhouettes of different persons (after registration using phase correlation) in the same viewpoint and pose, and hence the final silhouettes associated with each p_c^u are non-binary.

2.3. View invariant recognition of pose sequences

Given a new single camera video sequence of a person performing some actions, we perform keyframe extraction on it to obtain an observed sequence of silhouettes $(s_1, s_2, ..., s_n)$. We can now compare each observed silhouette s_k with the silhouette s_i^v corresponding to every pose-viewpoint pair p_i^v as follows: first we register s_k to s_i^v using the phase correlation procedure mentioned in the previous section to remove 2D translation, rotation and scaling. Then, we compute a matching measure $m(s_k, s_i^v)$ between the two silhouettes, which finds the ratio of their area of intersection to the area of their union. This measure is close to 1 for matching silhouettes. Now, we find the probability of p_i^v being a good match given the observed silhouette s_k to be

$$P(p_i^v|s_k) = \frac{m(s_k, s_i^v)}{\sum_{all\ s_i^v} m(s_k, s_i^v)}$$

But what we want is the probability for the production $p_i^v \rightarrow s_k$ which is denoted by $P(s_k|p_i^v)$. We can write this using Bayes theorem as follows:

$$P(s_k|p_i^v) = \frac{P(p_i^v|s_k)P(s_k)}{P(p_i^v)}$$

We assign equal values to all $P(p_i^v)$ (so that each pose-viewpoint can possibly be the starting state), and the unknown $P(s_k)$ will only contribute an overall constant multiplying factor $P(s_1)P(s_2)...P(s_n)$ when we apply the parsing algorithm. Thus, we can use the scaled likelihood $P(p_i^v|s_k)/P(p_i^v)$ in place of $P(s_k|p_i^v)$.

Thus, we complete our PCFG at runtime by creating this final set of productions $p_i^v \rightarrow s_k$ and probabilities $P(s_k|p_i^v)$, and we can then parse the video into the constituent actions, which yields a parse tree identifying the observed sequence of actions and transitions in viewpoint.

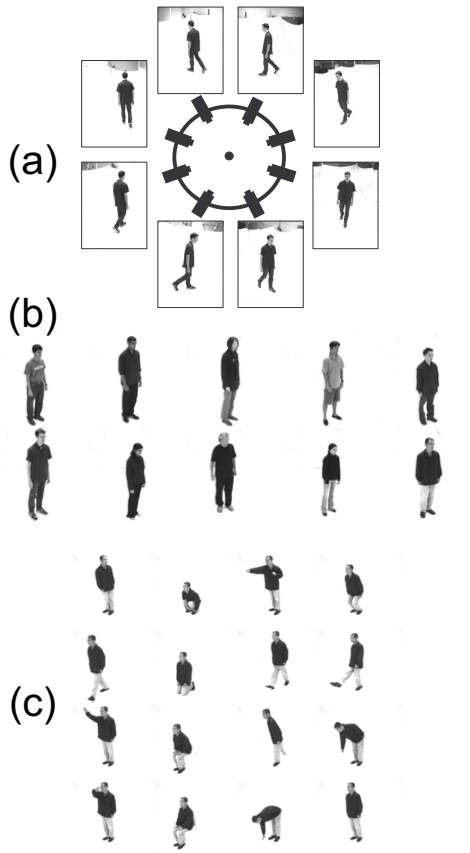

Figure 3. (a) Eight viewpoints were used (b) Ten people performed various actions (c) Some key poses for a person seen in one of the views

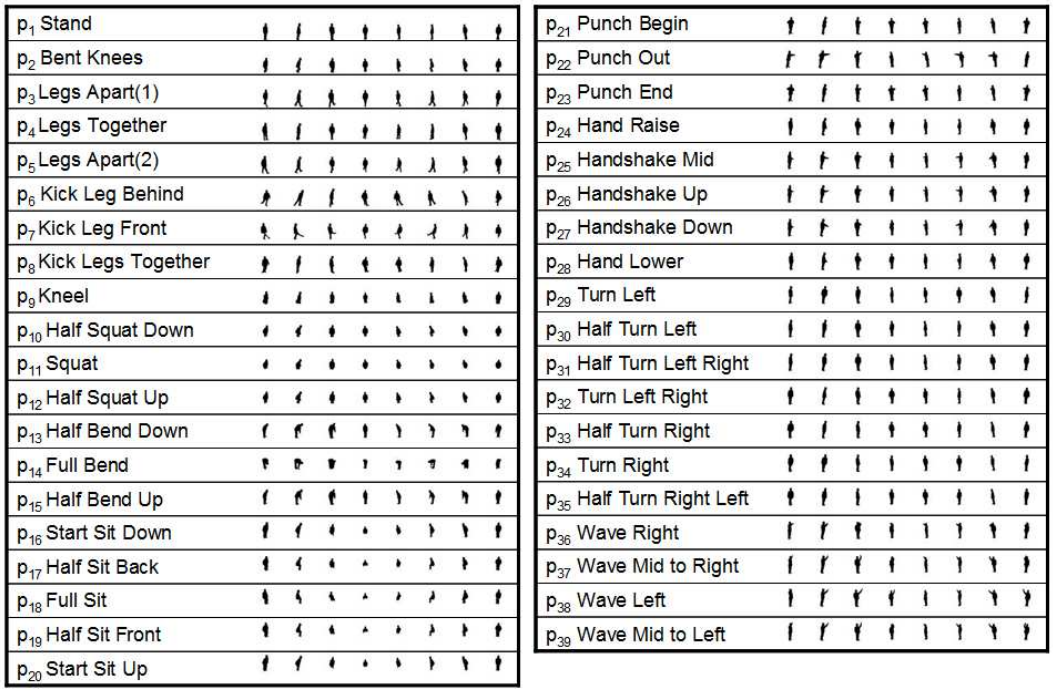

Figure 4. Set of 39 unique 3D key poses extracted from all the videos in the training dataset. Each pose is shown as a collection of silhouettes in eight viewpoints.

3. Experiments

We have used the PCFG implementation in the Natural Language Toolkit (NLTK at *http://nltk.sourceforge.net*), which incorporates a Viterbi-style parser for PCFG's. Our multiview data used for specifying the PCFG consisted of 11 actions (Walk, Jump, Pickup, Kick, Kneel, Squat, Punch, Turn, Sit, Wave, Handshake) being performed by 10 people and seen from 8 views, where the cameras are arranged in a *surround* configuration (see Figure 3 to see a sample of the dataset). Note that the actions have been given names (like Kneel) for presentation purposes. The extracted set of unique 3D key poses is shown in Figure 4 (text annotations are again included only for presentation purposes).

Our test dataset, which is different from the training dataset, consists of single camera video sequences. Figure 5 shows a result for the case where a person performs four actions in sequence. The most probable parse is shown in the figure, which clearly identifies the four actions (*walk, turn, kick and kneel).* Figure 6 shows a sequence where a person walks while turning, and then stops to pickup something. Only the deduced changes in viewpoint obtained after parsing are shown, and the viewpoint change is clearly observable, since the orange squares which indicate the deduced viewpoint shift from left to right, as we move downward, indicating a smooth transition between views. Equivalently, we could also use the case where the camera rotates around the person while the action is being performed. The results demonstrate that the presented method is capable of dealing with changes in viewpoint and pose simultaneously.

If we wish to correspond visually observed states to words, we must augment the existing system of poses with text annotations. A hierarchy of verbs can be constructed, which will contain observable visual verbs (such as *kneel*) at the bottom, and abstract verbs (such as *enter*) higher up. In a complete system for describing human activity, these visuo-linguistic relationships must be used together to parse an input video. Another limitation of this system is that it describes actions using static poses; this limitation can be overcome by using a pose plus motion representation. We are experimenting with such representations, and the initial results have been promising. The proposed model can also be extended to include multiple actors and interaction with objects, by including similar models for such actors and objects.

4. Conclusions

To summarize, we have presented a method for view-invariant action recognition using a probabilistic context-free grammar (PCFG). The PCFG construction process is

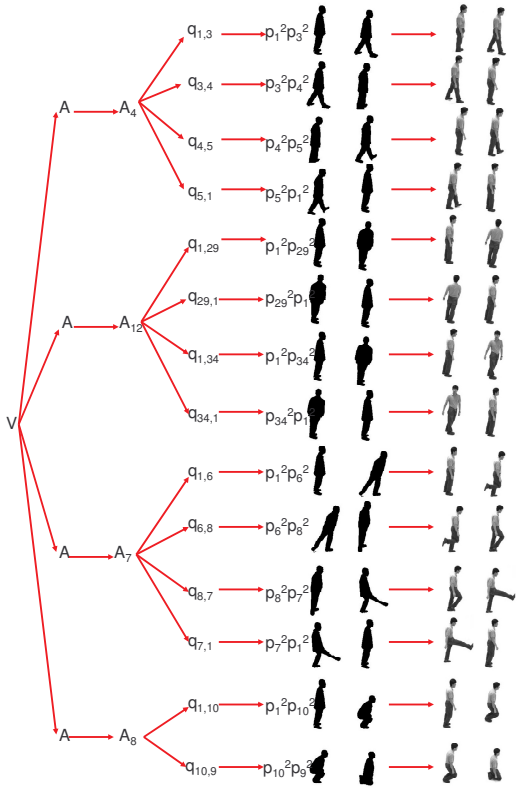

Figure 5. Parse tree obtained for the input video whose keyframes are arranged in pairs shown on the right. The parsed sequence consists of four actions A_4, A_{12}, A_7, A_8 (which we can also call *walk, turn, kick, kneel* respectively)

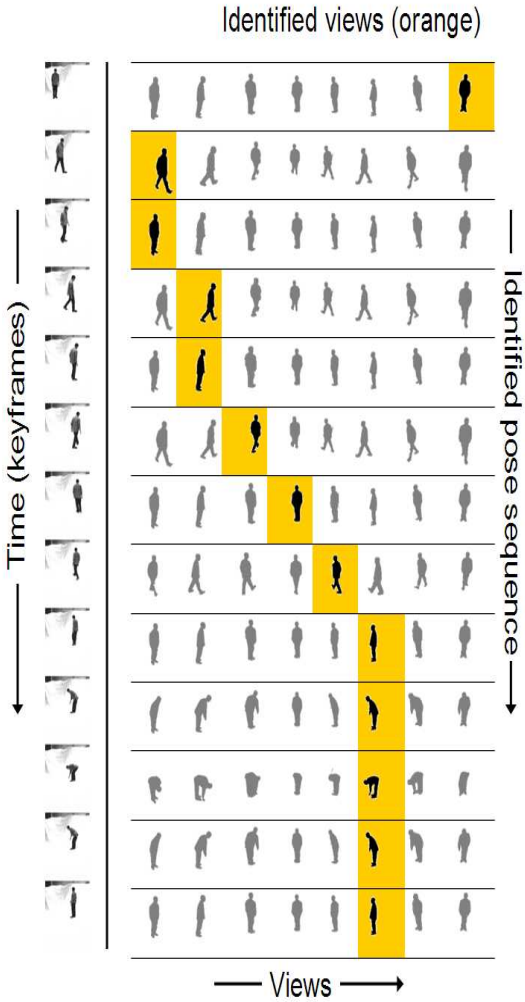

Figure 6. Changing viewpoint: Left hand column shows detected keyframes in the input (time increases from top to bottom). Person turns while walking, and then picks something up. Each row containing eight images on the right hand side collectively describes a 3D pose. Each element of the row shows a viewpoint. Detected viewpoints are marked in orange. Note that the figure does not display the parse tree, but only changes in viewpoint.

completely automatic and uses multiview data. The recognition process is also completely automatic, and parses a single viewpoint video to deduce actions and changes in viewpoint simultaneously. We have presented preliminary experimental results to demonstrate the abilities of the proposed method, and discussed possible extensions.

References

[1] J. Aggarwal and S. Park. Human motion: Modeling and recognition of actions and interactions. *Proceedings of the 2nd International Symposium on 3D Data Processing, Visualization and Transmission*, pages 640–647, 2004.

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

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

[4] C. Bregler. Learning and recognizing human dynamics in video sequences. *Proceedings of the IEEE Conf. Computer Vision and Pattern Recognition*, pages 568–574, 1997.

[5] J. W. Davis and A. Tyagi. A reliable-inference framework for recognition of human actions. *Proceedings of the IEEE Conference on Advanced Video and Signal Based Surveillance*, pages 169–176, 2003.

[6] X. Feng and P. Perona. Human action recognition by sequence of movelet codewords. *Proceedings of First International Symposium on 3D Data Processing Visualization and Transmission*, pages 717–721, 2002.

[7] A. Kojima, T. Tamura, and K. Fukunaga. Natural language description of human activities from video images based on concept hierarchy of actions. *IJCV*, 50(2):171–184, November 2002.

[8] T. Mori, Y. Segawa, M. Shimosaka, and T. Sato. Hierarchical recognition of daily human actions based on continuous hidden markov models. *Proceedings of IEEE International Conference on Automatic Face and Gesture Recognition*, pages 779 – 784, 2004.

[9] J. Park, S. Park, and J. K. Aggarwal. Model-based human motion tracking and behavior recognition using hierarchical finite state automata. *Lecture Notes in Computer Science, Proceedings of ICCSA*, 3046:311–320, 2004.

[10] V. Pavlovic, R. Sharma, and T. Huang. Visual interpretation of hand gestures for human-computer interaction: a review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):677–695, 1997.

[11] C. Rao, A. Yilmaz, and M. Shah. View-invariant representation and recognition of actions. *International Journal of Computer Vision*, 50(2):203–226, 2002.

[12] B. S. Reddy and B. Chatterji. An fft-based technique for translation, rotation and scale-invariant image registration. *IEEE Transactions on Image Processing*, 5(8):1266–1271, August 1996.

[13] G. Rizzolatti and M. A. Arbib. Language within our grasp. *Trends in Neurosciences*, 21(5):188–194, 1998.

[14] J. Sullivan and S. Carlsson. Recognizing and tracking human action. *Proceedings of European Conference on Computer Vision*, pages 629–644, 2002.

[15] L. Wang, W. Hu, and T. Tan. Recent developments in human motion analysis. *Pattern recognition*, 36:585–601, 2003.

[16] J. Yamato, J. Ohya, and K. Ishii. Recognizing human action in time sequential images using hidden markov model. *Proceedings of IEEE Conf. Computer Vision and Image Processing*, pages 379–385, 1992.