Model-free Viewpoint Invariant Human Activity Recognition

Zaw Zaw Htike, Simon Egerton, Kuang Ye Chow

Abstract—**The viewpoint assumption is becoming an obstacle in human activity recognition systems. There is increasing interest in the problem of human activity recognition, motivated by promising applications in many domains. Since camera position is arbitrary in many domains, human activity recognition systems have to be viewpoint invariant. The viewpoint invariance aspect has been ignored by a vast majority of computer vision researchers owing to inherent difficulty to train systems to recognize activities across all possible viewpoints. Fixed viewpoint systems are impractical in real scenarios. Therefore, we attempt to relax the infamous fixed viewpoint assumption by presenting a framework to recognize human activities from monocular video source from arbitrary viewpoint. The proposed system makes use of invariant human pose recognition. An ensemble of pose models performs inference on each video frame. Each pose model employs an expectation-maximization algorithm to estimate the probability that the given frame contains the corresponding pose. Over a sequence of frames, all the pose models collectively produce a multivariate time series. In the activity recognition stage, we use nearest neighbor, with dynamic time warping as a distance measure, to classify pose time series. We have performed some experiments on a publicly available dataset and the results are found to be promising.**

Index Terms— Viewpoint invariance, human activity recognition.

I. Introduction

The problem of automatic human activity recognition has become very popular due to its endless promising applications in many domains such as video surveillance, video indexing, computer animation, automatic sports commentary systems, human computer interaction systems, context-aware pervasive systems, smart home systems and other human-centered intelligent systems. There are a number of reasons why human activity recognition is a very challenging problem. Firstly, a human body is non-rigid and has many degrees of freedom, generating infinitesimal variations in every basic movement. Secondly, no two persons are identical in terms of body shape, volume and coordination of muscle contractions, making each person generate unique movements. The above mentioned problems get further compounded by uncertainties such as variation in viewpoint, illumination, shadow, self-occlusion, deformation, noise, clothing and so on. Since the problem is very vast, it is customary for researchers to make a set of assumptions to make the problem more tractable. However, the most common and the biggest assumption made by researchers happen to be the 'fixed viewpoint assumption'. Their systems can recognize activities only from the 'trained' viewpoint. Unfortunately, the fixed viewpoint assumption is not valid in many domains. In video indexing, for example, viewpoint is arbitrary and may not even be stationary. In video surveillance, camera position is again arbitrary. The assumption causes a 'bottleneck' in practical applications [2]. Therefore, the *fixed viewpoint assumption* needs to be removed. We will therefore relax that assumption and present a simple and novel framework to recognize and classify human activities.

A. Related work

Viewpoint invariance refers to the ability of the system to produce consistent results wherever the camera is positioned and however it is orientated. Fig 1 shows a snapshot of a video sequence from multiple images. In the recent literature, there are mainly two branches of research that tackle the viewpoint invariance issue: multiple-camera branch and single-camera branch. In a multiple-camera system, 3D information can be recovered by means of triangulation [3]. Some researchers fuse spatial information from multiple cameras to form what is called a *3D visual hull* [1, 4]. Multiple-camera approach is the most widely investigated approach. Unfortunately, in many domains, applications are limited to single camera. For example, in video indexing, there are no data available from extra cameras. Single-camera approach is significantly more difficult than multi-camera approach [2, 5]. 100% viewpoint invariance has barely been achieved in the single-camera branch. Most of the recent single-camera techniques (for instance [7-8]) are still at best partially invariant to viewpoint. Thus we will focus only on the single-camera or monocular branch. Most single-camera approaches in the literature further branch into two major categories: **model-based approach** and **model-free approach**. 'Model' in this case means model of the human body.

A **model-based** approach, which employs an explicit parametric anthropometric prior and attempts to recover structural information of the human body, is the more investigated approach. A human body is represented by a kinematic tree model or a stick figure, consisting of joints linked by segments. Most model-based systems essentially attempt to extract features from 3D coordinates of various joints of the human body from an image sequence. Therefore 3D joint coordinates are required to be inferred from corresponding 2D joint coordinates from the image sequence, either by direct inverse kinematics or by

Manuscript received January 12, 2011; revised February 7, 2011.

Zaw Zaw Htike is with the School of Information Technology, Monash University, Sunway Campus, Malaysia (e-mail: zaw@infotech.monash.edu.my)

Simon Egerton is also with the School of Information Technology, Monash University, Sunway Campus, Malaysia (email: simon.egerton@infotech.monash.edu.my)

Kuang Ye Chow is with the School of Engineering, Monash University, Sunway Campus, Malaysia (email: kuang.ye.chow@ieng.monash.edu.my)

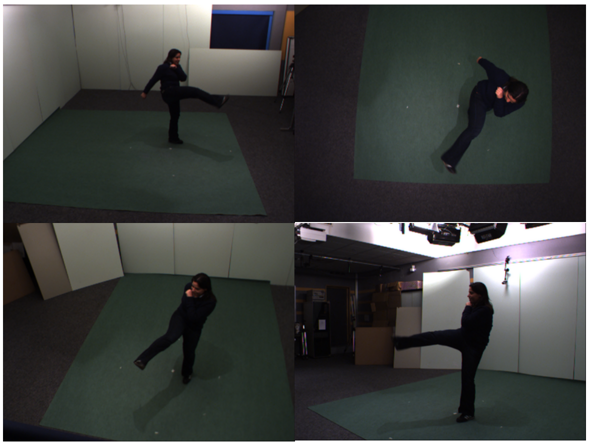


Figure 1: A video frame seen from different viewpoints [1]

constrained nonlinear numerical optimization [6]. Recovering 3D information from 2D information in a single-camera system is inherently an ill-posed problem. Furthermore, model-based systems generally require the output of *body part detection algorithms* as an input [7]. Unfortunately, body part detection is yet another unsolved problem in computer vision.

A **model-free** or model-less approach, on the other hand, makes no attempt to recover structural information of the human body and directly models activities using image-based or shape-based features. Majority of researchers avoid this route because the 2D body shape of a person changes drastically under different viewpoints as shown in Fig 1. Furthermore, model-less systems are generally more prone to overfitting due to redundant features present in images. Souvenir and Babbs [8] employ a 2-dimensional *R* transform and learn the viewpoint manifold using the Isomap algorithm. However, the system is invariant to just 4 DOF of the camera. Lv et al. [9] and Natarajan et al. [10] utilize a novel graphical model where each node stores a 2D representation of a 3D pose from a specific viewpoint. In the recognition phase, silhouette image from each frame is matched against all the nodes in the model using some distance measure. Ji et al. [11] employ exemplar-based hidden Markov models to achieve view-invariance. The work of Weinland et al. [12] is the closest to ours. They employ 3D visual hulls trained from multiple cameras. In the recognition phase, each visual hull is projected into a 2D image that best matches the given silhouette. The first problem with this approach is that projecting a 3D visual hull into a 2D image incurs a very high computation cost because a search is required to find the 2D image that best matches a given silhouette. The second problem is that a system of at least 5 calibrated cameras is required in order to form visual hulls.

B. Overview and contributions

We follow the model-free route to evade the ill-posed problems of body part detection and 3D pose reconstruction. Although the work of Weinland et al. [12] is the most similar to ours, it is quite fundamentally different. Our system achieves view-invariance by employing an ensemble of '*invariant pose models*', instead of 3D visual hulls. The biggest advantage of this is that a pose model can infer poses without any search. In addition, our system handles pose ambiguities well in an elegant multivariate pose series framework.

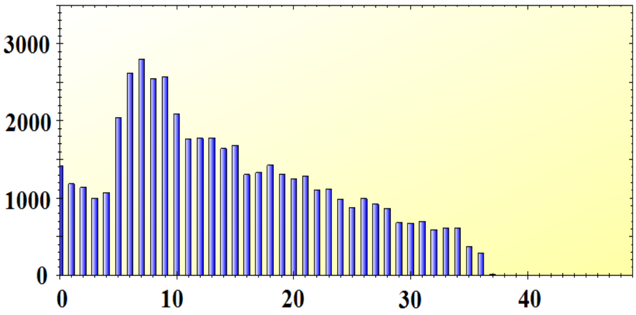


Figure 2: Chord distribution histogram

The paper is organized as follows. We present our pose recognition procedure and activity recognition procedure in Section 2 and 3 respectively. We describe our experiments in Section 4 and conclude in Section 5.

II. POSE RECOGNITION

A human activity is essentially a sequence of static poses. We use the result of viewpoint invariant pose recognition to achieve viewpoint invariant activity recognition. Therefore, static poses are needed to be sequentially labeled first. Pose recognition is the process of recognizing the underlying static poses from static images. Viewpoint invariant pose recognition is done by an ensemble of invariant pose models. Each invariant pose model specializes in recognizing just one pose from 2D images projected from any viewpoint. A camera has 6 degrees of freedom (D.O.F) comprising 3 rotations (namely: raw, pitch and row) and 3 translations alone the X-Y-Z axis. To be viewpoint invariant, we need a robust image presentation. Invariance to 4 D.O.F can be achieved in the feature extraction stage, by extracting rotation, scale and translation (RST) invariant features. The other 2 D.O.F (raw and pitch) have to be handled by the learning model.

For each input 2D image, typical pre-processing steps such as background subtraction, noise removal and silhouette extraction are performed. We then normalize the silhouette centered inside a 128x128 bounding box while preserving the aspect ratio of the original silhouette. A normalized silhouette is translation and scale invariant. Most systems in the literature extract shape features from binary silhouettes. However, because of the fact that binary silhouettes contain a lot of redundant inter-pixel information in them, silhouette-based features generally tend to be more prone to 'over-fitting' in the recognition phase. Furthermore, most binary silhouettes can "losslessly" be represented by their contours. Hence, the proposed system extracts features from the contour of the silhouette. A contour is a sequence of vertices of the silhouette found through a contour extraction algorithm. We resample the contour to have exactly 230 points. We then use chord distribution to present 2D poses [13]. We do this by first calculating all the pair-wise distances between points on the contour and then building a distance histogram as shown in Fig 2. We use 50 bins of length 4 units. We then normalize the histogram by dividing the length of each bar by the length of the highest bar. The normalized chord histogram is rotation, scale and translation invariant. Each pose is essentially represented by a feature vector $\mathbf{x} \in \mathbb{R}^{50}$.

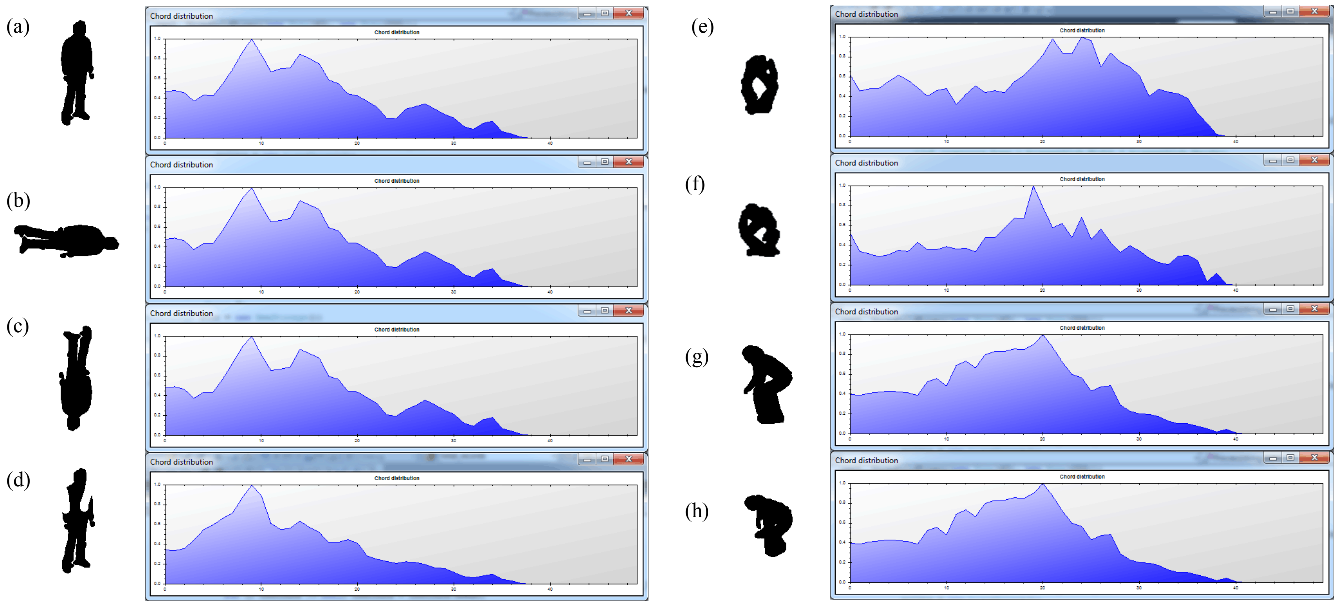


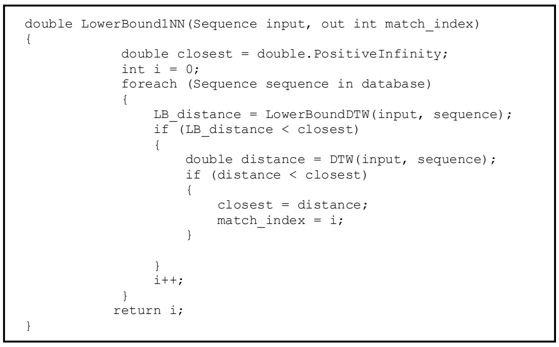
Figure 3: Chord distributions of various poses.

Fig 3 illustrates chord distributions of various poses. Poses in Fig 3b and Fig 3c are rotated versions of the pose in Fig 3a. The chord histograms of all three of those poses are identical. The pose in Fig 3c is the pose in Fig 3a with some significant artifacts added to it. Its chord histogram is still somewhat similar to the three previous histograms. Fig 3e to Fig 3h show histograms of a particular pose from 4 different viewpoints. Those four histograms are similar.

To train the invariant pose models, we used POSER PRO [14] to generate 20 static human poses and wrote a Python script to render each pose from different viewpoints automatically. We varied the *raw* angle and the *pitch* angle of the virtual camera from 0° to 360° with 22.5° step size. Therefore, there are 256 viewpoints in total. Note again that we did not vary the *roll* angle since the system is already invariant to *roll* angle (because of rotation invariance). There are 256 exported images for each pose. Therefore there are 5120 training examples in total (256x20). We employ *expectation-maximization* or EM to perform probabilistic classification. In the training phase, pose maps derived from all the 5120 images are presented to each pose model. Supervised learning is performed where the target value is 1 if the pose belongs to the respective model and 0 if otherwise. After the learning phase, given any input image, each pose model produces the probability that the given image contains the corresponding pose.

III. Activity Recognition

Activity recognition is performed by a higher-level structure to recognize a stream of poses. For each image frame, we perform preprocessing as given in Section 2 to produce a 50-D chord distribution feature vector. Each invariant pose model then produces the probability of the feature vector belonging to the respective pose model. Unlike in *hard competition* and *winner-take-all competition* schemes, rather than taking only most likely pose model with the highest probability level to present the frame, we take all the pose models into account by defining a *pose excitation vector* $\mathbf{z} \in \mathbb{R}^{\boldsymbol{n}}$ where n is the number of pose models. Each component of $\mathbf{z}$ is the probabilistic output of the respective pose model. $\mathbf{z}$ is then normalized so that all the components sum to 1. A sequence of $\mathbf{z}$ forms a multivariate time series. The dimension of the multivariate time series is the number of pose models. Fig 5a illustrates the example of individual univariate time series corresponding to the activity ‘walking’. As walking is a cyclic activity, the individual time series are periodic. Activity recognition is then the process of classifying multivariate series. We classify activities using Nearest Neighbor Algorithm (NN) with Dynamic Time Warping (DTW) as a distance measure. Dynamic Time Warping (DTW) is a well-known algorithm for time series comparison in the literature. DTW minimizes the effect of time shifting, distortion and scaling [15]. Uniform scaling is a desired property in activity recognition due to inherent spatial and temporal variability found in human motion. For example, a person may walk slowly or swiftly. DTW is essentially a global distance measure between two time series. DTW needs a local distance measure between two static points in the two time series. In the case of univariate time series, the local distance, d, between any two points in the time series, is simply the square-difference. For example, $d(3,4) = (3-4)^2$. For our multivariate case, the local distance, d, is the Euclidean distance between the two pose vectors.



```
double LowerBound1NN(Sequence input, out int match_index)
{
            double closest = double.PositiveInfinity;
            int i = 0;
            foreach (Sequence sequence in database)
            {
                LB_distance = LowerBoundDTW(input, sequence);
                if (LB_distance < closest)
                {
                    double distance = DTW(input, sequence);
                    if (distance < closest)
                    {
                        closest = distance;
                        match_index = i;
                    }

                }
                i++;
            }
           return i;
}
```

Figure 4: Pseudocode of 1-NN DTW sequential search algorithm with lower-bounding

$$d(\boldsymbol{a},\boldsymbol{b}) = \sum_{i=1}^{N}(\boldsymbol{a}[i] - \boldsymbol{b}[i])^2 \tag{1}$$

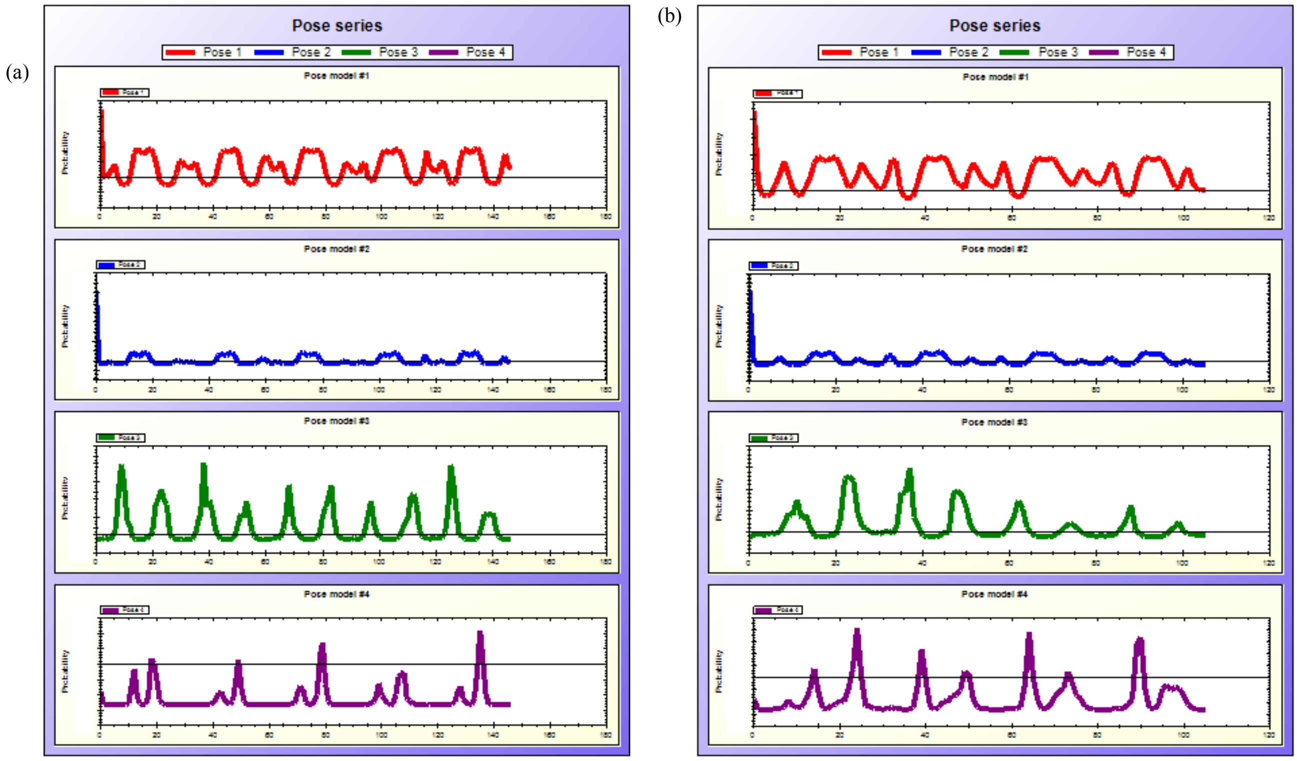


Figure 5: Two pose series corresponding the activity 'walking' performed by two different actors observed from two different viewpoints. Note that only 4 poses are shown for brevity.

where N is the dimension of the multivariate time series. ***N*** is adjustable based on the dimensions of the two pose series. If the pose series **a** and **b** are of different dimensions, N is the dimension of the shorter one.

As a typical NN algorithm, there is no specific learning phase. Our system stores a list of multivariate time series of known activities and their corresponding labels in a database. When an unknown activity is presented to the system, the system takes the unknown time series, performs a sequential search with lower-bounding (as shown in Fig 4 and outputs the label of the known activity which has the shortest global distance from the unknown time series. Fig 5 illustrates two pose series corresponding to the activity 'walking' performed by two different actors observed from two different viewpoints. The two pose series are very similar. It means that the pose series are indeed viewpoint invariant. The system is scalable and suitable to be employed in domains such as video indexing.

Since a stream of video frames may contain many activities consecutively, a long sequence of pose excitation vectors is needed to be temporally segmented such that, ideally, each segment contains a single activity. We use a method of segmentation by motion energy boundaries introduced in [1, 16] by monitoring for small rests or reversals in motion energy.

IV. Experiments

To test our activity recognition system, we used the IXMAS dataset [1]. It is a well-known benchmark dataset in the literature. It contains 13 activities performed 3 times by 10 actors. The actors freely changed their orientation for each acquisition and freely performed all the activities. All the activities were recorded by 5 cameras from different viewpoints. Fig 6 shows part of a clip from the IXMAS dataset from 5 viewpoints.

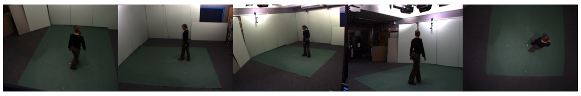


Figure 6: IXMAS dataset

In all our experiments, we use an ensemble of 20 pose models as described in Section 2. We performed *leave-1-camera-out* (L1CO) and *leave-1-actor out* (L1A) cross-validations. The L1CO is intended to focus on the viewpoint invariance property of the system. *Leave-1-camera-out* cross-validation, as the name suggests, takes 1 camera out, trains the system using the data from all the remaining cameras and then tests the system using the data from the camera left out of the training. Since there are multiple combinations of picking one camera out for testing, the whole process was repeated once for each camera. The results were then averaged over all the trials. Similarly, *leave-1-actor-ou*t cross-validation takes 1 actor out, trains the system using the data from all the remaining actors and then tests the system using the data from the actor left out of the training. To study the effect of the number of prior pose models in the system, we repeated each test with various number of pose models by removing some pose models randomly. Table 1 lists the experimental results in terms of accuracy in percentage for all the datasets. N in the table corresponds to the number of prior pose models.

Table 1: Experimental results

N	L1CO	L1AO
20	74.6%	68.6%
10	63.2%	58.5%
5	42.7%	48.1%

Table 2 compares the accuracy of the proposed system with some of the state-of-the-art systems. Most of the state-of -the-art systems in the literature are trained from **multiple views**.

Table 2: Comparison with literature for the IXMAS dataset

Method	Experimental protocols	Accuracy
Weinland et al. (2007) [13]	L1AO multi-view training	74.1%
Weinland et al. (2007) [13]	L1AO **single-view training** best 3 cameras	59.6%
Lv et al. (2007) [9]	L1CO multi-view training	80.6%
Ji et al. (2010) [11]	L1CO multi-view training	83.3%
Ours	L1CO **single-view training**	74.6%

In the literature, Ji et al. [11] achieved the highest recognition rate for on the IXMAS dataset under L1CO cross validation. However, their systems were trained from multiple views. Weinland et al. [12]'s system allows training from single viewpoint. However, its performance deteriorates to 59.6% when trained on single view as shown in Table 2. They also restricted their experiments to using the best 3 cameras in order to avoid ambiguous viewpoints. Our experimental results demonstrate that our system can achieve results on-par with state of the art systems despite the fact that our system was trained from just **one view**. The recognition rates did not plunge as dramatically as expected when we reduced the number of prior pose models. It implies that our system can correctly recognize some activities even if they contain no static poses closely similar to the pose models. This is because of the fuzzy assignment of poses and the usage of the pose excitation vectors instead of crisp poses.

V. Conclusion and Future Work

We have presented a novel framework for viewpoint invariant human activity recognition. The experimental results are quite promising. Our pose recognition system has a lot of room for improvement. We expect the accuracy rates to amplify significantly if we use many more prior pose models and provide more than 1 core training example for each pose (e.g. use different persons with different clothing). A few thousand key poses can be enough to cover everyday activities. The system will not be required to be re-trained upon adding new pose models because N in equation (1) is adjustable based on the dimension of the two pose series.

As future work, we would first like to add several hundreds of pose models. We would also like to find a mechanism of adding new pose models without supplying training examples across many viewpoints. Next, we would like to explore the feasibly of adding new pose models in an unsupervised manner. Furthermore, since DTW was used just for a proof-of-concept, we would like to find the best architecture to classify pose series.

References

[1] D. Weinland, *et al.*, "Free viewpoint action recognition using motion history volumes," *Comput. Vis. Image Underst.*, vol. 104, pp. 249-257, 2006.

[2] X. Ji and H. Liu, "Advances in View-Invariant Human Motion Analysis: A Review," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews,* vol. 40, pp. 13-24, 2010.

[3] H. Yung-Tai, *et al.*, "Human Behavior Analysis Using Deformable Triangulations," in *2005 IEEE 7th Workshop on Multimedia Signal Processing*, 2005, pp. 1-4.

[4] N. Jin and F. Mokhtarian, "Image-based shape model for view-invariant human motion recognition," in *IEEE Conference on Advanced Video and Signal Based Surveillance AVSS 2007*, 2007, pp. 336-341.

[5] C. Sminchisescu, "3D Human Motion Analysis in Monocular Video Techniques and Challenges," presented at the Proceedings of the IEEE International Conference on Video and Signal Based Surveillance, 2006.

[6] A. Agarwal and B. Triggs, "Recovering 3D human pose from monocular images," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 28, pp. 44-58, 2006.

[7] Y. Shen and H. Foroosh, "View-Invariant Action Recognition from Point Triplets," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 31, pp. 1898-1905, 2009.

[8] R. Souvenir and J. Babbs, "Learning the viewpoint manifold for action recognition," in *IEEE Conference on Computer Vision and Pattern Recognition, 2008. CVPR 2008.* , 2008, pp. 1-7.

[9] F. Lv and R. Nevatia, "Single View Human Action Recognition using Key Pose Matching and Viterbi Path Searching," in *IEEE Conference on Computer Vision and Pattern Recognition CVPR '07.*, 2007, pp. 1-8.

[10] P. Natarajan and R. Nevatia, "View and scale invariant action recognition using multiview shape-flow models," in *IEEE Conference on Computer Vision and Pattern Recognition CVPR 2008.* , 2008, pp. 1-8.

[11] X. Ji, *et al.*, "A New Framework for View-Invariant Human Action Recognition," in *Robot Intelligence*, H. Liu, *et al.*, Eds., ed: Springer London, 2010, pp. 71-93.

[12] D. Weinland, *et al.*, "Action Recognition from Arbitrary Views using 3D Exemplars," in *11th International Conference on Computer Vision ICCV 2007*, 2007, pp. 1-7.

[13] S. P. Smith and A. K. Jain, "Chord distributions for shape matching," *Computer Graphics and Image Processing,* vol. 20, pp. 259-271, 1982.

[14] "Pose Pro 2010," ed: Smith Micro, 2010.

[15] P. Senin, "Dynamic Time Warping Algorithm Review," Honolulu, USADecember 2008 2008.

[16] D. Weinland, *et al.*, "Automatic Discovery of Action Taxonomies from Multiple Views," presented at the Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 2, 2006.