Model-Based Viewpoint Invariant Human Activity Recognition from Uncalibrated Monocular Video Sequence

Zaw Zaw Htike[1], Simon Egerton[1], and Ye Chow Kuang[2]

[1] School of Information Technology
[2] School of Engineering
Monash University,
Sunway Campus, Malaysia
{zaw,simon.egerton}@infotech.monash.edu.my,
kuang.ye.chow@eng.monash.edu.my

Abstract. There is growing interest in human activity recognition systems, motivated by their numerous promising applications in many domains. Despite much progress, most researchers have narrowed the problem towards fixed camera viewpoint owing to inherent difficulty to train their systems across all possible viewpoints. Fixed viewpoint systems are impractical in real scenarios. Therefore, we attempt to relax the fixed viewpoint assumption and present a novel and simple framework to recognize and classify human activities from uncalibrated monocular video source from any viewpoint. The proposed framework comprises two stages: 3D human pose estimation and human activity recognition. In the pose estimation stage, we estimate 3D human pose by a simple search-based and tracking-based technique. In the activity recognition stage, we use Nearest Neighbor, with Dynamic Time Warping as a distance measure, to classify multivariate time series which emanate from streams of pose vectors from multiple video frames. We have performed some experiments to evaluate the accuracy of the two stages separately. The encouraging experimental results demonstrate the effectiveness of our framework.

Keywords: Viewpoint invariant, human activity recognition, 3D human pose estimation, Dynamic Time Warping.

1 Introduction

Human activity recognition is the recovery of human motion information from image sequences and labeling of the underlying activities of the human subjects. The problem of automatic human activity recognition has become very popular due to its countless 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

J. Li (Ed.): AI 2010, LNAI 6464, pp. 142–152, 2010.

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. That assumption is causing 'bottleneck' in practical applications [1]. 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.

Fig. 1. Images resulting from variations in camera orientation

1.1 Related Work

There is indeed some confusion among researchers about the term *viewpoint invariance*. Some researchers such as [2] claim 'viewpoint invariance' when their system is only capable of recognizing sequences up to 45° horizontal deviation from frontal view. *Robustness* to some degree of variation in viewpoint is not the same as viewpoint invariance. A camera has 6 *degrees of freedom* (DOF). Viewpoint invariance refers to the ability of the system to produce consistent results wherever the camera is positioned and however it is orientated as shown in Fig 1, be it front-view, side-view, back-view or any intermediate view. In recent literature, there are mainly two branches of research that attack 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* [4-5]. 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 [1; 6]. 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-less approach**.

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 pose is represented by a kinematic tree model or a stick figure, consisting of joints linked by segments. Most of the existing works in the literature in the model-based branch concentrate on a lower-level field of research called 'pose recovery' rather than higher-level activity recognition because human pose recovery (which is a prerequisite to activity recognition) itself is an unsolved problem. A model-based approach estimates human pose either by direct inverse kinematics or by numerical optimization over the pose variables [9]. The two major problems that arise in a single-camera system are depth ambiguity and self-occlusion. Depth ambiguity arises because we are trying to reconstruct 3D skeleton out of 2D information. Recovering 3D information from a single uncalibrated camera is inherently ill-posed because we are trying to solve equations with more unknowns than the number of equations. Researchers try to disambiguate by bringing in more assumptions or constraints in one form or another. For example, Wei and Chai [10] use at least 5 key frames to resolve ambiguity and a numerical constrained optimization algorithm to construct 3D human poses. However, the system is still not so invariant to viewpoint because it does not work for top view. Shen and Foroosh [11] model a sequence of poses as a sequence of planes defined by triplets of body points. Despite good results, it cannot handle self-occlusion.

A model-less approach makes no attempt to recover structural information of the human body. Most model-less approaches such as [9] are example-based, that is they utilize machine learning techniques to construct a mapping function between 2D image features and 3D poses. Some researchers such as [7] find a common lower dimensional representation of the projected image of the same pose under different viewpoints. However, these kinds of approaches have not been demonstrated to be able to handle self-occlusion. The main disadvantage of model-less approaches is that the training examples should be very diverse and numerous so as to correctly map unknown poses. They are also generally more prone to overfitting.

1.2 Contributions

This paper has two major contributions: viewpoint invariant 3D human pose estimation and viewpoint invariant human activity recognition. We follow the model-based route. However, we show that full body pose recovery is not necessary to recognize activities. Unlike previous work in the activity recognition literature, we present a robust technique that can recognize activities from partial joint information such as when half of the body is missing. To be invariant to viewpoint, the system needs to be able to function correctly even with 'crippled' input. We demonstrate how our activity recognition system achieves full invariance to viewpoint under 6 DOF of camera.

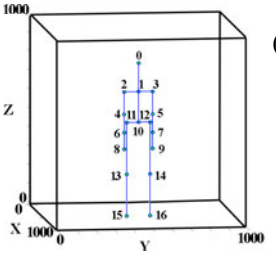
The paper is organized as follows. First, in Section 2, we explain how we estimate 3D human pose from a given video frame. In Section 3, we present how we extract pose vector from 3D human pose, merge pose vectors across successive frames to form a multivariate time series and then classify activities. We describe our experiments in Section 4 and conclude in Section 5.

2 Human Pose Estimation

First we need to estimate 3D human pose from each video frame. We employ a kinematic tree model of human body, consisting of 17 nodes or joints linked by 16 segments. Each joint has 3 DOF and predefined rotational ROM (range of motion). Estimating 3D pose from a 2D input image requires a list of approximate 2D coordinates of the joints as a prerequisite. The list can be obtained by body part detection algorithms. As body part detection itself is a diverse field of research, it will not be elaborated here due to space constraints. Further discussions can be found in [12]. We shall assume that body part detection has already been performed on the video frames, and that we are given an array of approximate 2D coordinates of 17 joints extracted from each frame as shown in Fig 3a. The input to our system is a vector $\mathbf{x} \in \mathbb{R}^{34}$ (Note that some of its components might be undefined for occluded joints). The pose estimation step takes $\mathbf{x}$ and produces a vector $\mathbf{y} \in \mathbb{R}^{51}$ which contains 3D coordinates of 17 joints. Since $\mathbf{x}$ can map to multiple $\mathbf{y}$, the previous output is used to disambiguate the mapping. The output of the estimation step for the i^{th} time step is then defined as $\mathbf{y}_i = f(\mathbf{y}_{i-1}, \mathbf{x})$, where $\mathbf{y}_{i-1}$ is the output of the previous time step. As there is no previous output for the first frame of a sequence, $\mathbf{y}_0$ is estimated through a lookup table which stores $\hat{\mathbf{x}}$-to-$\hat{\mathbf{y}}$ mappings of 50 primitive poses from 13 viewpoints. $\mathbf{y}_0$ is chosen from the lookup table as the value of $\hat{\mathbf{y}}$ in the table corresponding to $\hat{\mathbf{x}}$ that has the shortest Euclidean distance from the input $\mathbf{x}$.

Because of that fact that no two persons are identical in terms of body shape and volume and that skeleton size plays no role in activity recognition, we normalize the 3D human skeleton in a bounding cube (1000 units in each axis) in a right-handed coordinate system as shown in Fig 2a. Each segment has fixed length constraints as given in Fig 2b. Length constraints minimize the influence of inter-person structural differences.

(a)

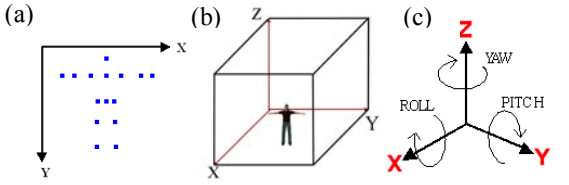


(b)

Segment	Distance	Segment	Distance
1-0	151	7-9	86
1-2	75	10-1	165
1-3	75	10-11	62
2-4	119	10-12	62
3-5	119	12-14	274
4-6	97	11-13	274
5-7	97	13-15	220
6-8	86	14-16	220

Fig. 2. (a) Normalized skeleton in 1000^3 bounding cube (b) Joint constraints table

A camera has 6 DOF with respect to an observed person: translations along x, y, z and rotations about x (roll, φ), y (pitch, θ), z (yaw, ψ). Fig 3c shows rotations about the axes. Because of normalization, translations along the axes have no effect on the system. Hence, our camera parameters are just ψ, θ and φ, each of which ranges from 0º to 360º. We standardize the direction of rotations as clockwise (following left-hand grip rule). Fig 3d shows rotational coordinates of basic viewpoints. Since the order of rotation matters in 3D, we will always follow the yaw-pitch-roll order.



(d)

Viewpoint	ψ	θ	φ
Front	0	0	0
Back	180	0	0
Right	270	0	0
Left	90	0	0
Top	0	90	0

Fig. 3. (a) Axes of the original data from preprocessing (b) Right-handed coordinate system of the 3D model (c) Yaw, pitch and roll (d) Rotational coordinates for various viewpoints

Fig 3a depicts an example of an original list of 2D coordinates of the joints obtained from preprocessing, with a negatively oriented coordinate system. If we assume for the moment that ψ, θ and φ are known, then projection equations given in Table 1 can be used to map **x** to a plane in the bounding cube. After that, we need to estimate 'depth'. For each segment, the ratio between the length of the segment and the average length of all the segments is calculated. The intuition is that if the ratio of a particular segment is smaller than that of the standard model, the segment must have some depth component whose direction is defined by the unit vector of the camera's line of sight (obtainable from ψ, θ and φ). Whether to project the depth component into the positive or negative direction of the unit vector depends upon the 'legality' of the new pose as defined by the joint rotational constraints. If both directions are allowed, multiple outputs will be produced. After all iterations of depth estimation, we normalize each segment's length. We keep the slope of each bone constant and change its length to that of the standard model defined in Fig 2b.

Table 1. Projection equations

	X	Y	Z
Roll (φ)	500	Sin(φ) x Y + Cos(φ) x X	- Cos(φ) Y + Sin(φ) X
Pitch (θ)	Cos(θ) X – Sin(θ) Z	Y	Sin(θ) X + Cos(θ) Z
Yaw (ψ)	Cos(ψ) X + Sin(ψ) Y	Sin(ψ) X + Cos(ψ) Y	Z

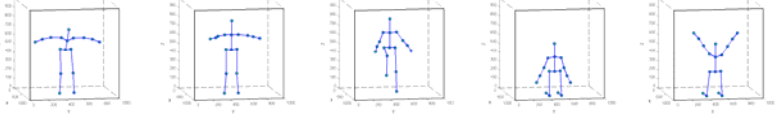


Fig. 4. 3D stick figures corresponding to various poses

After centering the skeleton in the cube with feet in the x-y plane, we get a 3D skeleton as shown in Fig 4. So far, we have assumed that ψ, θ and φ are known. But in reality we do not know the orientation of the camera. However, since inter-frame changes in relative camera position can be very small, ψ, θ and φ will not be so much different from the values of the preceding frame. We perform an exhaustive parallel search over ψ, θ and φ with initial values of the preceding frame bounded by ±45° envelope with 5° discrete step size. The search generates a list of legal 3D poses with ±5° tolerance. The last step of the pose estimation step is to choose one pose **y** from

the list that is most similar to previous output $\mathbf{y}_{i-1}$. Out of all legal poses in the list, we choose the one with shortest Euclidean distance from $\mathbf{y}_{i-1}$.

3 Human Activity Recognition

Activity recognition is the step after pose estimation. As the estimated pose is 3D, the resulting activity recognition system is inherently invariant to viewpoint. First, we extract a relevant *pose vector* from the 3D pose configuration. Each joint has 3 DOF. Fig 5 illustrates the degrees of freedom of right elbow. Note that the third DOF 'roll' or 'twist' is redundant for most joints in the stick figure pose representation. We represent pose vector by the configuration of the 12 joints as shown in Fig 6a. Each joint is represented by 2 angles (yaw and pitch). Therefore, each pose is represented by a 24-dimensional vector **p**. An activity is a sequence of poses. An activity is, therefore, represented by a multivariate time series matrix comprising 24 columns. Fig 6b illustrates one particular column of the matrix for the activity 'jumping'.

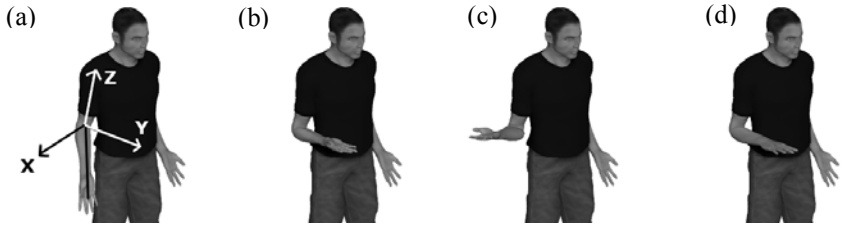


Fig. 5. (a) Original orientation of the forearm (b) Changing pitch angle from 0° to 90° (c) Changing yaw angle to 90° from b (d) Changing roll angle to 180° from b (twisting forearm)

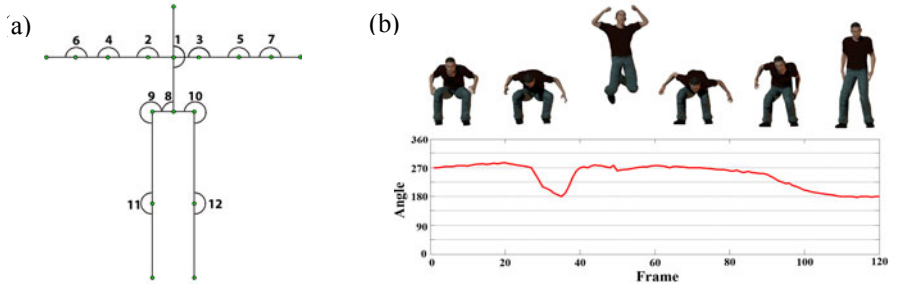


Fig. 6. (a) The 12 joints to present pose vector (b) a univariate time series comprising pitch angle of the right knee during the activity '*jumping*'

We classify activities using Nearest Neighbor Algorithm (NN) with Dynamic Time Warping (DTW) is 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 [13]. 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. Due to space constraints, we will not describe DTW in detail. Interested readers can refer to [14-15]. The only significant drawback of standard DTW is its relatively high computational cost $O(n^2)$ for comparing sequences of length n. However with global constraints (such as Sakoe-Chiba Band[16] and Itakura Parallelogram[17]) and various lower-bounding techniques (such as LB_Keogh [14]), the complexity can be reduced to almost linear time. In [14] , LB_Keogh lower-bounding technique, however, works only for univariate time series. For activity recognition, we extend Keogh's technique to perform lower-bounding of multivariate time series just like in [18]. The proof of the lower-bounding property of multivariate time series is also presented in [18]. 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.

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

where N is the dimension of the multivariate time series. Fig 7 depicts DTW of univariate time series and multivariate time series. The best thing about our algorithm is that N is adjustable based on the availability of joint information. For example, Fig 8

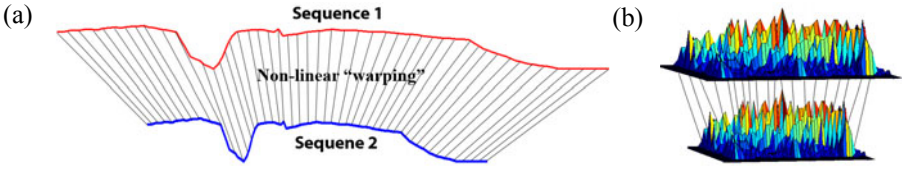


Fig. 7. (a) DTW for univariate time series) (b) DTW for multivariate time series

(b)

```
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;
}
```

Fig. 8. (a) "Jumping jacks" seen from right (b) Pseudo-code of 1-NN DTW sequential search algorithm with lower-bounding

shows part of a right-view image sequence of a person doing "jumping jacks" where left arm and leg are fully occluded. Unlike other activity recognition systems in the literature, our algorithm can leave out all the missing joints and compute d and DTW only based on available information. This makes our system very robust.

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 8b) and outputs the label of the known activity which has the shortest global distance from the unknown time series. The system is scalable and suitable to be employed in domains such as video indexing.

4 Experiments

We carried out two separate experiments to evaluate pose estimation performance and activity recognition performance. For pose estimation, we used two of our own datasets. The first dataset comprised 200 static poses from 5 viewpoints (front, back, left, right and top) generated by *POSER PRO* [19]. The poses were taken from the library that came with the software package. So the first dataset contained 1000 static poses in total. Fig 9 depicts some of the poses from our dataset. The second dataset contained poses synthesized from 2 motion sequences (jumping and walking) with 120 frames for each sequence taken from 5 viewpoints. So the second dataset contained a total of 1200 poses. The ground truths for all the poses were obtained by a *Python script* that translated POSER's coordinate system to ours. For each pose, the ground truth was the 3D coordinates of all the joints. To evaluate activity recognition, we used CMU Motion-Capture database [20]. In fact, there were well-known datasets for viewpoint invariant human activity recognition such as IXMAS dataset [4]. However, since those standard datasets contained no annotated joint information, the CMU dataset (which provides 3D joint information) was our only choice. We selected 10 activities (dribbling, walking, running, jumping, boxing, dancing, waving, sitting,

Fig. 9. Our dataset to test pose estimation

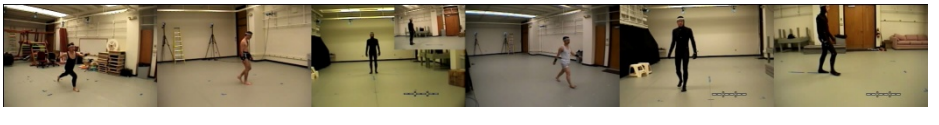


Fig. 10. CMU dataset to test activity recognition

climbing and kicking ball). Each activity was performed 4 times (mostly by the same actor). Fig 10 depicts some of the scenes from the CMU dataset. Their skeleton model was mapped to ours and joints coordinates are converted accordingly.

For pose estimation test, we took the *symmetric mean absolute percentage error* (SMAPE) as an error measure [21].

$$SMAPE = \frac{100}{n} \sum_{i=1}^{n} \left(\frac{||\boldsymbol{p}_i| - |\boldsymbol{t}_i||}{|\boldsymbol{p}_i| + |\boldsymbol{t}_i|} \right) \quad (2)$$

where n is the total number of poses, $\mathbf{p}_i$ is the i^{th} estimated pose vector and $\mathbf{t}_i$ is the i^{th} ground truth pose vector. Note that the default dimension of $\mathbf{p}_i$ and $\mathbf{t}_i$ is 24. But some components of $\mathbf{p}_i$ might be undefined due to occlusion. In those cases, we reduced the dimensionality by neglecting undefined components in $\mathbf{p}_i$ and the corresponding ones in $\mathbf{t}_i$. In activity recognition tests, we performed leave-3-out cross-validation. For each activity, we trained the system on the data from 1 out of 4 clips and then tested the system using the data from the other 3 clips. Since there were 4 combinations of picking one clip for training, the whole process was repeated 4 times and the resulting values were averaged. To test the robustness of our activity recognition system, we tested on various values of N (number of joints in the pose vector). We respectively tested without wrists, without lower body and without arms and shoulders. Table 2 and 3 list our experimental results. The results for Table 2 were calculated using (2) where n is 200 for static poses and 120 for motion sequences, for each viewpoint. Note that Table 2 reports error rates whereas Table 3 reports accuracy rates.

Table 2. Pose estimation results

	static poses	sequences
Front view	14.7%	12.3%
Back view	15.0%	13.8%
Left view	16.5%	13.2%
Right view	16.2%	13.9%
Top view	17.1%	14.1%
Average	15.9%	13.5%

Table 3. Activity recognition results

	Accuracy
N=12	97.5%
N=10 (without wrists [joints 6 and 7]*)	97.5%
N=8 (without lower body [9 to 12])	35.8%
N=6 (without arms & shoulders [2 to 8])	80.0%

*Note: Excluded joints numbers, according to Fig 6

The results of the post estimation tests demonstrate that the proposed system achieves decent performance in pose estimation. The error rates have shown to decrease when estimating poses from motion sequences. Despite one-shot learning, the results of the activity recognition tests demonstrate that our system easily achieves results on-par with current state of the art fixed view methods. The fact that the second test (N=10) gave the same accuracy rate as the first test (N=12) implies that wrist movement is minimal in the dataset and that it is redundant to take wrist configuration. The third test (N=8) gave very low accuracy rate (which was expected) because almost all the activities (especially running and kicking) had the highest variance in lower body configuration. Finally, the last test (N=6) did not produce low accuracy rate because only a few activities had the highest variance in arm configuration.

5 Conclusion and Future Work

We have presented a novel approach to viewpoint invariant human activity recognition system from uncalibrated monocular video source. Our system can learn from a small set of training examples. Our analysis and experiments show that we can indeed achieve viewpoint invariance in human activity recognition with high accuracy. This prototype limits classification of human activities to just 10 classes under a closed world assumption, but there are countless real-world activities. Since our system is scalable and the test results are promising, we could extend further to recognise a variety of common human activities.

As future work, we would first like to select a suitable body part detection algorithm from the literature and plug into our system. We would then obtain a standalone activity recognition system and be able to test our system on a variety of datasets.

References

1. Ji, X., Liu, H.: Advances in View-Invariant Human Motion Analysis: A Review. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews 40(1), 13–24 (2010)
2. Holte, M.B., Moeslund, T.B.: View invariant gesture recognition using 3D motion primitives. Paper Presented at the IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2008 (March 31-April 4 2008)
3. Yung-Tai, H., Jun-Wei, H., Hai-Feng, K., Liao, H.Y.M.: Human Behavior Analysis Using Deformable Triangulations. Paper Presented at the 2005 IEEE 7th Workshop on Multimedia Signal Processing (October 30-November 2, 2005)
4. Weinland, D., Ronfard, R., Boyer, E.: Free viewpoint action recognition using motion history volumes. Comput. Vis. Image Underst. 104(2), 249–257 (2006)
5. Jin, N., Mokhtarian, F.: Image-based shape model for view-invariant human motion recognition. Paper Presented at the IEEE Conference on Advanced Video and Signal Based Surveillance, AVSS 2007 (September 5-7, 2007)
6. Sminchisescu, C.: 3D Human Motion Analysis in Monocular Video Techniques and Challenges. In: Proceedings of the IEEE International Conference on Video and Signal Based Surveillance, p. 76. IEEE Computer Society, Los Alamitos (2006)
7. Souvenir, R., Babbs, J.: Learning the viewpoint manifold for action recognition. Paper Presented at the IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2008 (June 23-28, 2008)
8. Yeyin, Z., Kaiqi, H., Yongzhen, H., Tieniu, T.: View-invariant action recognition using cross ratios across frames. Paper Presented at the 16th IEEE International Conference on Image Processing (ICIP) (November 7-10, 2009)
9. Agarwal, A., Triggs, B.: Recovering 3D human pose from monocular images. IEEE Transactions on Pattern Analysis and Machine Intelligence 28(1), 44–58 (2006)
10. Wei, X.K., Chai, J.: Modeling 3D Human Poses from Uncalibrated Monocular Images. In: 12th IEEE International Conference on Computer Vision, Kyoto, Japan (2009)
11. Shen, Y., Foroosh, H.: View-Invariant Action Recognition from Point Triplets. IEEE Transactions on Pattern Analysis and Machine Intelligence 31(10), 1898–1905 (2009)

12. Lee, M.W., Cohen, I.: Human body tracking with auxiliary measurements. Paper Presented at the AMFG 2003. IEEE International Workshop on Analysis and Modeling of Faces and Gestures (October 17, 2003)
13. Senin, P.: Dynamic Time Warping Algorithm Review, Honolulu, USA (2008)
14. Keogh, E., Ratanamahatana, C.A.: Exact indexing of dynamic time warping. Knowl. Inf. Syst. 7(3), 358–386 (2005)
15. Yi, B.-K., Jagadish, H.V., Faloutsos, C.: Efficient Retrieval of Similar Time Sequences Under Time Warping. In: Proceedings of the Fourteenth International Conference on Data Engineering, pp. 201–208. IEEE Computer Society, Los Alamitos (1998)
16. Sakoe, H., Chiba, S.: Dynamic programming algorithm optimization for spoken word recognition. In: Readings in Speech Recognition, pp. 159–165. Morgan Kaufmann Publishers Inc., San Francisco (1990)
17. Itakura, F.: Minimum prediction residual principle applied to speech recognition. IEEE Transactions on Acoustics, Speech and Signal Processing 23(1), 67–72 (1975)
18. Rath, T.M., Manmatha, R.: Lower-Bounding of Dynamic Time Warping Distances for Multivariate Time Series. University of Massachusetts, Massachusetts (2003)
19. Pose Pro. 2010, Smith Micro (2010)
20. CMU Motion Capture Database, `http://mocap.cs.cmu.edu/`
21. Flores, B.E.: A pragmatic view of accuracy measurement in forecasting. Omega 14(2), 93–98 (1986)