EURASIP Journal on Advances in Signal Processing
a SpringerOpen Journal

RESEARCH Open Access

Human action recognition based on estimated weak poses

Wenjuan Gong*, Jordi Gonzàlez and Francesc Xavier Roca

Abstract

We present a novel method for human action recognition (HAR) based on estimated poses from image sequences. We use 3D human pose data as additional information and propose a compact human pose representation, called a *weak pose*, in a low-dimensional space while still keeping the most discriminative information for a given pose. With predicted poses from image features, we map the problem from image feature space to pose space, where a Bag of Poses (BOP) model is learned for the final goal of HAR. The BOP model is a modified version of the classical bag of words pipeline by building the vocabulary based on the most representative *weak poses* for a given action. Compared with the standard *k*-means clustering, our vocabulary selection criteria is proven to be more efficient and robust against the inherent challenges of action recognition. Moreover, since for action recognition the ordering of the poses is discriminative, the BOP model incorporates temporal information: in essence, groups of consecutive poses are considered together when computing the vocabulary and assignment. We tested our method on two well-known datasets: HumanEva and IXMAS, to demonstrate that *weak poses* aid to improve action recognition accuracies. The proposed method is scene-independent and is comparable with the state-of-art method.

Keywords: Human action recognition, Human pose estimation, Gaussian process regression, Bag of words

Introduction

Human action recognition (HAR) is an important problem in computer vision. Application fields include video surveillance, automatic video indexing and human computer interaction. One can categorize the scenarios found in the literature into several groups: single-human action [1], crowds [2], human-human interaction [3], and action recognition in aerial views [4], to cite but a few. Although the method proposed in this article mainly concentrates on single-HAR, it can be also applied to all the aforementioned scenarios, given that the 2D silhouettes of the agents are able to be extracted from image sequences.

Most solutions for HAR learn action patterns from sequences of image features like Space-Time Interest Points [5,6], temporal templates [7], 3D SIFT [8], optical flow [9,10], Motion History Volume [11], among others. These features are commonly used to describe human actions which are subsequently classified using techniques like Hidden Markov Models [10,12-15], and Support Vector Machines [6]. Recent and exhaustive reviews of methods for HAR can be found in [16,17]. While most of the related work are concentrating on exploring different input features and classification methods, very few of them explores the use of 3D motion capture data for 2D action recognition.

Ning et al. [1] propose a model by adding one hidden layer to conditional random fields (CRF) containing pose information. One of the advantages is that every video frame has an action label, so that action segmentation is integrated with action recognition as a whole. However, the optimal number of consecutive frames which contribute to the decision of the action label of the current frame is given by the model. In our proposal, the optimal frame number is calculated from the training data. Also, while Ning et al. in [1] use CRFs to model relations between image features and action labels, we label motion sequences with a bag of poses (BOP) model, an extension of bag of words (BOW). BOW has been widely applied in classification problem [18-22]. We will show that compared with BOW from only 2D image features, incorporation of *weak poses* combined with BOP

*Correspondence: wenjuan@cvc.uab.es
Computer Vision Center & Universitat Autònoma de Barcelona, Building O, UAB Campus, Barcelona, Spain

improves action recognition accuracy. The average action recognition accuracy of the proposed method is better than that in [1].

In this article, our main hypothesis is that estimating 3D poses from 2D silhouettes can be advantageous for action recognition. A challenge of this solution is the inherent ambiguities between 2D image features and 3D poses. Some researchers use multiple-view videos [23-25], although single-view image sequences are more generic and easy to acquire. Moreover, recent work shows that even in monocular image sequences, reconstruction ambiguity can be tackled using regression methods like relevance vector machine (RVM) [26]. RVM is a special case of Gaussian Process Regression (GPR) [27]: while RVM considers the most representative training samples (thus being fast in the learning step), GPR takes all the training samples thus being a more accurate regression technique. For this reason, GPR has been successfully used for modeling the mapping between 2D image features and 3D human poses [28,29].

Inspired by these works, the whole procedure presented in this article is shown in Figures 1 and 2. In essence the method is composed of two steps: training and prediction. In training, a set of Gaussian processes (first row Figure 1) and the BOP model (second row Figure 1) are learnt. On one hand, Gaussian processes are trained with pairs of 2D image features and our intermediate 3D pose representation or *weak poses*. For each dimension of the *weak pose* parameter space, we define a Gaussian process to map from 2D image features to this particular dimension. On the other hand, the BOP model is trained with *weak poses* and motion sequences. We introduce temporal information in BOW by grouping consecutive video frames. Similar to graphical models which account for the influence of neighboring data, in our case we take into account those neighboring frames by merging consecutive frames in a single word. After choosing the most representative *weak poses* for the vocabulary, each motion sequence is represented as a histogram and SVMs are finally trained. In the prediction step, given an unknown video sequence, we predict human poses with the trained set of Gaussian processes, and represent the video sequence using the histogram of the vocabulary. After that, we label the action by the trained SVMs.

The rest of the article is organized as follows: next section introduces our human body model and human posture representation; Section *Weak pose* estimation using GPR describes how we use a set of Gaussian processes for learning the mapping from 2D image features to 3D human poses; in Section BOP for action recognition, we describe a procedure for incorporating temporal information in a BOW schema, showing the results in Section Experimental results. Finally Section Conclusions and discussion presents the future avenues of research.

Data representation

The flexibility of the human body and the variability of human actions produce high-dimensional motion data.

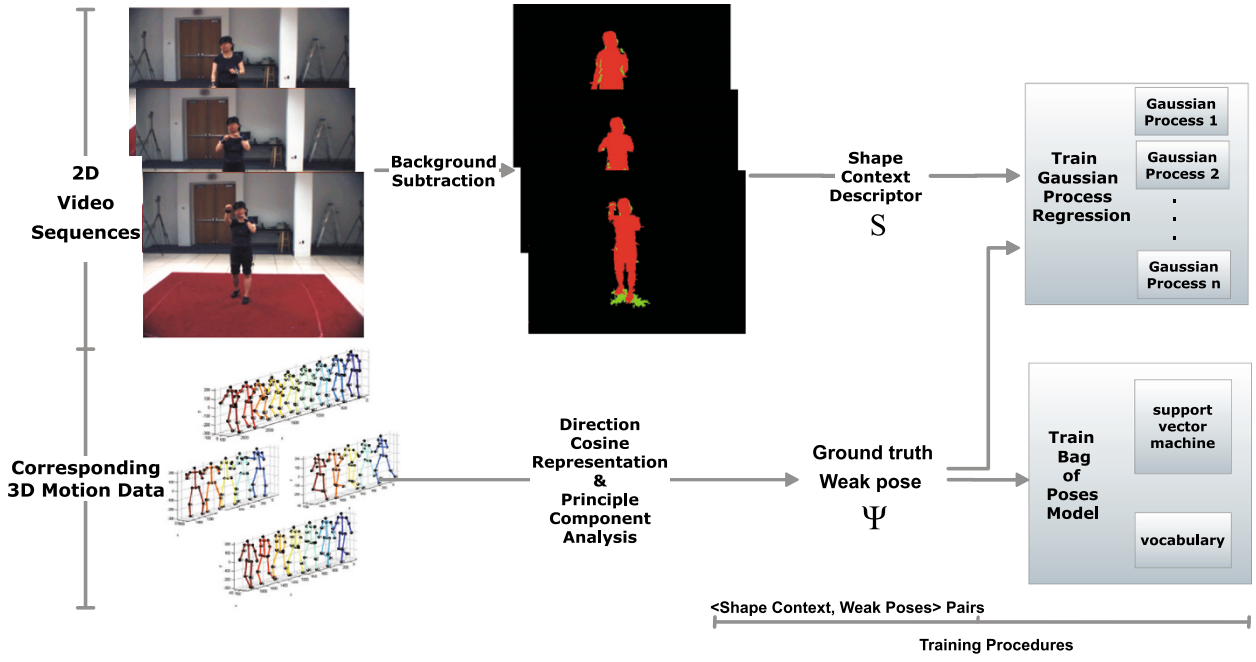

Figure 1 Learning step: we train Gaussian processes to learn the regression function from shape context descriptors (SCDs) to *weak poses*. In parallel, a BOP model is built for each action class by extracting key poses and training SVM classifiers.

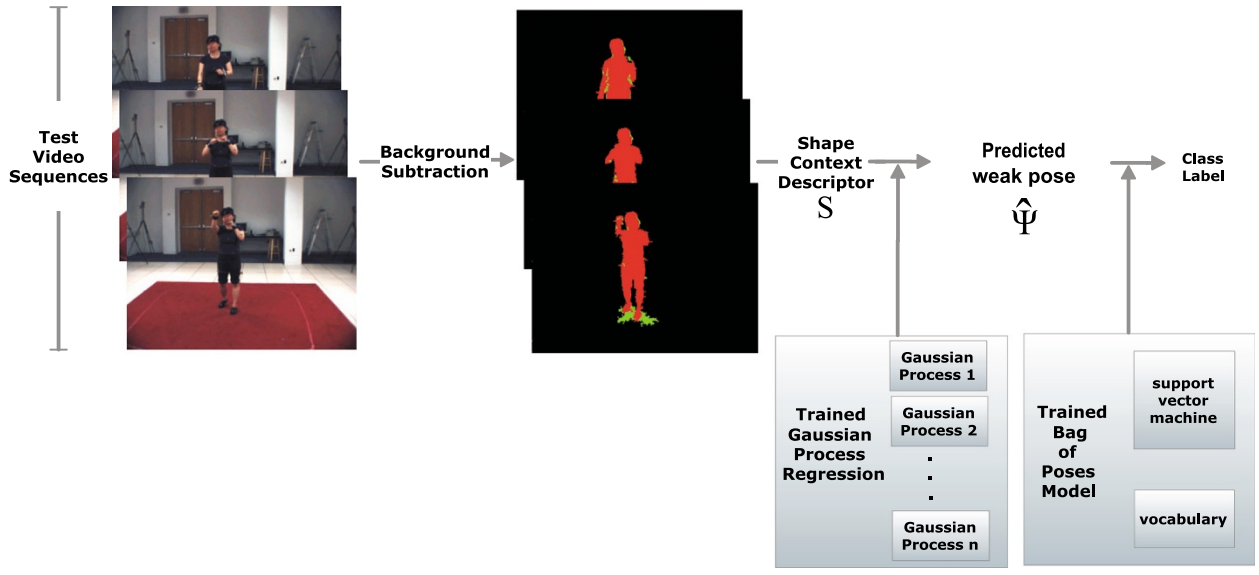

Figure 2 Predicting phase. The test video sequence is described using shape context descriptors as in the learning phase (see Figure 1). *Weak poses* are predicted from shape context descriptors using trained Gaussian processes and the video is represented as a histogram of the vocabulary learned in the training phase. The video is finally labeled using the ensemble of trained SVMs for each action class.

Given a number of video sequences of a single actor executing certain actions, in training each image has its corresponding 3D motion capture data. How to represent these data in a compact and effective way is also a challenge.

We select a compact representation of human postures in 3D, in our case a stick figure of 12 limbs. For representing 3D motion data, a human pose is defined using twelve rigid body parts: hip, torso, shoulder, neck, two thighs, two legs, two arms and two forearms. These parts are connected by a total of ten inner joints, as shown in Figure 3a. Body segments are structured in a hierarchical manner, constituting a kinematic tree rooted at the hip, which determines the global rotation of the whole body.

Some works represent human poses with 3D joint position, others have explored representing limb orientation with polar angles or direction cosines (DCs). In the latter case, the orientation of each limb is represented by three DCs of the angles formed by the limb in the world coordinate system. DCs embed a number of useful invariants, and by using them we can eliminate the influence of different limb lengths. Compared to Euler angles, DCs do not lead to angle discontinuities in temporal sequences. Lastly, DCs have a direct geometric interpretation which is an advantage over quaternions [30].

So we use the same representations for human postures and human motions as in [31]: a limb orientation is represented using three parameters, without modeling self rotation of the limb around its axes, as shown in Figure 3b. This results in a 36-D representation of the pose of the actor in frame j of video i:

$$\psi_j^i = [\cos\theta_1^x, \cos\theta_1^y, \cos\theta_1^z, \ldots, \cos\theta_{12}^x, \cos\theta_{12}^y, \cos\theta_{12}^z], \quad (1)$$

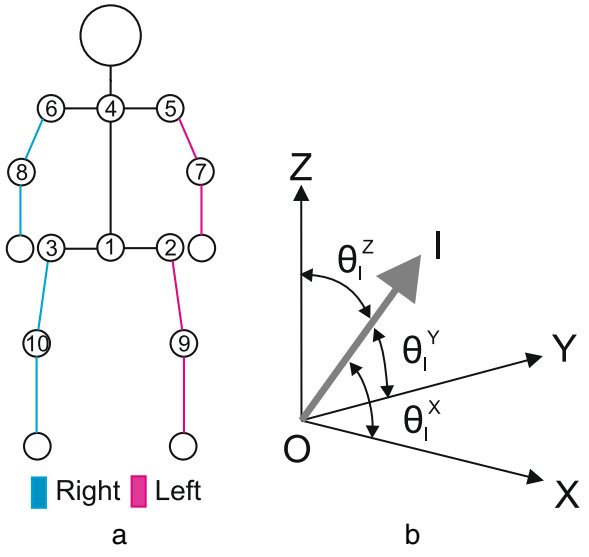

Figure 3 The 3D stick figure model used for representing human pose and limb orientation represented as direction cosine. (**a**) Ten principal joints corresponding to the markers used in motion capture are used for 3D stick figure. **(b)** Limb orientation is represented with the direction cosines of angles $(\theta_l^x, \theta_l^y, \theta_l^z)$ between the limb l and the axes.

where θ_l^x, θ_l^y and θ_l^z are the angles between the limb l and the axes as shown in Figure 3b.

With DCs, we represent the motion sequence of the i-th video as a sequence of poses:

$$\Psi_o^i = [\psi_1^i, \psi_2^i, \ldots, \psi_{n_i}^i], \tag{2}$$

where n_i is number of poses (frames) extracted from video i.

Universal action space (*UaSpace*)

Since natural constraints of human body motions lead to highly correlated data [32], we build a more compact, non-redundant representation of human pose by applying principle component analysis (PCA) to all actions. This universal action space (*UaSpace*) will become the basis for vocabulary selection and finally classification using BOP.

By projecting human postures into the *UaSpace*, distances between poses of different actions can be computed and used for classification. Figure 4 shows pose variation corresponding to the top (in terms of eigenvalues) nine eigenvectors in the *UaSpace*. From the figure, one can see which pose variations each eigenvector accounts for in the eigenspace decomposition. For example, one can see that the first eigenvector corresponds to the characteristic motion of the arms and the second eigenvector corresponds to the motion of the torso and the legs. In the following section, we describe how *weak poses* are estimated from video frame feature descriptors using GPR.

We denote the pose representation in the reduced dimensionality space as *weak poses* or ψ', and the motion sequence of *UaSpace* the i-th video is represented as:

$$\Psi^i = [\psi_1^{i\prime}, \psi_2^{i\prime}, \ldots, \psi_{n_i}^{i\prime}], \tag{3}$$

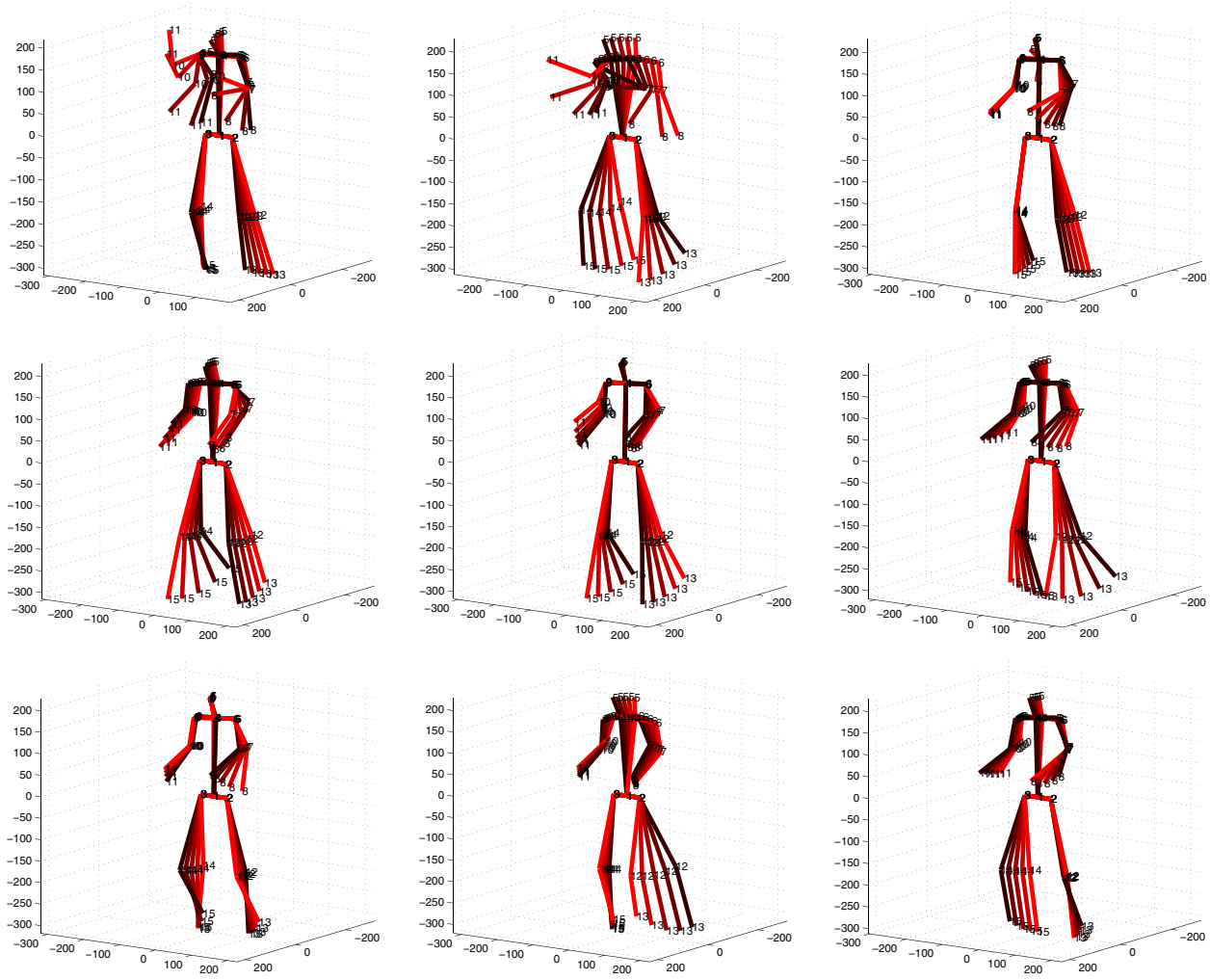

Figure 4 Visualizing the nine principal variations of the pose within *UaSpace* learnt from HumanEva data. Each plotted stick figure is a re-projected pose by moving it in one eigenvector's dimension from −3 up to 3 times the standard deviation.

where $\psi_j^{i'}$ is the *weak pose* corresponding to the *j*-th image frame in *i*-th video sequence.

Weak pose estimation using GPR

We use SCD to represent the human silhouette found using background subtraction [33]. Shape context is commonly applied to describe shapes given silhouettes [34,35], and have been proven that it is an effective descriptor for human pose estimation [36].

The main idea of our SCD is to place a sampled point on a shape in the origin of a radial coordinate system and then to divide this space into different range of radius and angle. In this way, the number of points that fall in each bin of the radial coordinate system are counted and encoded into a bin of an histogram. In our experiments, we place the origin of radial coordination on the centroid of a silhouette and divide radius into five bins equally spaced and divide angle into 12 equally spaced bins, as shown in Figure 5. As a result, the SCD vector is 60-D. Figure 6 shows examples of extracted silhouettes of actor "S1" performing action "Box" and action "Gesture". From the figure, we can see that background subtraction with the method in [33] gives promising background results. Although there are variances of centroid positions among similar silhouettes, from observations, we can say that centroids are still reliable. We set the centroid of the silhouette as the center of the local coordinate system, and the largest diameter is set as 1.25 times the diagonal length of the silhouette bounding box.

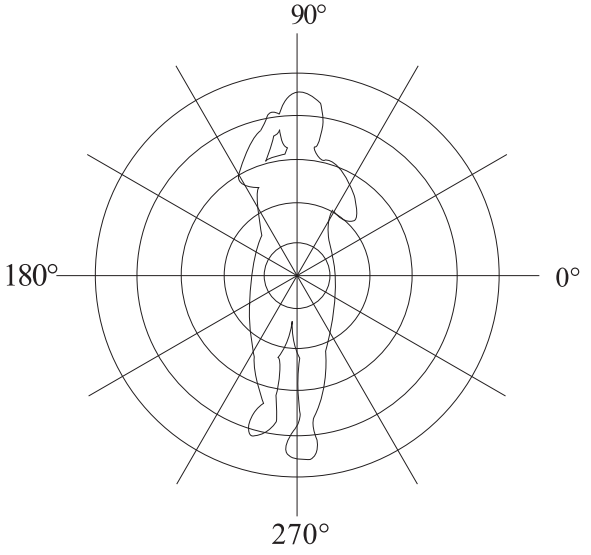

Figure 5 Radial coordinates for SCD. The origin of the polar coordinate system is placed on the centroid of the bounding box of the silhouette. The radius is divided equally into 5 bins and the circle is divided equally into 12 bins.

The normalization of the resulting SCD has a significant impact on the performance of GPR. We exploit two different ways of normalizing data: standard deviation and individual normalizations. Suppose $\mathbf{s}_{orig}$ denotes the original SCD from one image, and

$$\mathbf{s}_{\text{orig}} = [\, np^1, np^2, \ldots, np^i, \ldots, np^{60}\,], \tag{4}$$

where np^i is the number of pixels that fell in the *i*-th bin.

In standard deviation based normalization, we calculate standard deviations from all training SCDs $\mathbf{std} = [\,\text{std}^1, \text{std}^2, \ldots, \text{std}^{60}]$. Then we normalize each dimension of the SCD by dividing it with the corresponding standard deviation. Then the normalized SCD can be represented as:

$$\mathbf{s}_{\text{norm1}} = \left[\frac{np^1}{\text{std}^1}, \frac{np^2}{\text{std}^2}, \ldots, \frac{np^i}{\text{std}^i}, \ldots, \frac{np^{60}}{\text{std}^{60}}\right] \tag{5}$$

In individually normalizing method, we divide the pixel number in a bin by the total pixel number of the SCD. That is, if we represent the total number of pixels in one SCD as *npSum*, then in individually normalizing method, the normalized SCD is defined as:

$$\mathbf{s}_{\text{norm2}} = \left[\frac{np^1}{npSum}, \frac{np^2}{npSum}, \ldots, \frac{np^i}{npSum}, \ldots, \frac{np^{60}}{npSum}\right]. \tag{6}$$

We compare these two different ways of normalizing SCDs in experimental results.

Gaussian process regression

The problem of predicting 3D human postures from 2D silhouettes is highly non-linear. Gaussian processes have been effectively applied for modeling non-linear dynamics [37-39]. For example, Gaussian process has been applied to non-linear regression problems, like robot inverse dynamics [40] and nonrigid shape recovery [41].

With the method described in the above section, we extract human silhouettes from training video sequences and describe them with normalized SCD.

$$\mathbf{S} = [\,\mathbf{s}^1, \mathbf{s}^2, \ldots, \mathbf{s}^p\,], \tag{7}$$

where $\mathbf{s}^i$ is the vector of SCD extracted from the *i*-th training video sequence. The method described in [26] predicts 3D poses from 2D image features using RVM. RVM is more efficient during learning, but less accurate since RVM is a special case of GPR: during the learning phase, RVM takes the most representative training samples while GPR takes all training samples. Additionally, GPR has been successfully applied to pose estimation and tracking problems, for example [28,29]. So in our approach, we will

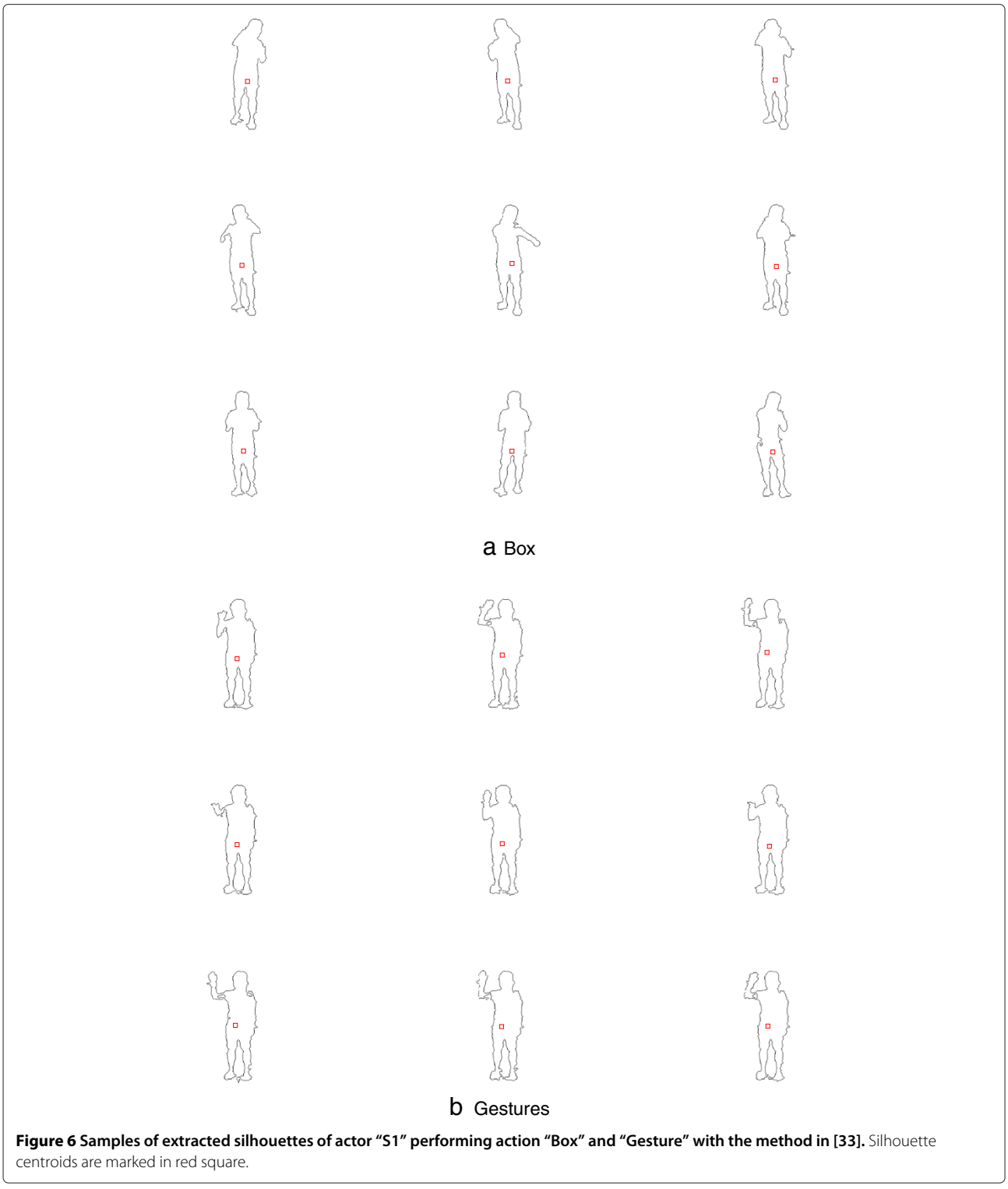

Figure 6 Samples of extracted silhouettes of actor "S1" performing action "Box" and "Gesture" with the method in [33]. Silhouette centroids are marked in red square.

use GPR for modeling the mapping between silhouettes and *weak poses*.

According to Rasmussen and Williams [27], Gaussian process is defined as: *a collection of random variables, any finite number of which have (consistent) joint Gaussian distribution*. A Gaussian process is completely specified by its mean function and a covariance function. Integrating with our problem, we denote the mean function as $m(\mathbf{s})$ and the covariance function as $k(\mathbf{s}, \mathbf{s}')$, so a Gaussian process is represented as:

$$\zeta(\mathbf{s}) \sim \mathcal{GP}_j(m(\mathbf{s}), k(\mathbf{s}, \mathbf{s}')), \tag{8}$$

where

$$\begin{aligned} m(\mathbf{s}) &= E[\,\zeta(\mathbf{s})\,], \\ k(\mathbf{s}, \mathbf{s}') &= E[\,(\zeta(\mathbf{s}) - m(\mathbf{s}))(\zeta(\mathbf{s}') - m(\mathbf{s}'))\,], \end{aligned} \tag{9}$$

We set a zero-mean Gaussian process whose covariance is a squared exponential function with two hyperparameters controlling the amplitude θ_1 and characteristic length-scale θ_2:

$$k_1(\mathbf{s}, \mathbf{s}') = \theta_1^2 \exp\left(-\frac{(\mathbf{s} - \mathbf{s}')^2}{2\theta_2^2}\right). \tag{10}$$

We assume prediction noise as a Gaussian distribution and formulate finding the optimal hyperparameters as an optimization problem. We seek the optimal solution of hyperparameters by maximizing the log marginal likelihood (see [27] for details):

$$\log p(\Psi'|\mathbf{s}, \theta) = -\frac{1}{2}\Psi'^T K_{\Psi'}^{-1}\Psi' - \frac{1}{2}\log|K_{\Psi'}| - \frac{n}{2}\log 2\pi, \tag{11}$$

where $K_{\Psi'}$ is the calculated covariance matrix of the target vector (vector of training *weak poses* in *UaSpace*) Ψ' under the kernel defined in Equation (9).

With the optimal hyperparameters, the prediction distribution is represented as:

$$\begin{aligned} \Psi'^*|\mathbf{s}^*, \mathbf{s}, \Psi' \sim \mathcal{N}(&\mathbf{k}(s^*, \mathbf{s})^T [K + \sigma_{\text{noise}}^2 I]^{-1} \Psi', k(s^*, s^*) \\ &+ \sigma_{\text{noise}}^2 - \mathbf{k}(s^*, \mathbf{s})^T [K + \sigma_{\text{noise}}^2 I]^{-1} \mathbf{k}(s^*, \mathbf{s})), \end{aligned} \tag{12}$$

where K is the calculated covariance matrix from training 2D image features $\mathbf{s}$ and σ_{noise} is the covariance of Gaussian noise. We train a set of Gaussian processes to learn regression from SCD to each dimension of the *weak poses* separately.

BOP for action recognition

Given a test video sequence, we extract SCDs from image sequences and then predict the *weak pose* by the set of trained Gaussian processes. With the predicted *weak poses*, the problem turns into a classification problem in the *UaSpace*.

Inspired by BOW [18-20], we apply the following steps for action recognition: compute descriptors for input data; compute representative *weak poses* to form vocabulary; quantize descriptors into representative *weak poses* and represent input data as histograms over the vocabulary, a BOP representation. Next we explain how to compute the vocabulary and perform classification with our modified BOP model.

Vocabulary selection

The classic BOW pipeline uses k-means for calculating the vocabulary. But this way of calculating the vocabulary does not give promising action recognition results [42]. While energy-based method proposed in [42] gives comparatively better results when applied for each action separately, it is not applicable here. Because the number of key poses calculated from energy-based method is closely related with numbers of motion cycles. When we use one vocabulary for all actions, key pose numbers increases dramatically. While the number of training sequences stays the same. Even we use techniques to create new training sequences, the experiment results are not ideal.

We combine these two methods and propose a new method for computing the vocabulary. First, we select candidate key *weak poses* using energy optimization as in [42]. The key *weak poses* are pre-selected as:

$$F_{\text{pre}}^i = \{f_1^i, f_2^i, \ldots, f_l^i\}, \tag{13}$$

where f_j^i corresponds to local maximum or local minimum energies in i-th motion sequence. And l is the total number of local maximum and local minimum values. Note, l is not a fixed value, and it depends on number of motion cycles and motion variations in the sequence.

Without taking into account temporal information, we cluster all preselected key *weak poses* from all performances: $F_{pre} = \{F_{\text{pre}}^1, F_{\text{pre}}^2, \ldots, F_{\text{pre}}^p\}$, where F_{pre}^i is calculated as in Equation (13) and p is the number of training motion sequences. Then, we select k most representatives *weak poses* F_k from F_{pre} with k-means. So F_k makes the vocabulary. We call the proposed method as energy-k-means. We will show in experiment section comparisons between the energy-k-means, k-means and energy-based method.

To incorporate temporal information into our solution, we consider d consecutive frames as one unit. That is, key *weak poses* with temporal information are preselected as

$$F_{\text{pre}}^t = \{F_{\text{pre}}^{t1}, F_{\text{pre}}^{t2}, \ldots, F_{\text{pre}}^{tl}\}, \tag{14}$$

where

$$F_{\text{pre}}^{tj} = [f_j^{\text{frm}-d+1}, f_j^{\text{frm}-d+2}, \ldots, f_j^{\text{frm}}] \tag{15}$$

is the j-th candidate for key *weak poses*. F_{pre}^{tj} is a concatenation of d consecutive *weak poses* and f_j^{frm} corresponds to local maximum or local minimum energies in j-th

motion sequence, and *tl* equals the total number of preselected key *weak poses*. Then, the vocabulary is calculated as k-means clustering centers F_k^t from F_{pre}^t.

Temporal step d is a critical factor. Experimental results show that, for *weak poses*, after temporal step d reaches a certain value, classification results remain comparatively steady. In Section Temporal step size, we will show how we fix d using cross validation on training data.

Action classification

A vocabulary is calculated as a collection of characteristic key *weak poses*. Then we represent our motion sequences statistically as occurrences of these characteristic key *weak poses*, that is, histograms over the vocabulary. To be specific, the i-th motion sequence Ψ^i represented as in Equation (3) in *UaSpace* can be represented statistically as:

$$\text{hist}^i = [n_1, n_2, \ldots, n_j, \ldots, n_{tk}], \tag{16}$$

where n_j is the number of *weak poses* in Ψ^i that are nearest (Euclidean distance) to j-th word in vocabulary F_k. To incorporate temporal information, we start from d-th frame of video sequence V^i, and compare a concatenation of consecutive d *weak poses* with each entry of the vocabulary F_k^t. And tk in Equation (16) is the number of words contained in vocabulary F_k^t.

For each action, we train a SVM with histograms and their corresponding action class labels. We choose a linear kernel according to experimental results and use cross validation to fix the cost value as 5. For measuring classification results, we use classification accuracy:

$$\text{accuracy} = \frac{tp + tn}{tp + tn + fp + fn}, \tag{17}$$

where tp, tn, fp, fn refer to true positive, true negative, false positive and false negative, respectively. $tp + tn$ represents correctly classified samples, and $tp + tn + fp + fn$ is the total number of all samples. We use this criterion as the maximizing target when we do cross validation to fix parameters, for example, number of Gaussian process m and temporal step size d.

Experimental results

To verify robustness of our method, we choose two public datasets: HumanEva and IXMAS. Ning et al. [1] gives state of art action classification accuracy for HumanEva dataset. We will compare with this result with our experiments on this dataset. There are several related works on action recognition with IXMAS dataset, for example [23-25,43]. Gu et al. [44] listed all state of art experimental results on this dataset. Among all, we will compare with experimental results in [43], because this method uses single viewpoint as input like our method while other methods need multiple viewpoints.

The composition of the data are:

1. HumanEva[a] dataset [45]. This dataset contains six actions: "Walking", "Jog", "Gesture", "Throw/Catch", "Box", and "Combo". We consider the first five actions, since "Combo" is a combination of "Walking", "Jog", and "Balancing on each of two feet". Four actors perform all actions a total of three times each. Trial 1 has both video sequences and 3D motion data; in trial 2, 3D motion data are withheld for testing purposes; trial 3 contains only 3D motion data.
2. IXMAS[b] dataset. We further apply trained models from HumanEva dataset to IXMAS dataset, to test robustness of our method. From this dataset, we take four actions: "Walk", "Wave", "Punch" and "Throw A Ball". They correspond to actions "Walking", "Gesture", "Box" and "Throw/Catch" in HumanEva dataset.

We take only the frontal view from the two dataset. Note that positions of vision cameras in these two dataset of frontal view are not set exactly the same.

Model training

In our experiments, we take the first half of each performance for training $< \mathbf{S}, \Psi >$ and the second half for validation $< \mathbf{S}_{Val}, \Psi_{Val} >$ and use cross validations to fix model parameters like number of Gaussian processes, vocabulary size, temporal step sizes and so on.

Energy-k-means method for vocabulary computation

In this section, we compare the proposed energy-k-means method with the traditional k-means and the energy-based method proposed in [42].

Table 1 Comparisons of classification accuracy (%) among energy-*k*-means method, *k*-means method and energy-based method in [42]

Methods			Number of GPs			
			3	6	10	20
Energy-k-means	Voc size	5	73.9	86.8	86.3	86.1
		10	67.7	83.6	82.9	84.4
		15	64.1	83.9	82.6	85.4
		20	64.7	79.0	77.5	78.4
k-means	Voc size	5	67.2	65.7	58.6	57.9
		10	52.9	68.6	67.9	66.4
		15	60.7	51.4	62.9	67.9
		20	52.2	48.6	55	64.3
Energy-based			35.7	39.3	64.3	64.3

Table 2 Vocabulary size calculated with energy-based method with different numbers of Gaussian processes

	Number of GPs			
	3	**6**	**10**	**20**
Voc size	608	602	639	641

Table 1 shows that the proposed energy-k-means method outperforms the k-means and the energy-based method in all experiment configurations. While for the k-means and the energy-based method, proper parameter settings are needed for better results. For example, with 10 Gaussian processes, the k-means outperforms the energy-based method when the vocabulary size equals 10, while the energy-based method performs better when the vocabulary size equals 5, 10 and 20. The reason that the energy-based method does not give promising results is big vocabulary size, see Table 2. Although we synthesize training data, still the number of training sequences is not enough for this vocabulary size.

Number of Gaussian processes

We train a set of Gaussian processes to learn mappings between SCDs and *weak poses* in *UaSpace* with the training data $< \mathbf{S}, \Psi >$. We calculate pose estimation errors between estimated *weak poses* $\hat{\Psi}$ and the ground truth *weak poses* Ψ' as:

$$\varepsilon = \frac{1}{N} \sum_{p=1}^{P} \sum_{f=1}^{F_p} \|\hat{\psi} - \psi'\|^2, \tag{18}$$

where N is the total number of frames used for training, P is the total number of training performances and F_p is frame numbers of the p-th training performance. To discard missing human detection, we first calculate the energy of SCD for each training frame and filter the training sequences based on calculated energies by keeping 90% of the energies over all frames. This effectively eliminates frames containing catastrophic silhouette extraction failures.

In our experiments, we evaluate different numbers of Gaussian processes (recall that we use one Gaussian process for each dimension in our *weak pose* space). From Table 3, we observe that with fewer than 20 Gaussian processes, increasing the number of Gaussian processes results in noticeable increases in classification accuracy and also decreases in pose estimation error. Our explanation for this is: a small numbers of Gaussian processes are not able to capture or describe all the motion possibilities for actions, which results in predictions that are not accurate. After 20 Gaussian processes, increasing number of Gaussian processes does not result in notable increases in classification accuracy or decreases in pose estimation error. So the best trade-off between accuracy and model complexity is found with 20 Gaussian processes with a vocabulary size of 10. The subsequent experiments are computed with these optimal settings.

Temporal step size

We also use cross validation to get optimal temporal step size d. We add Gaussian noise of different scales to the original 3D marker positions to test the robustness of the prosed method. We run each noise scale five times and calculate average accuracy for all noise scales. Experiment results are shown in Figure 7. This figure shows relations between numbers of temporal steps, numbers of key poses and action recognition accuracies. From the figure, we can see that the size of temporal steps has more influences than the number of key poses (vocabulary size). And after the size of temporal steps reaches 13, classification accuracy becomes rather stable. This implies that the decisive factor in action recognition comes from the continuous motion. Motion elements of short duration is more representative for an action than the overall distribution of important poses. Later on, we fix temporal step size as 13 for the rest of our experiments.

The effect of *weak poses*

To verify the effect of the incorporation of *weak poses*. We use only image features as input for modified BOW with

Table 3 Comparison of classification accuracy (%) and *weak pose* reconstruction error with different numbers of Gaussian processes and different vocabulary size

		Number ofGPs						
		3	**6**	**10**	**15**	**20**	**25**	**30**
Voc size	5	73.9	86.8	86.3	86.0	86.1	86.1	85.6
	10	67.7	83.6	82.9	83.0	84.4	84.2	84.2
	15	64.1	83.9	82.6	80.8	85.4	83.9	83.7
	20	64.7	79.0	77.5	79.7	78.4	84.2	82.2
Mean error	0.399	0.304	0.241	0.200	0.169	0.146	0.127	

Reconstruction error is the difference between predicted *weak poses* and ground truth *weak poses*.

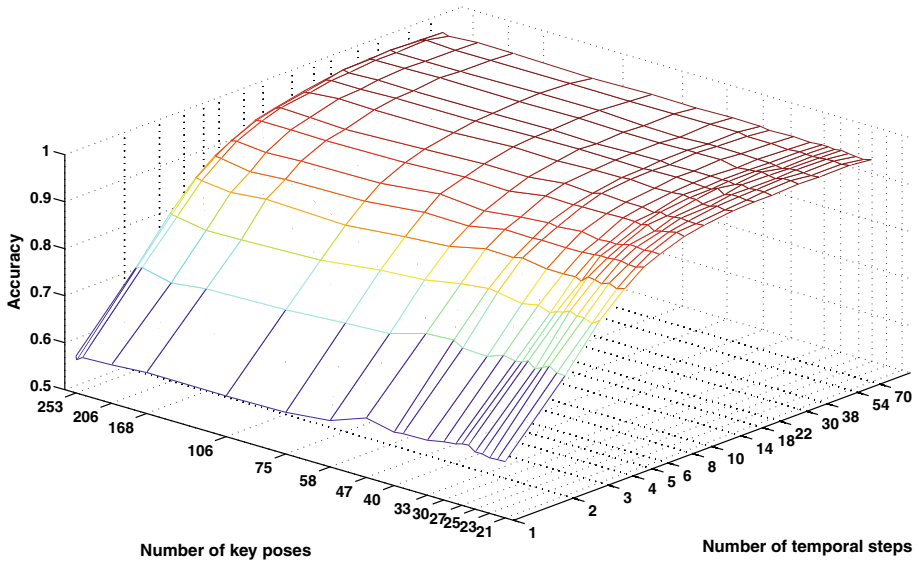

Figure 7 The relations between number of temporal steps, number of key poses and action recognition accuracy.

the optimum parameter settings. That is, we use energy-*k*-means for vocabulary selection and set vocabulary size of 10. Cost of support vector machine is as 5 and temporal step size is as 13. But instead of in *UaSpace*, vocabularies and histograms are calculated in 2D image feature space. Action recognition accuracy with only image features on the validation set is 80.0%, while the action recognition accuracy for the proposed method is 84.4% (see Table 3).

Action recognition accuracy

We utilize a BOP model in classifying actions, as described in Section BOP for action recognition. A set of Gaussian processes and a BOP model are trained on all training data including training and validation data. With the trained models, we evaluate our method on the test data from both HumanEva and IXMAS datasets.

As we take the whole performance as one training example, we have an acute lack of training data. We address this problem by synthesizing training data like [46]. We first split training performances into sub-performances. Then, we translate sub-performances with *trans* times the maximum difference of the training data, where

$$\text{trans} = \{-0.20, -0.15, -0.10, -0.05, 0.05, 0.10, 0.15, 0.20\}, \tag{19}$$

and scale sub-performances by

$$\text{scale} = \{0.80, 0.85, 0.90, 0.95, 1.05, 1.10, 1.15, 1.20\}. \tag{20}$$

We also split and translate test performances into sub-performances. The procedure is the same as for training date. Experimental results for HumanEva dataset are shown in Table 4. The method from [1] shows upper bound accuracy for initialized latent pose conditional random field model ($\text{LPCRF}_{\text{init}}$ in [1]) with the same training and test data.

In our experiments, normalization of input data is a very important step for GPR to make good predictions. So we experimented with two different ways of normalizing

Table 4 Comparison of action recognition accuracy (%) in HumanEva between our methods and the method presented in [1]

Acc.	Box	Jog	Gest	Walk	T/C	All − T/C	All + T/C
[1]	98.9	99.0	63.7	99.6	*no*	90.3	No
Std-norm	88.4	75.1	87.6	91.0	80.0	85.5	84.4
Ind-norm	97.1	91.8	91.9	94.6	80.0	**93.9**	**91.1**

Classification accuracy is defined as correctly labeled samples over total number of samples (refer to Equation (17)). "Std-norm" and "Ind-norm" refer to standard deviation normalizing method and individually normalizing method (refer to Section *Weak pose* estimation using GPR). The column "$All - T/C$" shows the average classification accuracy for all actions excluding "Throw/Catch" and and the column "$All + T/C$" including "Throw/Catch". Bold values show the best results of action recognition accuracies averaged over all actions.

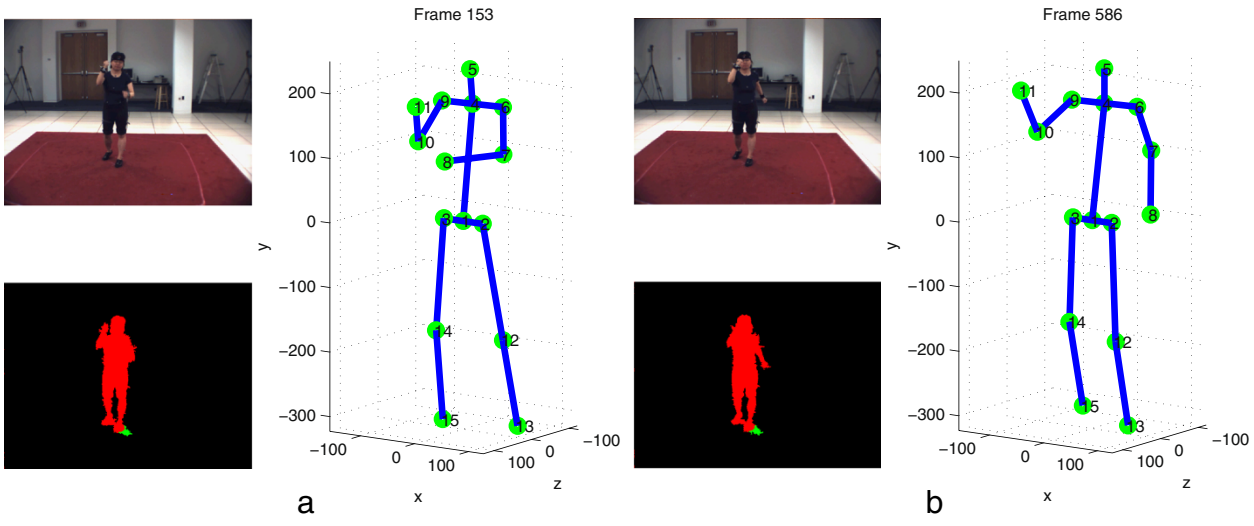

Figure 8 Two exampled frames of good estimation of *weak poses* in HumanEva dataset. *Weak poses* are back-projected from *UaSpace* to the original parameter space and visualized as human poses.

data: standard-deviation based and individual normalizations. Our method with individual normalization has better average classification accuracy than the approach presented in [1].

Due to illumination changes and errors from background subtraction, human silhouettes from every image frame have variant qualities. As a result, the total pixel numbers vary from one frame to another. Individually normalizing method eliminates these differences. So that, later histograms are computed on the same basis. On the contrary, standard deviation based normalization are more suitable to cases while different dimensions from image features have different range of variations. In this case, different dimensions are separately normalized. In later experiments, we fix our normalization as individual normalization.

From experimental results, we observe that for "Throw/Catch" action, in both normalization strategies, classification accuracy are not as satisfactory as other actions. One possible reason for this is the limited

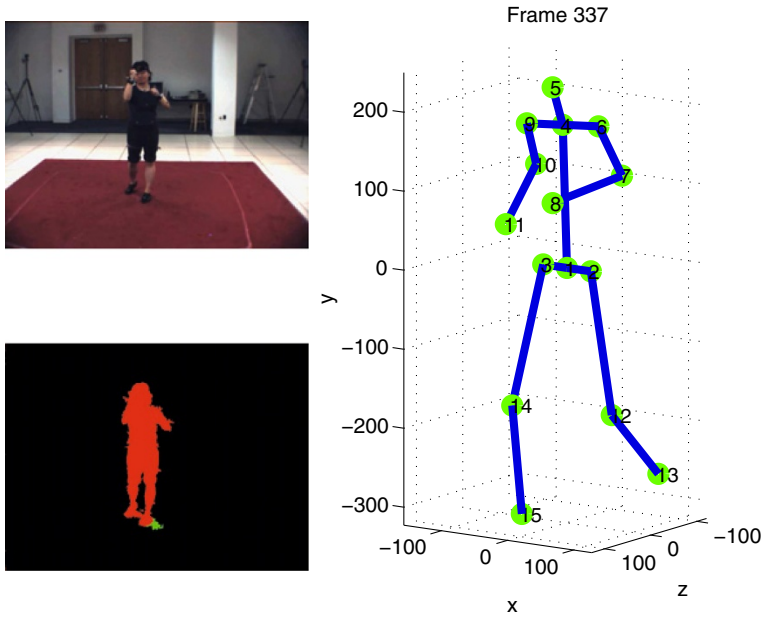

Figure 9 An example of bad estimation of a *weak pose* in HumanEva dataset.

Table 5 Action recognition accuracy (%) of our individually normalizing method for IXMAS dataset using the models learnt from HumanEva dataset compared with the method prosed in [43]

Accuracy	Punch	Wave	Throw a ball	Walk	All actions
Ind-normal	75.0	79.2	75	87.5	79.2
[43]	86.8	79.9	82.4	79.7	82.2

number of training samples for this action. We are using PCA in reducing representation dimensionality. In this case, if training examples for an action are too few, the variations of this action would not be able to be captured by the main eigenvectors. As a result, action recognition accuracy is not as good as other classes. Another observation is, for "Jog" and "Box", individual normalization has a much better performance than the standard-deviation based one. Our explanation for this is, "Jog" and "Box" have more variate poses compared with "Gesture" (the lower body parts of the performer are relatively stable), "Throw/Catch" (the lower body parts are also relatively stable) and "Walking" (the movements of body parts are not as fierce as in "Jog" and "Box"). As a result, when we normalize all training data together, these action classes are more likely to be influenced. While individual normalization keeps variate information of the SCD from each image frame.

To visualize results of *weak pose* reconstruction, we project weak poses from *UaSpace* back to the original parameter space. Figures 8 and 9 show some examples of estimated *weak poses*. We can see that in Figure 8, pose estimation results are satisfactory. In Figure 9, there is a difference between the estimation and the ground truth. Since our ultimate goal is action recognition but not pose estimation, we will not concentrate on further improvements on pose estimation. This pose estimation precision give promising action recognition accuracies.

We run the experiments on a personal computer with four 3.19 Hz processors, and 12 GB memory. Most of the time, the usage of CPU is around 30%, that is, the power of a single core. The time cost for training one Gaussian process is 6.5 h, and predicting one dimension is 3.1 min. And the time cost for calculating the vocabulary is 0.2 s.

We further test our action model (trained with HumanEva data) on IXMAS dataset and experimental results are shown in Table 5. We compare our results with method in [43]. Note that camera settings in HumanEva dataset and IXMAS dataset are slightly different. This results in slight difference between human silhouettes from these two dataset. Also although we have four corresponding actions, they are not exactly the same action. We label all actions in IXMAS dataset semantically with those from HumanEva dataset. For example, "Gesture" action in HumanEva dataset semantically contains "Wave" and "Come". The proposed method is scene independent but not viewpoint independent. The compared method [43] is trained on IXMAS dataset and tested on the same dataset. We need to consider all these factors when compare these two methods.

Despite the differences between these two datasets, our models trained on HumanEva dataset obtain a relatively

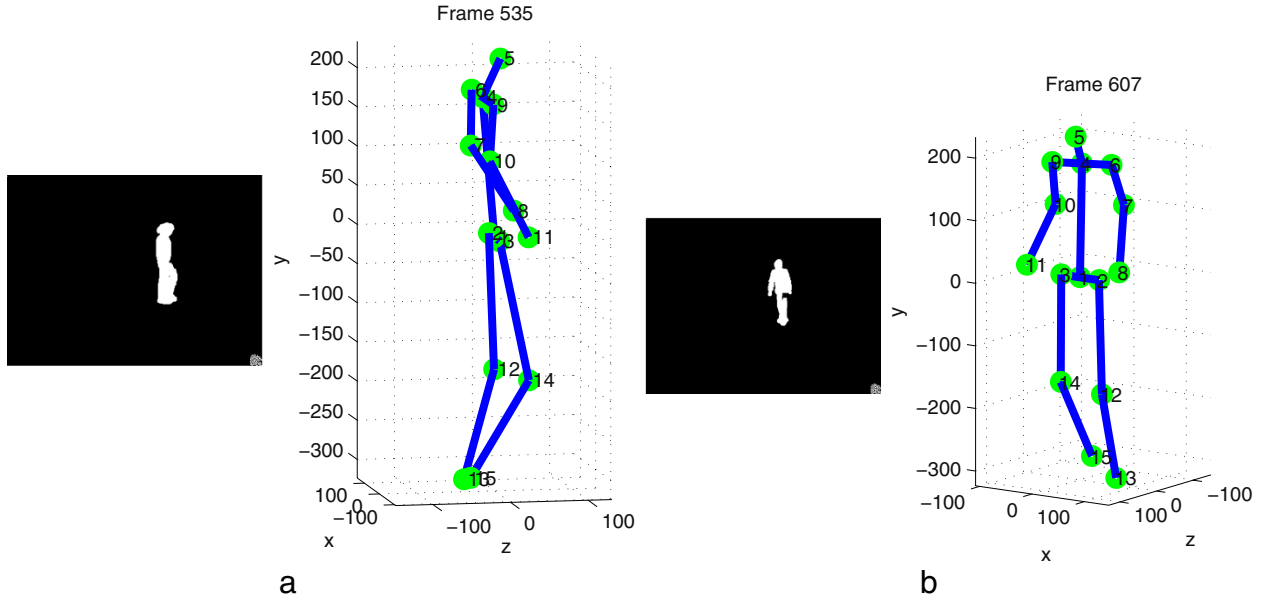

Figure 10 Two exampled frames of good estimations of *weak poses* in IXMAS dataset. *Weak poses* are back-projected from *UaSpace* to the original parameter space and visualized as human poses.

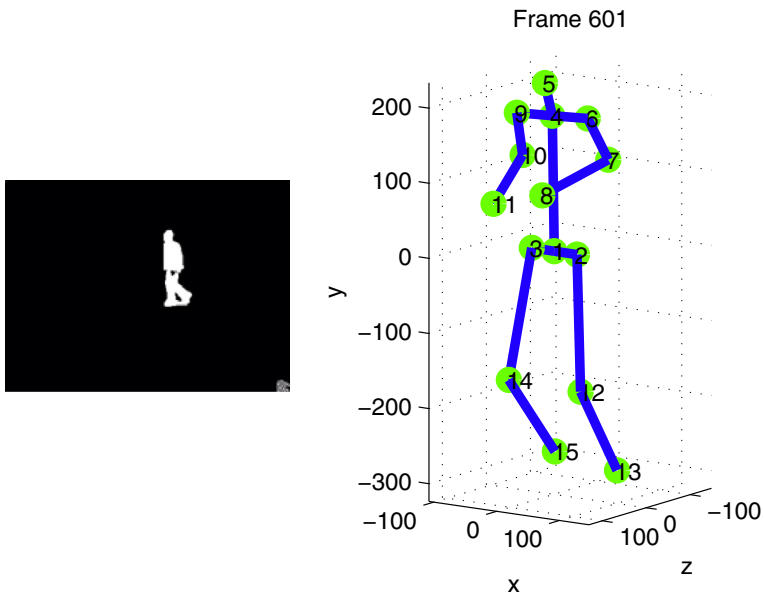

Figure 11 An example frame of bad estimation of a *weak pose* in IXMAS dataset.

close result as method in [43]. We even achieve better results with action "Walk". One explanation is that test data in "Walk" have more frames than other actions in IXMAS dataset, and our holistic method performs better with more frames. Another reason might be, "Walk" is a comparatively repetitive action that does not have as much variance as other actions when performed by a different human. While for other action, this is not the case. For example, for "Box" in HumanEva dataset, performer "*S*1" does not move his legs while performer "*S*2" jumps forward and backwards during the performances.

In Figures 10 and 11, we show sampled reconstruction of *weak poses*. We can see that in the condition of similar camera viewpoint and similar silhouette shapes, like in Figure 10, reconstructed poses can be very precise. While the differences between HumanEva dataset and IXMAS dataset, for example, different ways of actors performing the same actions, might cause some false prediction. One example is shown in Figure 11, where a walking pose is predicted as a running pose because the fierce movement of the legs is similar to that in a running pose from training.

Conclusions

In this article we have proposed a novel approach to action recognition using a BOP model with *weak poses* estimated from silhouettes. We have applied GPR to model the mapping from silhouettes to *weak poses*. We modify the classic BOW pipeline by incorporating temporal information. We train our models with the HumanEva dataset and test it with test data from HumanEva and IXMAS datasets. Experimental results show that our method performs effectively for the estimation of *weak poses* and action recognition. Even though different datasets have different camera setting and different perception about performing actions, our method is robust enough to obtain satisfactory results. Note that although the proposed method is not view-invariant, it is straightforward to extend to multiple view solution by including training data from all viewpoints. In prediction phase, viewpoint will be naturally selected in the regression procedure.

In further work, it would be interesting to model the dynamics of human poses in actions and also utilize this as priors for action recognition. An integrated regression model that incorporated 3D pose and 3D motion models into the GPR model described in this paper would likely improve the robustness of both *weak pose* estimation and action recognition.

Endnotes

[a] http://vision.cs.brown.edu/humaneva/
[b] http://4drepository.inrialpes.fr/public/viewgroup/6

Competing interests
The authors declare that they have no competing interests.

Acknowledgements
The authors acknowledge the support of the Spanish Research Programs Consolider-Ingenio 2010: MIPRCV (CSD200700018); Avanza I+D ViCoMo (TSI-020400-2009-133) and DiCoMa (TSI-020400-2011-55); along with the Spanish projects TIN2009-14501-C02-01 and TIN2009-14501-C02-02.

Received: 6 October 2011 Accepted: 15 May 2012
Published: 25 July 2012

References

1. H Ning, W Xu, Y Gong, T Huang, Latent pose estimator for continuous action recognition. in *ECCV,* (2008), pp. 419–433
2. P Siva, T Xiang, Action detection in crowd,. in *BMVC,* (2010), pp. 9.1–9.11
3. MS Ryoo, JK Aggarwal, Spatio-temporal relationship match: video structure comparison for recognition of complex human activities. in *ICCV,* (2009)
4. CC Chen, JK Aggarwal, Recognizing human action from a far field of view. in *IEEE Workshop on Motion and Video Computing,* (2009)
5. I Laptev, T Lindeberg, Space-time interest points. in *ICCV,* (2003), pp. 432–439
6. C Schüldt, I Laptev, B Caputo, Recognizing human actions: a local svm approach. in *ICPR,* (2004), pp. 32–36
7. JW Davis, AF Bobick, The representation and recognition of human movement using temporal templates. in *CVPR,* (1997), pp. 928–934
8. P Scovanner, S Ali, M Shah, A 3-dimensional sift descriptor and its application to action recognition. in *Proceedings of the 15th international conference on Multimedia,* (2007), pp. 357–360
9. S Ali, M Shah, Human action recognition in videos using kinematic features and multiple instance learning. PAMI. **32**, 288–303 (2010)
10. M Ahmad, SW Lee, Hmm-based human action recognition using multiview image sequences. in *ICPR,* (2006), pp. 263–266
11. D Weinland, R Ronfard, E Boyer, Motion history volumes for free viewpoint action recognition. in *ICCV PHI,* (2005)
12. M Brand, N Oliver, A Pentland, Coupled hidden markov models for complex action recognition. in *CVPR,* (1997), pp. 994–999
13. X Feng, P Perona, Human action recognition by sequence of movelet codewords. in *International Symposium on 3D Data Processing Visualization and Transmission,* (2002), pp. 717–721
14. D Weinland, E Boyer, R Ronfard, Action recognition from arbitrary views using 3d exemplars. in *ICCV,* (2007), pp. 1–7
15. M Zobl, F Wallhoff, G Rigoll, Action recognition in meeting scenarios using global motion features. in *In Proceedings Fourth IEEE International Workshop on Preformance Evaluation of Tracking and Surveillance,* (2003), pp. 32–36
16. R Poppe, A survey on vision-based human action recognition. Image Vis. Comput. **28**, 976–990 (2010)
17. D Weinland, R Ronfard, E Boyer, A survey of vision-based methods for action representation, segmentation and recognition. Comput. Vis. Image Understand. **115**, 224–241 (2011)
18. FF Li, P Perona, A bayesian hierarchical model for learning natural scene categories. in *CVPR,* (2005), pp. 524–531
19. S Lazebnik, C Schmid, J Ponce, Beyond bags of features: spatial pyramid matching for recognizing natural scene categories. in *CVPR,* (2006), pp. 2169–2178
20. A Bosch, A Zisserman, X Munoz, Representing shape with a spatial pyramid kernel. in *CIVR,* (2007), pp. 401–408
21. C Wallraven, B Caputo, ABA Graf, Recognition with local features: the kernel recipe. in *ICCV,* (2003), pp. 257–264
22. K Grauman, T Darrell, The pyramid match kernel: discriminative classification with sets of image features. in *ICCV,* (2005), pp. 1458–1465
23. SN Vitaladevuni, V Kellokumpu, LS Davis, Action recognition using ballistic dynamics. in *CVPR,* (2008), pp. 1–8
24. K Kulkarni, E Boyer, R Horaud, A Kale, An unsupervised framework for action recognition using actemes. in *ACCV,* (2011), pp. 592–605
25. R Souvenir, J Babbs, Viewpoint manifolds for action recognition. in *CVPR,* (2008), pp. 1–7
26. A Agarwal, B Triggs, Recovering 3d human pose from monocular images. PAMI. **28**, 44–58 (2006)
27. CE Rasmussen, CKI Williams, *Gaussian Processes for Machine Learning* (MIT Press, US, 2006)
28. R Urtasun, DJ Fleet, P Fua, 3d people tracking with gaussian process dynamical models. in *CVPR,* (2006), pp. 238–245
29. R Urtasun, T Darrell, Sparse probabilistic regression for activityindependent human pose inference. in *CVPR,* (2008), pp. 1–8
30. VM Zatsiorsky, *Kinetics of Human Motion* (Human Kinetics Publishers, US, 2002)
31. I Rius, J Gonzàlez, J Varona, FX Roca, Action-specific motion prior for efficient bayesian 3d human body tracking. Pattern Recogn. **42**, 2907–2921 (2009)
32. VM Zatsiorsky, *Kinematics of Human Motion* (Human Kinetics Publishers, US, 1998)
33. A Amato, M Mozerov, AD Bagdanov, J Gonzàlez, Accurate moving cast shadow suppression based on local color constancy detection. TIP. **20**, 2954–2966 (2011)
34. G Mori, J Malik, Recovering 3d human body configurations using shape contexts. PAMI. **28**, 1052–1062 (2006)
35. A Agarwal, B Triggs, Recovering 3d human pose from monocular images. PAMI. **28**, 44–58 (2006)
36. R Poppe, M Poel, Comparison of silhouette shape descriptors for example-based human pose recovery. in *Proceedings of the 7th International Conference on Automatic Face and Gesture Recognition,* (2006) pp. 541–546
37. BB Sofiane, A Bermak, Gaussian process for nonstationary time series prediction. Comput. Stat. Data Anal. **47**, 705–712 (2004)
38. JM Wang, DJ Fleet, A Hertzmann, Gaussian process dynamical models for human motion. PAMI. **30**, 283–298 (2008)
39. G Gregorčič, G Lightbody, Gaussian process approach for modelling of nonlinear systems. Eng. Appl. Artfic. Intell. **22**, 522–533 (2009)
40. KM Chai, C Williams, S Klanke, S Vijayakumar, Multi-task gaussian process learning of robot inverse dynamics. in *NIPS,* (2008), pp. 265–272
41. J Zhu, S Hoi, M Lyu, Nonrigid shape recovery by gaussian process regression. in *CVPR,* (2009), pp. 1319–1326
42. W Gong, AD Bagdanov, J Gonzàlez, FX Roca, Automatic key pose selection for 3d human action recognition. in *AMDO,* (2010)
43. F Lv, R Nevatia, Single view human action recognition using key pose matching and viterbi path searching. in *CVPR,* (2007) pp. 1–8
44. J Gu, X Ding, S Wang, Y Wu, Action and gait recognition from recovered 3-d human joints. IEEE Trans. Syst. Man Cybern. Part B. **40**, 1021–1033 (2010)
45. L Sigal, MJ Black, Humaneva: Synchronized video and motion capture dataset for evaluation of articulated human motion, Tech. Rep., Brown University, (2006)
46. A Bosch, A Zisserman, X Munoz, Image classification using random forests and ferns. in *ICCV,* (2007) pp. 1–8

doi:10.1186/1687-6180-2012-162
Cite this article as: Gong *et al.*: **Human action recognition based on estimated weak poses.** *EURASIP Journal on Advances in Signal Processing* 2012 **2012**:162.