EURASIP manuscript No.
(will be inserted by the editor)

Human Action Recognition based on Estimated Weak Poses

Wenjuan Gong · Jordi Gonzàlez · F.Xavier Roca

Received: date / Accepted: date

Abstract We present a novel method for human action recognition based on estimated poses from image sequences. The key idea behind our method is to take advantage of a compact human pose representation, called *weak pose*, in a low-dimensionality space while still keeping the most discriminative information for a given pose. Once the 2D silhouettes are detected in an input image and represented using shape context, Gaussian Process Regression is applied to estimate *weak poses*. Subsequently, we perform action recognition by considering a Bag of Poses model built on these estimated *weak poses*. The Bag of Poses 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 Bag of Poses 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 our results are scene-independent and go beyond the state of art.

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

1 Introduction

Human Action Recognition (HAR) is an important problem in computer vision. Application fields include video surveillance, automatic video indexing

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

and human computer interaction. Most solutions for HAR learn action patterns from sequences of image features like Space-Time Interest Points [3,4], temporal templates [5], 3D SIFT [6], optical flow [7,8], Motion History Volume [9], among others. These features are commonly used to describe human actions which are subsequently classified using techniques like Hidden Markov Models [8,10–13], and Support Vector Machines [4]. Recent and exhaustive reviews of methods for HAR can be found in [1,2].

Despite of its wide range of applications and the huge number of research works, action recognition from 2D image sequences still remains a challenging problem. One of the reasons is due to the high variability of scenarios and situations which can be found in videos, thus resulting in very different image qualities and content. As a result, we need to choose robust features and classification methods which can work well in multiple scenarios and for different actions.

One can categorize the scenarios found in the literature into several groups: single-human action [14], crowds [15], human-human interaction [16], and action recognition in aerial views [17], to cite but a few. Although the method proposed in this paper mainly concentrates on single-human action recognition, our contribution can be also applied to all the aforementioned scenarios, given that the 2D silhouettes of the agents are extracted from image sequences.

In this paper, our main hypothesis is that estimating 3D poses from 2D silhouettes can be advantageous for action recognition. Considering this, another reason that makes action recognition a challenging problem is the inherent ambiguities between 2D image features and 3D poses. Some researchers use multiple-view videos [18–20], 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) [21].

RVM is a special case of Gaussian Process Regression (GPR) [48]: 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 [22, 23].

Inspired by these works, the whole procedure presented in this paper is shown in figs 1 and 2. In essence the method is composed of two steps: training and prediction. In training, a set of Gaussian processes (first row fig. 1) and the Bag of Poses (BoP) model (second row fig. 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

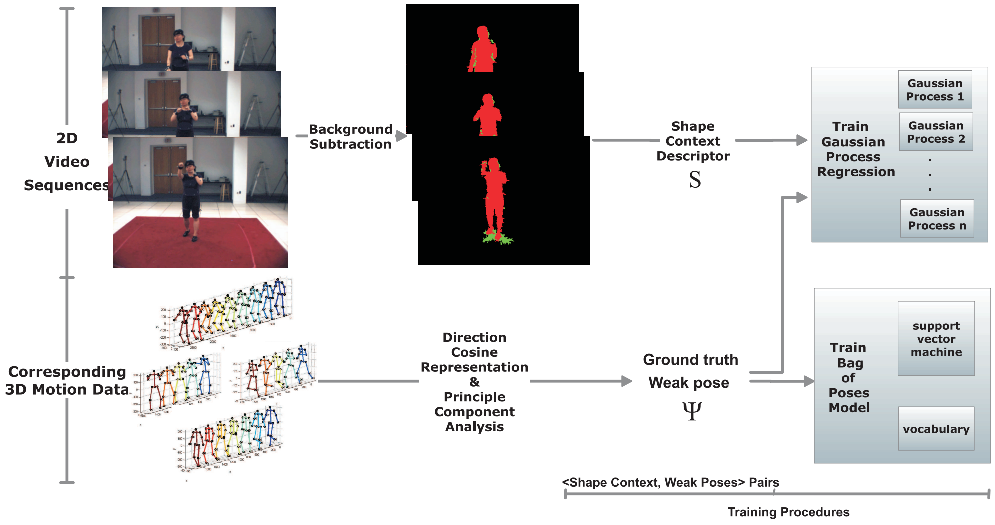


Fig. 1 Learning step: we train Gaussian processes to learn the regression function from shape context descriptors to *weak poses*. In parallel, a BoP model is built for each action class by extracting key poses and training SVM classifiers.

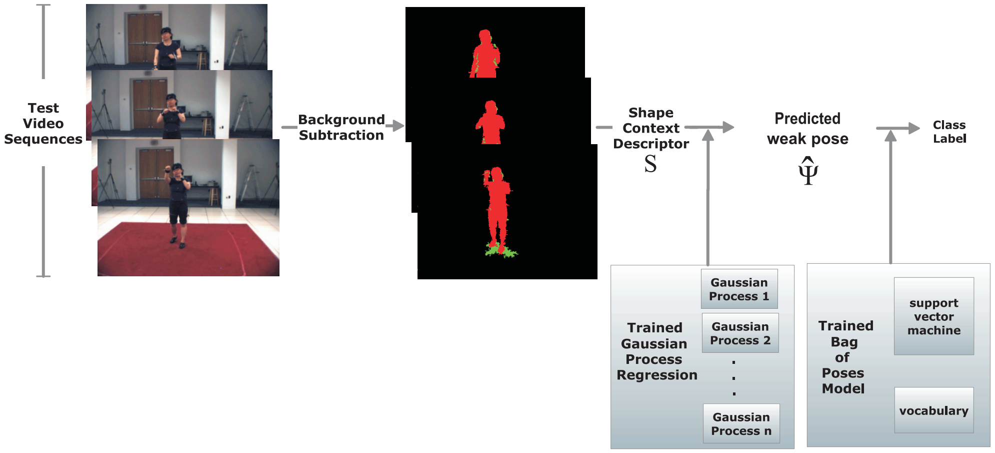


Fig. 2 Predicting phase. The test video sequence is described using shape context descriptors as in the learning phase (see fig. 1). *Weak poses* are predicted from shape context descriptors using trained Gaussian processes and the video is represented as a histogram of the vocabulary learned in the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

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 work most similar to our framework was proposed by [14]. They 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 authors in [14] use CRFs to model relations between image features and action labels, we label motion sequences with a BoP model, an extension of BoW [24–28].

The rest of the paper is organized as follows: next section introduces our human body model and human posture representation; section 3 describes how we use a set of Gaussian processes for learning the mapping from 2D image features to 3D human poses; in section 4, we describe a procedure for incorporating temporal information in a BoW schema, showing the results in section 5. Finally section 6 presents the future avenues of research.

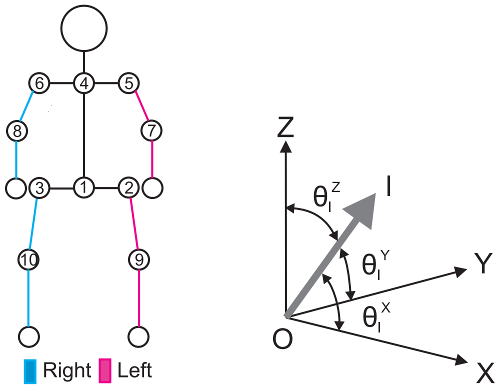
2 Data representation

The flexibility of the human body and the variability of human actions produce high-dimensional motion data. 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 twelve 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 fig. 3(a). 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.

Although some works only consider the 3D position of the markers at each time step [31–33], others have explored representations like polar angles [34] or Direction Cosines (DCs) [35]. In the latter case, the orientation of each limb is represented by three direction cosines 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 [36].

So we use the same representations for human postures and human motions as in [35]: a limb orientation is represented using three parameters, without



(a)

(b)

Fig. 3 (a) The 3D stick figure model used for representing human pose. Ten principal joints corresponding to the markers used in motion capture are used [38]. (b) The angles $(\theta_l^x, \theta_l^y, \theta_l^z)$ between the limb l and the axes [35].

modeling self rotation of the limb around its axes, as shown in fig. 3(b). This results in a 36-D representation of the pose of the actor in frame j of video i:

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

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

After representing static human postures using direction cosines, we represent a motion sequence of the actor in video i as a sequence of poses:

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

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

2.1 Universal Action Space or *UaSpace*

Since natural constraints of human body motions lead to highly correlated data [37], we build a more compact, non-redundant representation of human pose by applying Principle Component Analysis (PCA). This universal action space (*UaSpace*) will become the basis for vocabulary selection and finally classification using BoP.

We denote the pose representation in the reduced dimensionality space as *weak poses* or ψ'. By projecting human postures into the *UaSpace*, distances between poses of different actions can be computed and used for classification. Fig. 4 shows pose variation corresponding to the top (in terms of eigenvalues) 9 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

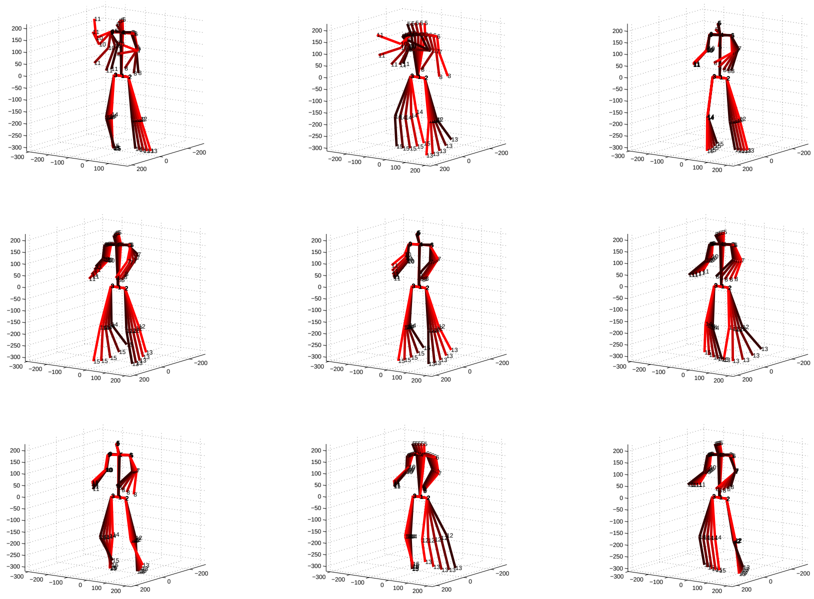


Fig. 4 Visualizing the 9 principal variations of the pose within *UaSpace* learnt from HumanEva data. Each plotted stick figure is a re-projected pose by moving it in one eigenvector's dimension from −3 up to 3 times the standard deviation.

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.

3 *Weak pose* estimation using GPR

We use Shape Context Descriptor (SCD) to represent the human silhouette found using background subtraction [44]. Shape context is commonly applied to describe shapes given silhouettes [45,46], and have been proven that it is an effective descriptor for human pose estimation [47] .

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 radiuses and angles. 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 5 bins equally spaced and divide angle into 12 equally spaced bins, as shown in fig. 5. As a result, the SCD vector is 60-D.

The normalization of the resulting SCD has a significant impact on the performance of Gaussian process regression. We exploit two different ways of

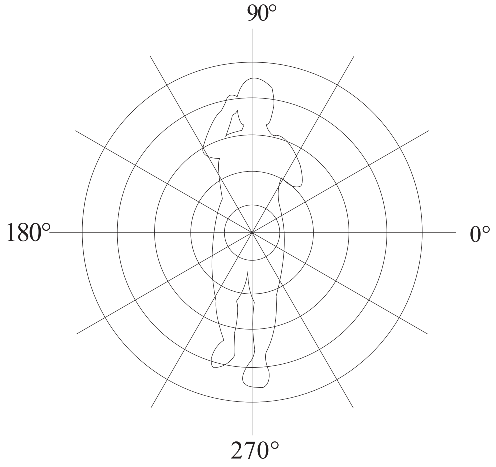


Fig. 5 Radial coordinates for shape context descriptor. The origin of the polar coordinate system is placed on the centroid of the bounding box of the silhouette. The radius is divided equally into 5 bins and the circle is divided equally into 12 bins.

normalizing data: standard deviation and individual normalizations. Suppose $\mathbf{s}_{orig}$ denotes the original shape context descriptor from one image, and

$$\mathbf{s}_{orig} = [np^1, np^2, \ldots, np^i, \ldots, np^{60}], \tag{3}$$

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 shape context descriptors $\mathbf{std} = [std^1, std^2, \ldots, std^{60}]$. Then we normalize each dimension of the shape context descriptor by dividing it with the corresponding standard deviation. If we represent the normalized shape context descriptor as $\mathbf{s}_{normlized}$, then

$$\mathbf{s}_{norm1} = [\frac{np^1}{std^1}, \frac{np^2}{std^2}, \ldots, \frac{np^i}{std^i}, \ldots, \frac{np^{60}}{std^{60}}] \tag{4}$$

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

$$\mathbf{s}_{norm2} = [\frac{np^1}{npSum}, \frac{np^2}{npSum}, \ldots, \frac{np^i}{npSum}, \ldots, \frac{np^{60}}{npSum}]. \tag{5}$$

We compare these two different ways of normalizing shape context descriptors in experimental results.

3.1 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 [39–41]. For example, Gaussian process has been applied to non-linear regression problems, like robot inverse dynamics [42] and nonrigid shape recovery [43].

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

$$\mathbf{S} = [\mathbf{s}^1, \mathbf{s}^2, \ldots, \mathbf{s}^p], \tag{6}$$

where $\mathbf{s}^i$ is the vector of shape context descriptor extracted from the i-th training video sequence. The methd described in [21] predicts 3D poses from 2D image features using Relevance Vector Machine (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 [22,23]. So in our approach, we will use GPR for modeling the mapping between silhouettes and *weak poses*.

According to [48], 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:

$$\zeta(\mathbf{s}) \sim \mathcal{GP}_j(m(\mathbf{s}), k(\mathbf{s}, \mathbf{s}')), \tag{7}$$

where

$$\begin{aligned} m(\mathbf{s}) &= E[\zeta(\mathbf{s})], \\ k(\mathbf{s}, \mathbf{s}') &= E[(\zeta(\mathbf{s}) - m(\mathbf{s}))(\zeta(\mathbf{s}') - m(\mathbf{s}'))], \end{aligned} \tag{8}$$

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:

$$k_1(\mathbf{s}, \mathbf{s}') = \theta_1^2 \exp(-\frac{(\mathbf{s} - \mathbf{s}')^2}{2\theta_2^2}). \tag{9}$$

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 [48] for details):

$$\log p(\Psi' | \mathbf{s}, \theta) = -\frac{1}{2} \Psi'^T K_{\Psi'}^{-1} \Psi' - \frac{1}{2} \log |K_{\Psi'}| - \frac{n}{2} \log 2\pi, \tag{10}$$

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 8.

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

$$\begin{aligned}\Psi'^{*}|\mathbf{s}^{*}, \mathbf{s}, \Psi' \sim \mathcal{N}(&\mathbf{k}(s^{*}, \mathbf{s})^{T}[K + \sigma^{2}_{noise}I]^{-1}\Psi', \\ &k(s^{*}, s^{*}) + \sigma^{2}_{noise} - \mathbf{k}(s^{*}, \mathbf{s})^{T}[K + \sigma^{2}_{noise}I]^{-1}\mathbf{k}(s^{*}, \mathbf{s})),\end{aligned} \quad (11)$$

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.

4 Bag of Poses 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 [24–26], 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 Bag of Poses (BoP) representation. Next we explain how to compute the vocabulary and perform classification with our modified BoP model.

4.1 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 [49]. We propose a new method for computing the vocabulary. First, we select candidate key *weak poses* using energy optimization as in [49]. The key *weak poses* are pre-selected as:

$$F^{i}_{pre} = \{f^{i}_{1}, f^{i}_{2}, , \ldots, f^{i}_{l}\}, \quad (12)$$

where f^{i}_{j} 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: $F_{pre} = \{F^{1}_{pre}, F^{2}_{pre}, \ldots, F^{p}_{pre}\}$, where F^{i}_{pre} is calculated as in equation 12 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.

To incorporate temporal information into our solution, we consider d consecutive frames as one unit. That is, key *weak poses* are preselected as $F_{pre}^{t} = \{F_{pre}^{t1}, F_{pre}^{t2}, \ldots, F_{pre}^{tl}\}$, where

$$F_{pre}^{tj} = [f_j^{frm-d+1}, f_j^{frm-d+2}, \ldots, f_j^{frm}] \tag{13}$$

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 5.1.3, we will show how we fix d using cross validation on training data.

4.2 Action Classification

A vocabulary is calculated as a collection of characteristic key *weak poses*. Then we represent our motion sequences as histograms over the vocabulary. That is, video sequence V^i is represented as:

$$hist^i = [n_1, n_2, \ldots, n_i, \ldots, n_{tk}], \tag{14}$$

where n_i is the number of *weak poses* in sequence V^i that are nearest to i-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.

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:

$$accuracy = \frac{tp + tn}{tp + tn + fp + fn}, \tag{15}$$

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.

5 Experimental results

To verify robustness of our method, we choose two public datasets: HumanEva and IXMAS. [14] 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 [18–20, 29]. Authors of [30] listed all state of art experimental results on this dataset. Among all, we will compare with experimental results in [29], 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 [1] dataset. 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 [2] 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.

5.1 Model training

In our experiments, we take the first half of each performance for training $< \mathbf{S}, \mathbf{\Psi} >$ and the second half for validation $< \mathbf{S}_{Val}, \mathbf{\Psi}_{Val} >$.

5.1.1 Number of Gaussian processes

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

$$\varepsilon = \frac{1}{N} \sum_{p=1}^{P} \sum_{f=1}^{F_p} \|\hat{\psi} - \psi'\|^2, \qquad (16)$$

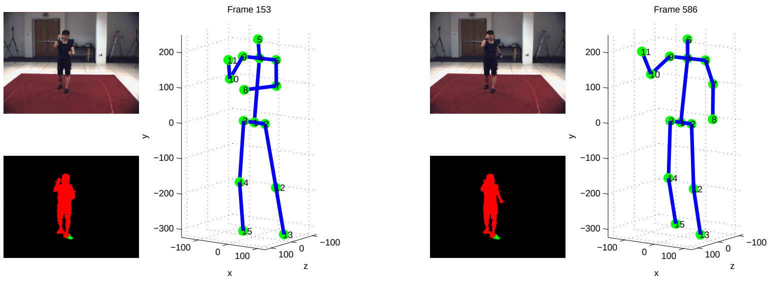
where N is the total number of frames used for training, P is the total number of training performances and F_p is frame numbers of the p-th training performance. To discard missing human detection, we first calculate the energy of shape context descriptor for each training frame and filter the training

[1] http://vision.cs.brown.edu/humaneva/

[2] http://4drepository.inrialpes.fr/public/viewgroup/6

GP dims	3	6	10	15	20	25	30
Voc size: 5	73.9	86.8	86.3	86.0	86.1	86.1	85.6
Voc size: 10	67.7	83.6	82.9	83.0	84.4	84.2	84.2
Voc size: 15	64.1	83.9	82.6	80.8	85.4	83.9	83.7
Voc size: 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

Table 1 Comparison of classification accuracy (%) and *weak pose* reconstruction error with different numbers of Gaussian processes and different vocabulary size. Reconstruction error is the difference between predicted *weak poses* and ground truth *weak poses*.



(a) (b)

Fig. 6 Two examples of good estimation of *weak poses* in back-projected from *UaSpace* to the original parameter space and visualized as human poses.

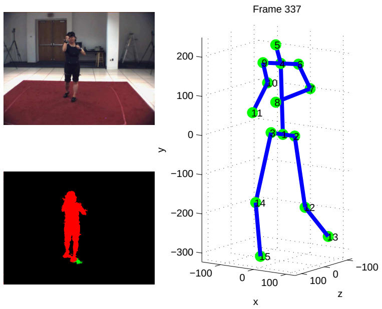


Fig. 7 An example of bad estimation of a *weak pose*.

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 1, we observe that with fewer than 20 Gaussian processes, increasing the number of Gaussian processes results in noticeable increases in

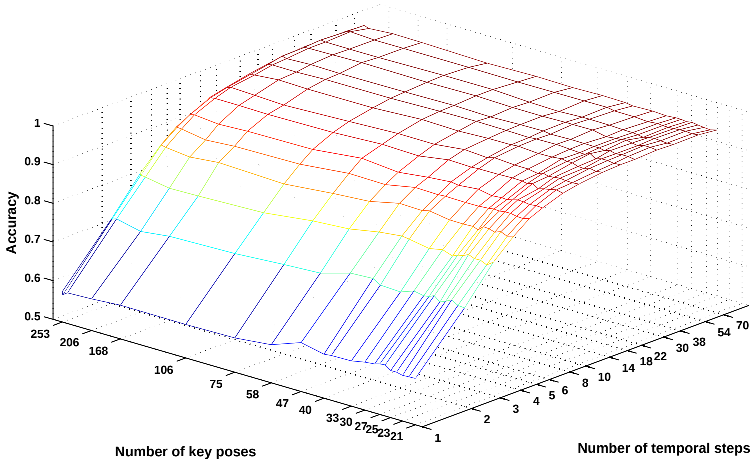


Fig. 8 The relations between number of temporal steps, number of key poses and action recognition accuracy.

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.

5.1.2 Weak pose *reconstruction results*

To visualize results of *weak pose* reconstruction, we project weak poses from *UaSpace* back to the original parameter space. Figs 6 and 7 show some examples of estimated *weak poses*. We can see that in fig. 6, pose estimation results are satisfactory. In fig. 7, there is a big difference between the estimation and the ground truth. But since our ultimate goal is action recognition and not pose estimation, we will not concentrate on further improvements on pose estimation. We show in following sections, that this pose estimation precision give a promising action recognition rate.

5.1.3 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 simulate

Acc.	Box	Jog	Gest	Walk	T/C	All - T/C	All + T/C
[14]	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**

Table 2 Comparison of action recognition accuracy (%) in HumanEva between our methods and the method presented in [14]. Classification accuracy is defined as correctly labeled samples over total number of samples(refer to equation 15). “Std-norm” and “Ind-norm” refer to standard deviation normalizing method and individually normalizing method (refer to section 3). 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”.

noisy 2D feature descriptors from video sequences. We run each noise scale 5 times and calculate average accuracy for all noise scales. Experiment results are shown in fig. 8. 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.

5.2 Action recognition accuracy

We utilize a BoP model in classifying actions, as described in section 4. 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 [50]. We first split training performances into sub-performances. Then, we translate sub-performances with *trans* times the maximum difference of the training data, where

$$trans = \{-0.20, -0.15, -0.10, -0.05, 0.05, 0.10, 0.15, 0.20\}, \qquad (17)$$

and scale sub-performances by

$$scale = \{0.80, 0.85, 0.90, 0.95, 1.05, 1.10, 1.15, 1.20\}. \qquad (18)$$

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 2. The method from [14] shows upper bound accuracy for initialized latent pose conditional random field model ($LPCRF_{init}$ in [14]) with the same training and test data.

Accuracy	Punch	Wave	Throw a ball	Walk	All actions
Ind-normal	75.0	79.2	75	87.5	79.2
[29]	86.8	79.9	82.4	79.7	82.2

Table 3 Action recognition accuracy (%) of our individually normalizing method for IX-MAS dataset using the models learnt from HumanEva dataset compared with the method prosed in [29].

In our experiments, normalization of input data is a very important step for Gaussian process regression to make good predictions. So we experimented with two different ways of normalizing data: standard-deviation based and individual normalizations. Our method with individual normalization has better average classification accuracy than the approach presented in [14].

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 normalized separatively. 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 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.

We further test our action model (trained using HumanEva data) on IXMAS dataset and experimetal results is shown in table 3. We compare our results with method in [29]. 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. But all corresponding actions in IXMAS dataset are subsets from HumanEva dataset. For example, “Gesture” action in HumanEva dataset semantically contains “Wave” and “Come”.

Despite the differences between these two datasets, our models trained on HumanEva dataset obtain a relatively close result as method in [29]. 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 relatively repetive 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 “$S1$” does not move his legs while performer “$S2$” jumps forward and backwards during the performances.

6 Conclusions and discussion

In this paper 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 have modifed 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.

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.

References

1. R. Poppe, A survey on vision-based human action recognition, Image and Vision Computing 28(2010) 976–990.
2. D. Weinland, R. Ronfard, E. Boyer, A survey of vision-based methods for action representation, segmentation and recognition, Computer Vision and Image Understanding 115(2011) 224–241.
3. I. Laptev, T. Lindeberg, Space-time interest points, ICCV 2003, pp. 432–439.
4. C. Schüldt, I. Laptev, Caputo, B., Recognizing human actions: a local svm approach, ICPR 2004, pp. 32–36.
5. J.W. Davis, A.F. Bobick, The representation and recognition of human movement using temporal templates, CVPR 1997, pp. 928–934.
6. 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.
7. S. Ali, M. Shah, Human action recognition in videos using kinematic features and multiple instance learning, PAMI 32(2010) 288–303.

8. M. Ahmad, S.W. Lee, Hmm-based human action recognition using multiview image sequences, ICPR 2006, pp. 263–266.
9. D. Weinland, R. Ronfard, E. Boyer, Motion history volumes for free viewpoint action recognition, ICCV PHI 2005.
10. M. Brand, N. Oliver, A. Pentland, Coupled hidden markov models for complex action recognition, CVPR 1997, pp. 994–999.
11. 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.
12. D. Weinland, E. Boyer, R. Ronfard, Action recognition from arbitrary views using 3d exemplars, ICCV 2007, pp. 1–7.
13. M. Zobl, F. Wallhoff, G. Rigoll, Action recognition in meeting scenarios using global motion features, In Proceedings Fourth IEEE International Workshop on Preformance Evaluation of Tracking and Surveillance 2003, pp. 32–36.
14. H. Ning, W. Xu, Y. Gong, T. Huang, Latent pose estimator for continuous action recognition, ECCV 2008, pp. 419–433.
15. P. Siva, T. Xiang, Action detection in crowd, BMVC 2010, pp. 9.1–9.11.
16. M.S. Ryoo, J.K. Aggarwal, Spatio-temporal relationship match: Video structure comparison for recognition of complex human activities, ICCV 2009.
17. C.C. Chen, J.K. Aggarwal, Recognizing human action from a far field of view, In IEEE Workshop on Motion and Video Computing 2009.
18. S.N. Vitaladevuni, V. Kellokumpu, L.S. Davis, Action recognition using ballistic dynamics, CVPR 2008, pp. 1–8.
19. K. Kulkarni, E. Boyer, R. Horaud, A. Kale, An unsupervised framework for action recognition using actemes, ACCV 2011, pp. 592–605.
20. R. Souvenir, J. Babbs, Viewpoint manifolds for action recognition, CVPR 2008, pp. 1–7.
21. A. Agarwal, B. Triggs, Recovering 3d human pose from monocular images, PAMI 28(2006) 44–58.
22. R. Urtasun, D.J. Fleet, P. Fua, 3d people tracking with gaussian process dynamical models, CVPR 2006, pp. 238–245.
23. R. Urtasun, T. Darrell, Sparse probabilistic regression for activityindependent human pose inference, CVPR 2008, pp. 1–8.
24. F.F. Li, P. Perona, A bayesian hierarchical model for learning natural scene categories, CVPR 2005, pp. 524–531.
25. S. Lazebnik, C. Schmid, J. Ponce, Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories, CVPR 2006, pp. 2169–2178.
26. A. Bosch, A. Zisserman, X. Munoz, Representing shape with a spatial pyramid kernel, CIVR 2007, pp. 401–408.
27. C. Wallraven, B. Caputo, A.B.A. Graf, Recognition with local features: the kernel recipe, ICCV 2003, pp. 257–264.
28. K. Grauman, T. Darrell, The pyramid match kernel: discriminative classification with sets of image features, ICCV 2005, pp. 1458–1465.
29. F. Lv, R. Nevatia, Single view human action recognition using key pose matching and viterbi path searching, CVPR 2007, pp. 1–8.
30. J. Gu, X. Ding, S. Wang, Y. Wu, Action and gait recognition from recovered 3-d human joints, IEEE Transactions on Systems, Man, and Cybernetics Part B 40(2010) 1021–1033.
31. F. Lv, R. Nevatia, M.W. Lee, 3d human action recognition using spatio-temporal motion templates, ICCV Workshop on Human-Computer Interaction 2005, pp. 120–130.
32. F. Lv, R. Nevatia, Recognition and segmentation of 3d human action using hmm and multi-class adaboost, ECCV 2006, pp. 359–372.
33. M. Raptis, K. Wnuk, S. Soatto, Flexible dictionaries for action classification, In Proceedings of the Workshop on Machine Learning for Visual Motion Analysis, 2008.
34. J. Gonzàlez, D. Rowe, J. Varona, F.X. Roca, Understanding dynamic scenes based on human sequence evaluation, Image and Vision Computing 27(2009) 1433–1444.
35. I. Rius, J. Gonzàlez, J. Varona, F.X. Roca, Action-specific motion prior for efficient bayesian 3d human body tracking, Pattern Recognition 42(2009) 2907–2921.

36. V.M. Zatsiorsky, Kinetics of Human Motion, Human Kinetics Publishers, 2002.
37. V.M. Zatsiorsky, Kinematics of Human Motion, Human Kinetics Publishers, 1998.
38. L. Sigal, M.J. Black, Humaneva: Synchronized video and motion capture dataset for evaluation of articulated human motion, Tech. Rep., Brown University, 2006.
39. B.B. Sofiane, A. Bermak, Gaussian process for nonstationary time series prediction, Computational Statistics and Data Analysis 47(2004) 705–712.
40. J.M. Wang, D.J. Fleet, A. Hertzmann, Gaussian process dynamical models for human motion, PAMI 30(2008) 283–298.
41. G. Gregorčič, G. Lightbody, Gaussian process approach for modelling of nonlinear systems, Engineering Applications of Artificial Intelligence 22(2009) 522–533.
42. K.M. Chai, C. Williams, S. Klanke, S. Vijayakumar, Multi-task gaussian process learning of robot inverse dynamics, NIPS 2008.
43. J. Zhu, S. Hoi, M. Lyu, Nonrigid shape recovery by gaussian process regression, CVPR 2009, pp. 1319–1326.
44. A. Amato, M. Mozerov, I. Huerta, J. Gonzàlez, J.J. Villanueva, Background subtraction technique based on chromaticity and intensity patterns, ICPR 2008, pp. 1–4.
45. G. Mori, J. Malik, Recovering 3d human body configurations using shape contexts, PAMI 28(2006) 1052–1062.
46. A. Agarwal, B. Triggs, Recovering 3d human pose from monocular images, PAMI 28(2006) 44–58.
47. 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.
48. C.E. Rasmussen, C.K.I. Williams, Gaussian Processes for Machine Learning, MIT Press, 2006.
49. W. Gong, A.D. Bagdanov, J. Gonzàlez, F.X. Roca, Automatic key pose selection for 3d human action recognition, AMDO 2010.
50. A. Bosch, A. Zisserman, X. Munoz, Image classification using random forests and ferns, ICCV 2007, pp. 1–8.