A Hierarchical Model of Shape and Appearance for Human Action Classification

Juan Carlos Niebles
Universidad del Norte, Colombia
University of Illinois at Urbana-Champaign, USA
jnieble2@uiuc.edu

Li Fei-Fei
Princeton University, USA
feifeili@cs.princeton.edu

Abstract

We present a novel model for human action categorization. A video sequence is represented as a collection of spatial and spatial-temporal features by extracting static and dynamic interest points. We propose a hierarchical model that can be characterized as a constellation of bags-of-features and that is able to combine both spatial and spatial-temporal features. Given a novel video sequence, the model is able to categorize human actions in a frame-by-frame basis. We test the model on a publicly available human action dataset [2] and show that our new method performs well on the classification task. We also conducted control experiments to show that the use of the proposed mixture of hierarchical models improves the classification performance over bag of feature models. An additional experiment shows that using both dynamic and static features provides a richer representation of human actions when compared to the use of a single feature type, as demonstrated by our evaluation in the classification task.

1. Introduction

It is of great practical and scientific interests to understand articulated body motions, especially those of the human body. In computer vision, one intriguing problem is to represent the different types of human motions with effective models. In this paper, we focus on the problem of human motion categorization under uncontrolled camera condition. In particular, we propose a generative model that takes into account both static and dynamic features of human motion. Our aim is to offer a generic solution to both human motion and pose categorization via flexible yet highly descriptive models.

Based on the recent works in human motion categorization [2, 10, 14, 16], we make two key observations that will in turn influence the design of our model. The first observation is based on the usage of different feature descriptors to represent human body and/or human motion. The second observation deals with the choice of the category model that uses such features for corresponding classification.

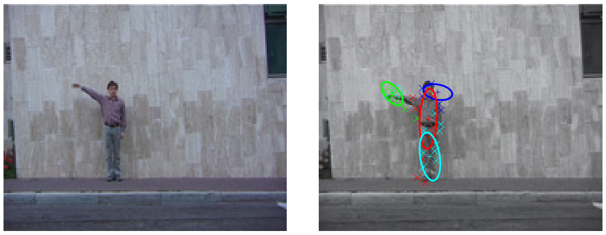


Figure 1. **Recognizing human action classes:** A sample frame and a four part model for *hand waving* over imposed on the original image. Static and dynamic features are shown, colored according to their part membership.

Using good features to describe pose and motion has been widely researched in the past few years. Generally speaking, there are three popular types of features: static features based on edges and limb shapes [7, 11, 15]; dynamic features based on optical flows [7, 9, 18], and spatial-temporal features that characterizes a space-time volume of the data [2, 6, 8, 13]. Spatial-temporal features have shown particular promise in motion understanding due to its rich descriptive power [3, 14, 17]. On the other hand, to only rely on such features means that one could only characterize motions in videos. Our daily life experiences tell us, however, humans are very good at recognizing motion based on a single gesture. Fanti et al. proposed in [10] that it is fruitful to utilize a mixture of both static and dynamic features. In their work, the dynamic features are limited to simple velocity description. We therefore propose the *hybrid usage of static shape features as well as spatial-temporal features* in our framework.

Model representation and learning are critical for the ultimate success of any recognition framework. In human motion recognition, most models are divided into either discriminative models or generative models. For example, based on the spatial-temporal cuboids, Dollar et al. [8] applied an SVM classifier to learn the differences among videos containing different human motions. Ramanan et al. [15] recently proposed a Conditional Random Field model to estimate human poses. While discriminative frameworks are often very successful in the classification results, they suffer either the laborious training problem or a lack of true understanding of the videos or images. In the CRF framework, one needs to train the model by labeling by hand each part of the human body. And in the SVM framework, the model is not able to "describe" the actual motion of the person. Some researchers, therefore, have proposed several algorithms based on probabilistic graphical model frameworks in action categorization/recognition. Song et al. [20] and Fanti et al. [10] represent the human action model as a triangulated graph. Boiman and Irani [3] recently propose to extract ensemble of local video patches to localize irregular action behavior in videos. Dense sampling of the patches is necessary in their approach and therefore the algorithm is very time-consuming. It is not suitable for action recognition purpose due to the large amount of video data commonly presented in these settings.

For structured objects such as human bodies, it is important to model the mutual geometric relationship among different parts. Constellation models offer such a solution [10, 21]. Unfortunately due to the computational complexity of the model, previous works can only use a very small number of features (typically 4 to 6) or approximate the connections by triangulation [10, 20]. Another approach is to lose all the geometric information and consider "bag of words" models. They have proven to be highly efficient and effective in classifying objects [12, 19] and human motion [8, 14]. We propose here a method to exploit both the geometric power of the constellation model as well as the richness of the "bag of words" model. We recognize the computational limit of having a very small number of fully connected parts in the constellation model. But instead of applying it directly onto the image level features, we attach a "bag of words" model to each part of the constellation model. The overall representation embodies a hierarchical model that combines a constellation model of few parts with bag of words models of a large and flexible number of features (see Fig. 2). Our model is partly inspired by a hierarchical model proposed by Bouchard and Triggs in [4]. In their framework, they also use the idea of attaching large number of features at the image level to a handful of intermediate level parts. The key difference between our model and theirs is that our intermediate level parts are fully connected whereas theirs are not, offering a much richer constraint. In addition, we use a mixture of models for our motion classes whereas it is not immediately clear whether their framework could be easily extended to a mixture model.

In summary, we show in this paper a hierarchical model that learns different categories of human motion using a hybrid of spatial-temporal and static features. Our model can be characterized as a constellation of bag of words. Our results show that compared to previous works, our model offers superior classification performances on a number of large human motion datasets. In addition, it can do it either on a video sequence or in individual frames.

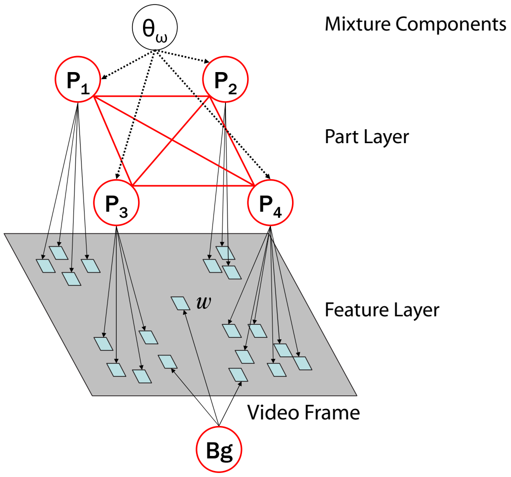


Figure 2. **Hierarchical model for human actions:** The proposed model combines, in a hierarchical way, the geometric strength of the constellation model with the larger number of features utilized in the bag of keypoints models. The higher layer is a constellation of P parts, each associated to a bag of features in the lower layer. The parts are interrelated by a distribution of their relative positions. Additionally, each part defines a distribution of appearance and position of features assigned to it.

2. Theoretical Framework

In the simplest version, our model is a two layered hierarchical model. The higher layer is close in spirit to the shape term of the constellation model. It is composed by a set of P parts whose position is represented by a Gaussian density of their relative locations. Each of the P_p parts ($p = 1 \dots P$) is connected to N_p image features in the lower level, and is associated to distributions of appearance and relative location of the features assigned to it. In other words, the higher layer is a constellation of parts, and each of these parts is associated to a "bag of features" in the lower layer. Due to its geometric constraints, this model is suitable to capture similar body configurations or poses.

Following the observation that human actions are results of sequences of poses, which arise from a few sets of similar body configurations, we believe that a single action is better represented as a multimodal distribution of shape and appearance. To account for this multimodality, we use a mixture of hierarchical models, where each component corresponds to a set of poses clustered together according to their similarity.

2.1. The Hierarchical Model

Given a video frame $\mathbf{I}$, we find a set of N observed features $\mathbf{w} = \{\mathbf{x}, \mathbf{a}\}$, where $w_i = \{x_i, a_i\}$ denotes position x_i and appearance a_i information. We also suppose that there is a known finite set $\mathbf{Y}$ of possible positions for the P parts in the image. One can think of $\mathbf{Y}$ as pixel locations or any arbitrary choice. We can compute the likelihood of the observed data given an action model θ as the following:

$$p(\mathbf{w}, \mathbf{Y}|\theta) = \sum_{\omega=1}^{\Omega} \sum_{\mathbf{h}\in H} p(\mathbf{w}, \mathbf{Y}, \mathbf{h}, \omega|\theta) \tag{1}$$

$$= \sum_{\omega=1}^{\Omega} \sum_{\mathbf{h}\in H} p(\mathbf{w}, \mathbf{Y}, \mathbf{h}|\omega, \theta) p(\omega|\theta) \tag{2}$$

$$= \sum_{\omega=1}^{\Omega} \left[\pi_\omega \sum_{\mathbf{h}\in H} p(\mathbf{w}, \mathbf{Y}, \mathbf{h}|\omega, \theta) \right] \tag{3}$$

where ω indicates the mixture component, we define $\pi_\omega = p(\omega|\theta)$ such that $\sum_\omega \pi_\omega = 1$ and $\mathbf{h}$ is an indexing variable which we call a *hypothesis* (similar to the constellation model). If $|\mathbf{Y}|$ is the number of possible locations for the P parts, then $\mathbf{h}$ is a vector of length P, where each element is between 1 and $|\mathbf{Y}|$. Additionally, we introduce the variable $\mathbf{m}$, which indicates an assignment of features to parts. In particular, each $\mathbf{m}$ is a vector of N elements which can take integer values on the interval $[0, P]$. That means, each feature can be assigned to the background (0) or to one of the P parts $(1 \ldots P)$. Marginalizing over $\mathbf{m}$, we rewrite the observed data likelihood as:

$$p(\mathbf{w}, \mathbf{Y}|\theta) = \sum_{\omega=1}^{\Omega} \left[\pi_\omega \sum_{\mathbf{h}\in H} \sum_{\mathbf{m}\in M} p(\mathbf{w}, \mathbf{Y}, \mathbf{h}, \mathbf{m}|\theta_\omega) \right] \tag{4}$$

$$p(\mathbf{w}, \mathbf{Y}|\theta) = \sum_{\omega=1}^{\Omega} \Bigg[\pi_\omega \sum_{\mathbf{h}\in H} \Bigg(p(\mathbf{h}|\theta_\omega) p(\mathbf{Y}|\mathbf{h}, \theta_\omega) \sum_{\mathbf{m}\in M} p(\mathbf{w}|\mathbf{Y}, \mathbf{m}, \mathbf{h}, \theta_\omega) p(\mathbf{m}|\mathbf{Y}, \mathbf{h}, \theta_\omega) \Bigg) \Bigg] \tag{5}$$

Calculating the likelihood in Eq (5) requires to compute $O((P+1)^N)$ different assignments for each $\mathbf{h}$. Considering that $|H| = |\mathbf{Y}|^P$, we need to compute the probabilities of $O\left((P+1)^N |\mathbf{Y}|^P\right)$ different combinations of hypothesis-assignment. In order to make the model more computationally tractable, we propose the following approximation:

$$\sum_{\mathbf{m}\in M} p(\mathbf{w}|\mathbf{Y}, \mathbf{m}, \mathbf{h}, \theta_\omega) p(\mathbf{m}|\mathbf{Y}, \mathbf{h}, \theta_\omega) \approx p(\mathbf{w}|\mathbf{Y}, \mathbf{h}, \mathbf{m}^*, \theta_\omega)$$

That is, we compute only one assignment per hypothesis. If we assume that $p(\mathbf{m}|\mathbf{Y}, \mathbf{h}, \theta)$ is uniform, then $\mathbf{m}^*$ is selected such that:

$$\mathbf{m}^* = \arg\max_{\mathbf{m}} p(\mathbf{w}|\mathbf{Y}, \mathbf{h}, \mathbf{m}, \theta) \tag{6}$$

Applying this to (5), the approximated observed data likelihood is:

$$p(\mathbf{w}, \mathbf{Y}|\theta) \approx \sum_{\omega=1}^{\Omega} \left[\pi_\omega \sum_{\mathbf{h}\in H} p(\mathbf{h}|\theta_\omega) \underbrace{p(\mathbf{Y}|\mathbf{h}, \theta_\omega)}_{\textit{Part layer}} \underbrace{p(\mathbf{w}|\mathbf{Y}, \mathbf{m}^*, \mathbf{h}, \theta_\omega)}_{\textit{Local feature layer}} \right] \tag{7}$$

Part layer term: We represent the joint probability of the position of the P parts in the model as a multivariate gaussian distribution:

$$p(\mathbf{Y}|\mathbf{h}, \theta) = \mathcal{N}(\mathbf{Y}_\mathbf{T}(\mathbf{h})|\boldsymbol{\mu}_L, \boldsymbol{\Sigma}_L)$$

In order to obtain translation invariance, we map $\mathbf{Y}$ into a translation invariance space, by constructing $\mathbf{Y}_\mathbf{T}(\mathbf{h})$, a $2(P-1)$ dimensional vector that contains the relative positions of $(P-1)$ parts with respect to the topmost part.

Local feature layer term: Given a part-to-feature assignment, each part P is instantiated as a set of image features that carry appearance and location information. Thus, each part is associated with an appearance distribution as well as a relative position distribution of image features. We adopt the bag-of-features assumption, where the observations $\mathbf{w}_n \in \mathbf{I}$ are conditionally independent given their parent assignments in $\mathbf{m}$. This assumption allows us to write the likelihood of a set of observations $\mathbf{w}$, given the possible part locations $\mathbf{Y}$, a hypothesis $\mathbf{h}$, an assignment $\mathbf{m}$ and the model parameters θ, as:

$$\begin{aligned} p(\mathbf{w}|\mathbf{Y}, \mathbf{h}, \mathbf{m}, \theta) &= \prod_{w_n \in \mathbf{I}} p(\mathbf{w}_n|\mathbf{Y}, \mathbf{h}, \mathbf{m}_n, \theta) \\ &= \prod_{\mathbf{w}_j \in bg} p(\mathbf{w}_j|\theta_0) \prod_{p=1}^{P} \prod_{\mathbf{w}_i \in P_p} p(\mathbf{w}_i|\mathbf{Y}, h_p, \theta_p) \\ &= \prod_{\mathbf{w}_j \in bg} p(x_j^r|\theta_0^X) p(a_j|\theta_0^A) \prod_{p=1}^{P} \prod_{\mathbf{w}_i \in P_p} p(x_i^r|\mathbf{Y}, h_p, \theta_p^X) p(a_i|\theta_p^A) \end{aligned} \tag{8}$$

where we define $\theta_p^X = \left\{ \boldsymbol{\mu}_p^X = 0, \boldsymbol{\Sigma}_p^X \right\}$ to be the parameters of a Gaussian distribution that determines the relative position of the features that belong to the p-th parent. Note that given a particular $\mathbf{m}$, the position information x_i of the i-th image feature can be transformed to the relative location, x_i^r, of the feature to its assigned parent. Similarly, θ_p^A are the parameters of a multinomial distribution that describe the appearance of the features assigned to the p-th parent. In the same manner, we define θ_0^X and θ_0^A as parameters for the appearance and position distribution of features assigned to the background. Note that the notations $\mathbf{w}_i \in P_p$ and $\mathbf{w}_j \in bg$ indicate assignments that depend on both $\mathbf{h}$ and $\mathbf{m}$.

Additionally, the conditionally independence assumption allows us to maximize $p(\mathbf{w}_n|\mathbf{Y}, \mathbf{h}, \mathbf{m}_n, \theta_\omega)$ with respect to $\mathbf{m}$ for each $\mathbf{w}_n$ independently. In other words, our

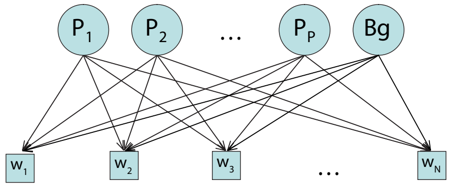


Figure 3. **Matching features (children) to parts (parents):** The weight of each link between a parent node P_p and a child node $\mathbf{w}_n$ is the probability of generating $\mathbf{w}_n$ from the appearance and position distributions assigned to P_p: $p(\mathbf{w}_i|\mathbf{Y}, h_p, \theta_p) = p(x_i|\mathbf{Y}, h_p, \theta_p^X)p(a_i|\theta_p^A)$. We also include a background node at the parent level, to allow features to be assigned to the background.

task reduces to find the best parent for each child node $\mathbf{w}_n$ in the graph of Figure 3.

Note that from this procedure, it is possible for $\mathbf{m}^*$ to be a feature to part assignment such that a part has no features assigned to it. This allows the model to handle naturally missing or occluded parts. An alternative to be explored is to assign features to parts softly, instead of using the single best parent for each child.

Approximated data likelihood: Assuming that the prior probability of selecting a particular hypothesis $\mathbf{h}$ is uniform, i.e. $p(\mathbf{h}|\theta) = |H|^{-1}$, we can finally rewrite our likelihood equation as:

$$p(\mathbf{w}, \mathbf{Y}|\theta) \approx \frac{1}{|H|} \sum_{\omega=1}^{\Omega} \left[\pi_\omega \sum_{\mathbf{h} \in H} \mathcal{N}(\mathbf{Y}_\mathbf{T}(\mathbf{h})|\boldsymbol{\mu}_L, \boldsymbol{\Sigma}_L) p(\mathbf{w}|\mathbf{Y}, \mathbf{m}^*, \mathbf{h}, \theta_\omega) \right] \tag{9}$$

2.2. Learning

Learning consists of estimating the model parameters for each action category. In the case of the mixture of models, each action class has the set of parameters $\theta_\omega = \left\{ \boldsymbol{\mu}_{L,\omega}, \boldsymbol{\Sigma}_{L,\omega}, \boldsymbol{\Sigma}^X_{p,\omega}, \theta^A_{p,\omega}, \theta^X_0, \theta^A_0 \right\}$ for $p = 1 \ldots P$ and $\omega = 1 \ldots \Omega$. To accomplish this purpose, we adopt an EM algorithm.

Initialization: The convergence of the EM algorithm to a sensible minimum depends greatly on the starting point. In order to select a good initial point, we cluster video frames from the training data into a number of clusters equal to the number of mixture components. The clustering procedure is done by representing each video frame with a histogram of features. Then, we select a small number of frames from each resulting cluster and fit a 1-component model to them. The output of this procedure is a set of initial parameters θ^{old}.

E-Step: Evaluate the responsibilities using the current parameter values θ^{old}:

$$p(\mathbf{h}, \omega|\mathbf{w}, \mathbf{Y}, \theta^{old}) \approx \frac{\pi_\omega p(\mathbf{Y}|\mathbf{h}, \theta^{old}_\omega) p(\mathbf{h}|\theta^{old}_\omega) p(\mathbf{w}|\mathbf{Y}, \mathbf{h}, \mathbf{m}^*, \theta^{old}_\omega)}{p(\mathbf{w}, \mathbf{Y}|\theta^{old})} \tag{10}$$

M-Step: Calculate updated parameters θ^{new} using the current responsibilities:

$$\theta^{new} = \arg\max_\theta \sum_\mathbf{h} p(\mathbf{h}, \omega|\mathbf{w}, \mathbf{Y}, \theta^{old}) \ln p(\mathbf{w}, \mathbf{Y}, \mathbf{h}, \omega|\theta) \tag{11}$$

2.3. Recognition

Given a new video frame and the learnt models for each action class, the task is to classify the new data as belonging to one of the action models. Suppose that we have C number of classes. We calculate the likelihood of observing the image data given that it has been generated from each class C. This produces a C-dimensional feature vector of the input in the model space. We calculate these feature vectors for each example in a *validation set*, and use them to train a discriminative classifier. Therefore, a classification decision is made by first calculating the likelihood of the input according to each of the C action models, and then categorizing this C-dimensional feature vector using the discriminative classifier.

Additionally, decisions can be made over a range of video frames by adopting a bag-of-frames strategy. First, each frame is categorized independently, and assigns a vote in favor of an action class. The complete video sequence is classified to be from the category with the majority of the votes.

3. The System

Image Features: We represent each video frame as a set of detected patches $\mathbf{w} = \{\mathbf{x}, \mathbf{a}\}$, where $w_i = \{x_i, a_i\}$, $i = 1 \ldots N$. The appearance information $\mathbf{a}$ is obtained by assigning each patch a membership to a large dictionary of codewords. We show now how these patches are obtained and memberships assigned.

We adopt a rich representation by detecting static and motion features. This allows the model to characterize a larger number of human actions than when using motion alone. Specifically, certain actions, such as hand waving in [2], produce a small number of motion features since most body parts remain static.

Static features are obtained by first computing an edge map using Canny edge detector. A set of edge points is sampled from the edge map, and a descriptor is obtained for an image patch around each selected point by calculating its shape context [1].

Motion features are obtained using the separable linear filter method in [8]. Small video patches are extracted and described by concatenating their gradients on space and time directions.

Given the collection of detected static features from the training images of all categories, we learn a codebook by the employment of a k-means algorithm. Codewords are then defined as the centers of the learnt clusters, and each static patch is assigned to the closest codeword. A similar procedure is performed to obtain a codebook of motion features, and the corresponding memberships.

The employment of two different types of features requires to adopt two different distributions of feature appearance for each part. In particular, $p(a_i|\theta_p^a)$ is actually modeled as two multinomial distributions, one for static features and other for motion features. Thus, given a particular feature, we use the appropriate appearance distribution when calculating 8. Note that the proper distribution to use can be determined unambiguously since the type of the feature is always known.

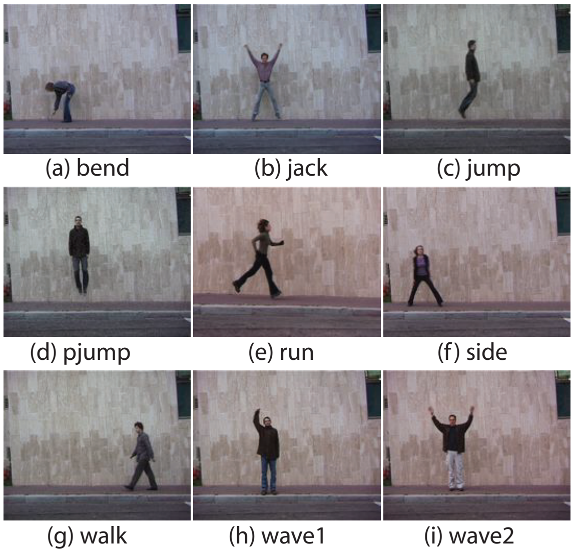
Implementation Details: In our implementation, we detect spatial features at each frame by sampling edge points from the output of the Canny edge detector. The number of samples is fixed to 100. Each sampled edge point is described using shape context with 3 spatial and 8 angular bins. The dimensionality of both descriptor types (static and dynamic) is reduced using PCA. Consequently, we cluster static and motion descriptors into codebooks of size 100. The discriminative classifier described in section 2.3 is instantiated by a Support Vector Machine. For this purpose, we use a linear SVM trained with LIBSVM [5].

4. Experiments

We test our model using the human action dataset from [2]. It contains 9 action classes performed by 9 different subjects, some example frames are shown in figure 4. There are 83 sequences in total, since each class contains 9 or 10 videos.

We adopt a leave-one-out scheme for evaluation, by taking videos of one subject as testing data, and randomly splitting the sequences from the remaining subjects into training and validation sets. The training set is always composed by sequences of 5 subjects, while the sequences of the remaining 3 subjects are used for validation.

We train a 4 part model with 3 mixture components for each action class. In order to illustrate the learnt models, Fig. 6 shows an example frame from a jack sequence with the corresponding action model component over imposed. Parts are colored in blue, red, green and cyan, and represented as ellipses which illustrate the gaussian distribution of the feature relative positions. Static features are represented by crosses and motion features by diamonds. Each feature has been colored with the color of its corresponding parent. Features in yellow and magenta were assigned to the background. Further examples from all classes are shown in Fig. 10.


(a) bend (b) jack (c) jump

(d) pjump (e) run (f) side

(g) walk (h) wave1 (i) wave2

Figure 4. **Human actions dataset:** Example frames from video sequences in the dataset from [2]. The dataset contains 83 videos from 9 different human action classes.

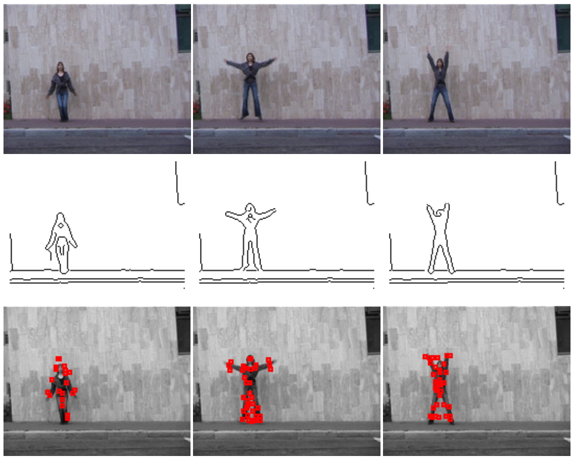


Figure 5. **Feature detection:** The first row contains example frames from a training sequence. The edge maps shown in the second row are obtained using the Canny edge detector. The third row illustrates the spatial-temporal interest point detection. The motion features are obtained using the method in [8]. The figure is best viewed in color.

We investigated the performance of our method in frame-by-frame classification, as well as video classification using the voting scheme presented above. The confusion tables are shown in Fig. 7 and Fig. 8. When classifying entire sequences, our system can correctly categorize 72.8% of the testing videos. Note that the confusions are reasonable in

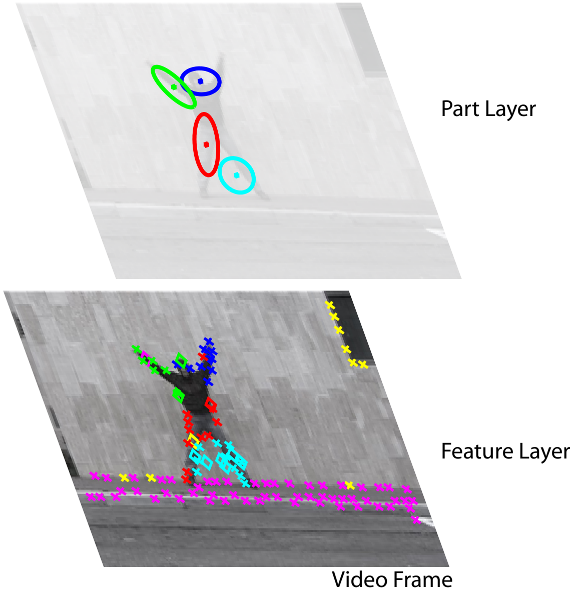
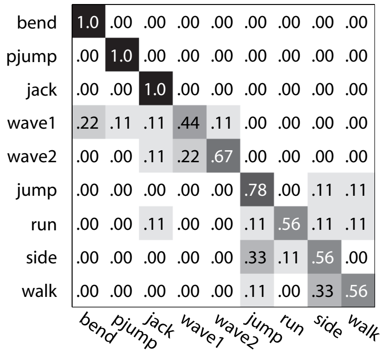


Figure 6. **Action model over a testing frame:** The figure shows our hierarchical model imposed over a testing frame. Parts are represented as ellipses, which illustrate the distribution of the relative position of their children features. Static features are drawn as crosses, while motion features as diamonds. The color of the feature indicates the feature parent. Features in magenta and yellow belong to the background. The figure is best viewed in color.

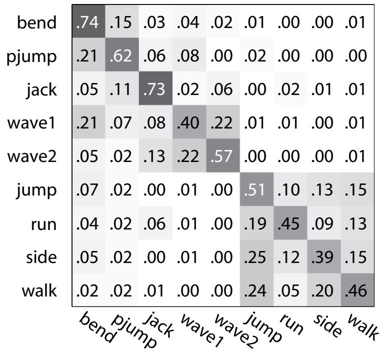


	bend	pjump	jack	wave1	wave2	jump	run	side	walk
bend	1.0	.00	.00	.00	.00	.00	.00	.00	.00
pjump	.00	1.0	.00	.00	.00	.00	.00	.00	.00
jack	.00	.00	1.0	.00	.00	.00	.00	.00	.00
wave1	.22	.11	.11	.44	.11	.00	.00	.00	.00
wave2	.00	.00	.11	.22	.67	.00	.00	.00	.00
jump	.00	.00	.00	.00	.00	.78	.00	.11	.11
run	.00	.00	.11	.00	.00	.11	.56	.11	.11
side	.00	.00	.00	.00	.00	.33	.11	.56	.00
walk	.00	.00	.00	.00	.00	.11	.00	.33	.56

Figure 7. **Video Classification:** Horizontal lines are ground truth, and vertical columns are predicted labels. The table summarizes the result of 9 runs in a leave-one-out procedure. The system correctly classifies 72.8% of the testing sequences.

the sense that most of the time missclassification occurs between very similar motions, for instance there is confusion between wave1, wave2 and jacks, as well as confusion between run, walk, side and jump (please refer to Fig. 4).

In order to evaluate the contribution of the hierarchical model, as well as the use of dynamic and static features, we perform several control experiments. For this purpose, we randomly select one subject and use the corresponding sequences as the testing set. The videos from the remaining subjects are randomly split into training and validation sets.



	bend	pjump	jack	wave1	wave2	jump	run	side	walk
bend	.74	.15	.03	.04	.02	.01	.00	.00	.01
pjump	.21	.62	.06	.08	.00	.02	.00	.00	.00
jack	.05	.11	.73	.02	.06	.00	.02	.01	.01
wave1	.21	.07	.08	.40	.22	.01	.01	.00	.01
wave2	.05	.02	.13	.22	.57	.00	.00	.00	.01
jump	.07	.02	.00	.01	.00	.51	.10	.13	.15
run	.04	.02	.06	.01	.00	.19	.45	.09	.13
side	.05	.02	.00	.01	.00	.25	.12	.39	.15
walk	.02	.02	.01	.00	.00	.24	.05	.20	.46

Figure 8. **Frame-by-frame classification:** Horizontal lines are ground truth, and vertical columns are predicted labels. The tables are the average over 9 runs in a leave-one-out procedure. In average, the algorithm assigns the correct label to 55.0% of the testing frames.

We evaluate the contribution of the mixture of hierarchical models by comparing it to a one component hierarchical model and a bag of keypoints model. We believe that a class of human action (*e.g.* walking) can be represented by a small number of distinctive (static or dynamic) poses. We have therefore chosen a mixture of models to represent each action. In order to show that this representation is more powerful than a single component, we have trained 1-component models for each action class. Additionally, to demonstrate that including geometric information is useful, we train bag of keypoints models for each action class. For this purpose each sequence is represented as a histogram of static and dynamic features. The training examples are kept in a database and new video frames are classified using a nearest neighbor procedure. The bar plot on the left in Fig 9 shows the comparison of the performance of each model under the described settings. The outcome of this experiment supports the intuition that human actions contain certain multimodality which can be better represented by a mixture of hierarchical models. The inclusion of the constellation layer and the geometric constraints that it encodes is also useful, since ignoring the geometric arrangement of features and adopting a bag of keypoints model produces poorer classification results.

Finally, we also explore the contribution of each feature type into the classification performance. We trained our mixture of hierarchical models using static features only, dynamic features only and also using both types of features. The bar plot on the right in Fig 9 shows the comparison of the performance of the model when using different types of features. These results empirically support the intuition that a combination of both static and motion features provide the best representation for human actions. Additionally, the experiment shows that if one is to choose a single feature type, motion features are preferable; which is also intuitive in the sense that motion features provides a richer representation

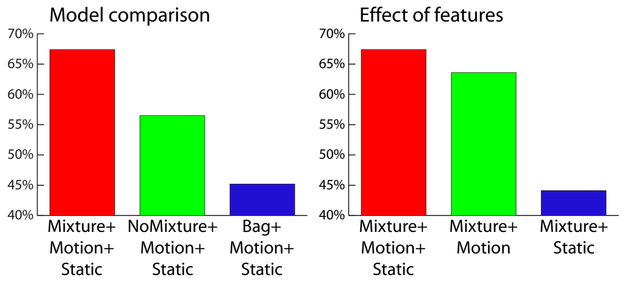


Figure 9. **Effect of model and features:** The plot shows a performance comparison of recognition accuracy under different settings. *Mixture* indicates the use of a 3-component mixture model; *NoMixture* indicates the use of a 1-component model; and *Bag* denotes a pure bag of keypoints model. On the *left*, the plot shows that the employment of our new hierarchical model improves the performance over the use of pure bag-of-features model. Also, using a mixture of models helps to account for the multimodality nature of the action models, therefore better recognition is obtained when compared to the 1-component model. On the *right* plot, the results show that using a combination of static and motion features provides the best description of the human actions, which translates into the best recognition accuracy.

of dynamic events than static features.

The first reported classification results on this dataset appeared on [2]. Their method achieved a classification error rate of 0.39%. It is however, difficult to make a fair comparison. Their method requires a background substraction procedure, global motion compensation, and it cannot take classification decisions frame by frame. Please also note, that our model is general in the sense that it aims to offer a generic framework for human motion and pose categorization.

5. Conclusions

In this paper, we presented a hierarchical model of shape and appearance for human action categorization. The model combines the strong shape representation of the constellation model with the large number of features that utilizes the bag-of-words model. Our constellation-of-bags-of-features model is able to combine static and motion image features in a principled way, as well as perform categorization in a frame-by-frame basis.

Future directions include to adopt robust features that help to account for more general camera motion and unconstrained environments. We believe this model has the potential to be able to characterize more complex motions and configurations of the highly articulated human body.

References

[1] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(4):509–522, 2002.

[2] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. In *ICCV*, 2005.

[3] O. Boiman and M. Irani. Detecting irregularities in images and in video. In *ICCV*, 2005.

[4] G. Bouchard and B. Triggs. Hierarchical part-based visual object categorization. In *CVPR*, 2005.

[5] C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001. Software available online at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

[6] V. Cheung, B. J. Frey, and N. Jojic. Video epitomes. In *CVPR*, 2005.

[7] N. Dalal, B. Triggs, and C. Schmid. Human detection using oriented histograms of flow and appearance. In *ECCV*, 2006.

[8] P. Dollár, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. In *VS-PETS*, 2005.

[9] A. A. Efros, A. C. Berg, G. Mori, and J. Malik. Recognizing action at a distance. In *ICCV*, 2003.

[10] C. Fanti, L. Zelnik-Manor, and P. Perona. Hybrid models for human motion recognition. In *CVPR*, 2005.

[11] X. Feng and P. Perona. Human action recognition by sequence of movelet codewords. In *3DPVT*, pages 717–723, 2002.

[12] K. Grauman and T. Darrell. Unsupervised learning of categories from sets of partially matching image features. In *CVPR*, 2006.

[13] I. Laptev and T. Lindeberg. Velocity adaptation of space-time interest points. In *CVPR*, 2004.

[14] J. C. Niebles, H. Wang, and L. Fei-Fei. Unsupervised learning of human action categories using spatial-temporal words. In *BMVC*, 2006.

[15] D. Ramanan. Learning to parse images of articulated bodies. In *Advances in Neural Information Processing Systems*. 2006.

[16] D. Ramanan and D. A. Forsyth. Automatic annotation of everyday movements. In *Advances in Neural Information Processing Systems*. 2004.

[17] C. Schuldt, I. Laptev, and B. Caputo. Recognizing human actions: A local svm approach. In *ICPR*, 2004.

[18] H. Sidenbladh and M. J. Black. Learning the statistics of people in images and video. *International Journal of Computer Vision*, 54(1-3):183–209, 2003.

[19] J. Sivic, B. C. Russell, A. A. Efros, A. Zisserman, and W. T. Freeman. Discovering objects and their localization in images. In *ICCV*, 2005.

[20] Y. Song, L. Goncalves, and P. Perona. Unsupervised learning of human motion. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(25):1–14, 2003.

[21] M. Weber, M. Welling, and P. Perona. Unsupervised learning of models for recognition. In *ECCV*, 2000.

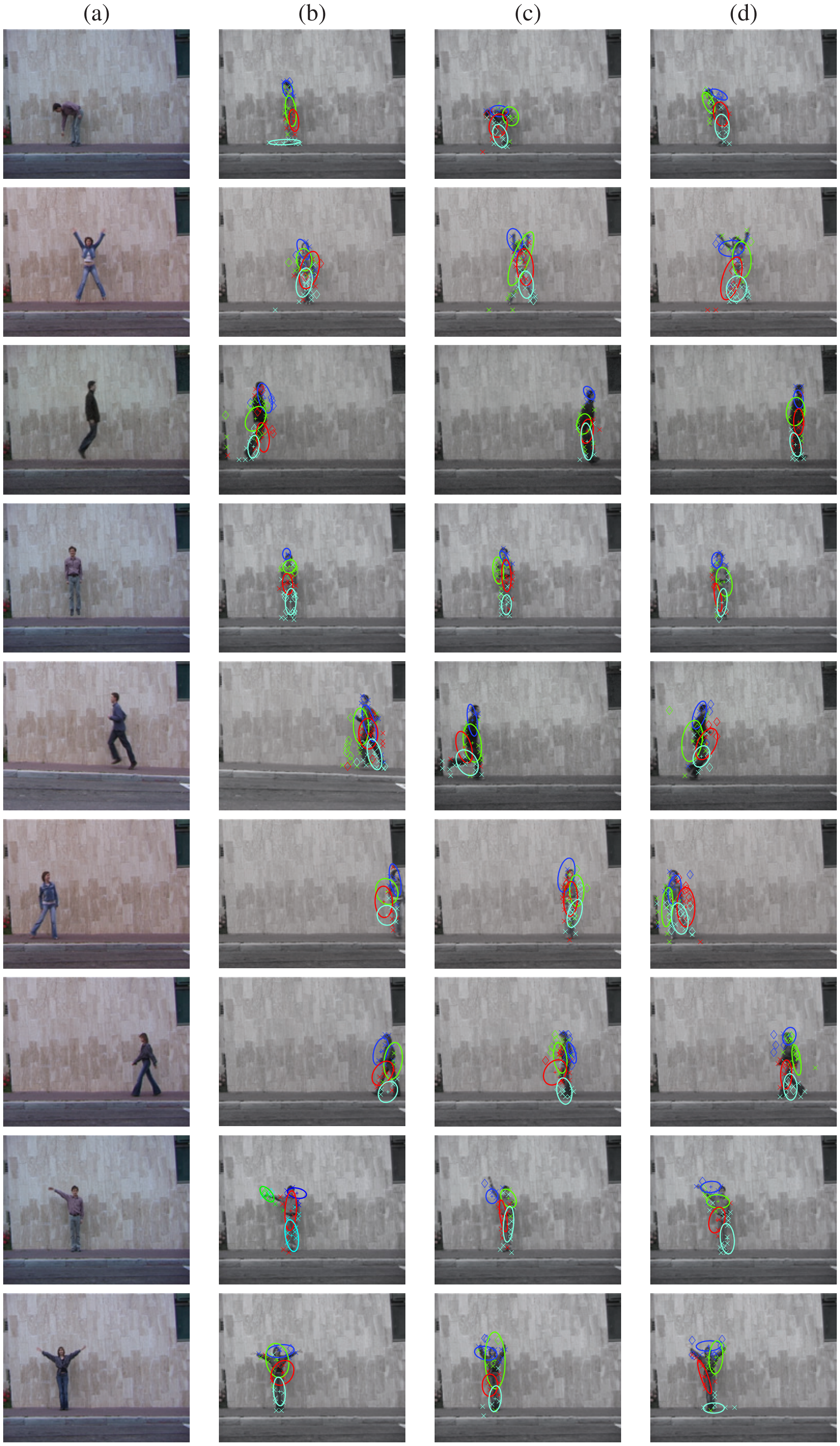


Figure 10. **Learnt action models:** Each row illustrate a different action category: bend, jack, jump, p-jump, run, side, walk, wave1, wave2. Column (a) shows example frames from the original sequence. (b)-(d) show the three mixture components for each action model. Static features are represented by crosses and motion features by diamonds. Each image feature is colored according to its part membership. Ellipses illustrate the variance of the position distributions for each part. The figure is best viewed in color.