Multi-Task Sparse Learning with Beta Process Prior for Action Recognition

Chunfeng Yuan[1], Weiming Hu[1], Guodong Tian[1], Shuang Yang[1], and Haoran Wang[2]

[1]National Laboratory of Pattern Recognition, CASIA, {cfyuan, wmhu, gdtian, syang}@nlpr.ia.ac.cn
[2]School of Automation, Southeast University, China, whr1fighting@gmail.com

Abstract

In this paper, we formulate human action recognition as a novel Multi-Task Sparse Learning(MTSL) framework which aims to construct a test sample with multiple features from as few bases as possible. Learning the sparse representation under each feature modality is considered as a single task in MTSL. Since the tasks are generated from multiple features associated with the same visual input, they are not independent but inter-related. We introduce a Beta process(BP) prior to the hierarchical MTSL model, which efficiently learns a compact dictionary and infers the sparse structure shared across all the tasks. The MTSL model enforces the robustness in coefficient estimation compared with performing each task independently. Besides, the sparseness is achieved via the Beta process formulation rather than the computationally expensive l_1 norm penalty. In terms of non-informative gamma hyper-priors, the sparsity level is totally decided by the data. Finally, the learning problem is solved by Gibbs sampling inference which estimates the full posterior on the model parameters. Experimental results on the KTH and UCF sports datasets demonstrate the effectiveness of the proposed MTSL approach for action recognition.

1. Introduction

Recognition of human actions [1] in videos is an important but challenging task in computer vision. It has many potential applications, such as smart surveillance, human-computer interface, video indexing and browsing, automatic analysis of sports events, and virtual reality. However, it is a challenging task not only because of geometric variations between intra-class objects or actions, but also because of changes in scale, rotation, viewpoint, illumination, and occlusion.

The fusion of multiple features is effective for recognizing actions as the single feature based representation is not enough to capture the visual variations (view-point, illumination etc.). Yuan *et al.* [10] employ color feature, texture feature and shape feature for face recognition and object categorization. Tran *et al.* [12] represent the video sequence as several part motion descriptors and then encode these descriptors by sparse representation for classification. They employ and solve the sparse representation on each part motion descriptor independently, but actually each part is inter-related. Multi-task learning [2] has recently received much attention in the fields of machine learning, and computer vision. It capitalizes on shared information between related tasks to improve the performance of individual tasks, and it has been successfully applied to vision problems such as image classification [10], image annotation [3], and object tracking [4]. In [10][3][4], the l_1 norm is employed for joint regularization. They extend the l_1 framework for learning single sparse model to a setting where the goal is to learn a set of jointly sparse models.

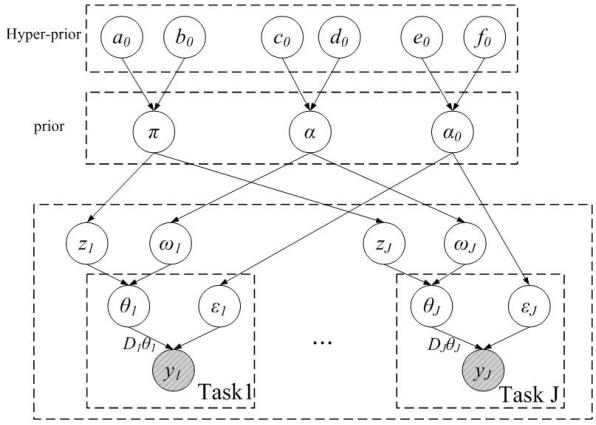

Figure 1. A hierarchical Bayesian model representation of the multi-task sparse learning method. The detailed parameters are introduced in Section 3.

In this paper, motivated by the success of multi-task learning, we propose a novel Multi-Task Sparse Learning(MTSL) model combined with Beta Process Prior for

human action recognition. The proposed MTSL model jointly formulates multiple modalities of features, each feature modality is viewed as a task in MTSL, and the sparsity structures are shared across all the tasks. Moreover, we introduce a Beta process prior to the proposed MTSL model. In terms of the Beta process formulation, the inferred sparsity level is intrinsic to the data, while solving the l_1 norm usually needs assume a reconstructed residual error or the sparsity level. Via setting the common prior for all the task, the sparse structure shared across all the tasks is efficiently inferred which enforces the robustness in coefficient estimation compared with dealing with each task independently. A graphical model representation of the MTSL is illustrated in Figure 1. The bottom layer of this hierarchical model is composed of individual models with task-specific parameters; On the layer above, tasks are connected together via the common prior placed on the tasks; the top layer above is the hyper-prior, invoked on parameters of the prior at the level below.

In this paper, two kinds of action features are used for efficiently describing actions in videos. Based on the popular and efficient bag-of-features model, histogram feature and co-occurrence feature are constructed from local 3D SIFT descriptors. The histogram feature statistically records the local appearance information about the video. The co-occurrence feature exploits the spatio-temporal geometric distribution information about local features in 3D space which is totally ignored in the histogram feature. These two features describe the video sequence at two aspects: local appearance information, and geometric distribution information.

The main contributions of this paper are:

- We propose a multi-task sparse learning method for action recognition, which can efficiently combine multiple features to improve the recognition performance.
- The proposed MTSL is a robust sparse coding method that mines correlations among different tasks to obtain the shared sparsity pattern which is ignored when learning each task individually.
- It uses the non-parametric Bayesian prior instead of the l_1 norm regularization. In this way, the dictionary is learned using the Beta process construction [8], and the sparsity level is decided by the data and one doesn't need to assume the stopping criteria.

1.1. Related Work

In the past few years, sparse representation has been successfully applied to computer vision applications, such as object classification [10], tracking [11], face recognition [5], and action recognition [6]. In [6], the local spatio-temporal descriptor is represented as some linear combination of few dictionary elements via sparse representation instead of the traditional bag-of-features methods that involve clustering and vector quantization. Then one action class or individual training sample is computed by summing the sparse coefficients of all the local descriptors contained. The sparse representation of a local feature is richer and more accurate than quantizing it to a single word via nearest neighbor in the bag-of-features method. However, the manner of applying sparse representation on low-level features will be time consuming due to large members of low-level feature and high computational complexity of solving sparse representation. Guo *et al.* [7] use the log-covariance matrix to represent each video, and employ sparse representation on the log-covariance matrices. Sparse representation is applied on the high level video feature, and the label of a test sample is decided on the reconstruction residual error of the training samples from every class.

In most sparse representation methods, sparsity is measured by the l_1-norm with a Lagrangian constant to balance the reconstruction error and the sparsity constraint. Recently, several variations and extensions of the l_1 minimization have been introduced to them. In these methods, the sparse coefficient is estimated via point estimate (without a posterior distribution), typically based on orthogonal matching pursuits, basis pursuits or related methods, for which the stopping criteria is defined by assuming the reconstructed residual error or the sparsity level. However, in many applications one may not know the residual error or sparsity level. Zhou *et al.* [8] and Ji *et al.* [9] introduce a non-parametric Bayesian formulation to address these problems. In [9] a new multi-task compressive sensing (CS) modeling with Bayesian regression is developed, which addresses a variety of issues that previously have not been addressed: (i) a stopping criterion for determining when a sufficient number of CS measurements have been performed, and (ii) simultaneous inverse of multiple related CS measurements. These methods[8][9] are proved very efficient in image denoising, image inpainting, and compressive sensing. In this paper, motivated by the success of non-parametric Bayesian formulation in CS, we propose a novel Multi-Task Sparse Learning(MTSL) model combined with Beta process prior for human action recognition.

1.2. Organization

The paper is organized as follows. Section 2 introduces the multiple features used for action representation. Section 3 gives a detailed description of the proposed MTSL model. Section 4 reports experimental results on several human action datasets. Section 5 concludes the paper.

2. Multi-feature representations for video context modeling

We employ two features to represent each video sequence: histogram feature, and co-occurrence feature. Lo-

cal interest point features are a popular way to represent videos. They achieve state-of-the-art results for action recognition when combined with the bag-of-features representation. However, the histogram representation ignores the geometric distribution information about the local features. Co-occurrence matrices exploit the spatio-temporal proximity distribution about local features in 3D space to characterize geometric context of action class.

Histogram Features: We first perform the spatio-temporal interest point detection for each video sequence. The Harris3D detector [13] is employed to detect spatio-temporal interest points at each frame, and the 3D SIFT descriptor [14] is used to describe the cuboid extracted at each interest point. Afterwards, 3D SIFT descriptors from the training videos are quantized to form a visual vocabulary by using the k-means clustering method. The histogram feature for each video sequence is namely the statical histogram of SIFT descriptors extracted from it.

co-occurrence feature: It is a co-occurrence matrix of visual words for capturing the geometric information. Let V denote a video, which is described as $\{(\boldsymbol{l}_i, f_i)\}1 \leq i \leq M$, where $\boldsymbol{l}_i$ is the spatio-temporal position vector of the ith detected local feature and f_i is its visual word index. M is the total number of the local features detected in the video. Spatio-temporal co-occurrence matrix is defined as $O = (o_{ij}) \in R^{K \times K}$ with each element as

$$O(i,j) = o_{ij} = \sharp\{(f_h, f_m) | f_h = i, f_m = j, \|\boldsymbol{l}_h - \boldsymbol{l}_m\| \leq d\} \quad (1)$$

where f_l and f_m are a pair of neighboring features with the distance not larger than d, and the $\sharp$ means the number of feature pairs satisfying all the conditions listed in the brackets in Eq.(1).

3. Multi-Task Sparse Learning for Action Recognition

The proposed multi-task sparse modeling method makes use of multiple modalities of features for action representation. We generate J tasks from J different modalities of features associated with the same input video sequences. Learning the sparse representation of the video in one feature space is viewed as an individual task. The multi-task learning shares information between related tasks to improve the performance of each individual task.

3.1. Multi-Task Sparse Learning with Beta Process Formulation

Let $\boldsymbol{y}_j \in \mathbb{R}^{m_j}, j = 1, ..., J$ represent a target sample with the J tasks, where m_j is the dimensionality of the jth modality of feature. Each vector $\boldsymbol{y}_j$ employs a dictionary matrix $D_j \in \mathbb{R}^{m_j \times K}$ and is represented as

$$\boldsymbol{y}_j = D_j \boldsymbol{\theta}_j + \boldsymbol{\epsilon}_j, j = 1, ..., J \quad (2)$$

where $\boldsymbol{\theta}_j$ is the set of sparse transform coefficients and $\boldsymbol{\epsilon}_j$ is a residual error vector associated with task j. $\boldsymbol{\epsilon}_j \in \mathbb{R}^{m_j}$ is modeled as m_j *i.i.d.* draws from a zero-mean Gaussian distribution with an unknown precision α_0 (variance $1/\alpha_0$).

The feature representations $\boldsymbol{y}_j, j = 1, ..., J$ are extracted from the same test video sequence, and $D_j, j = 1, ..., J$ are computed from the features of the same training videos. Thus, the sparse coefficients $\boldsymbol{\theta}_j$ for different tasks $j = 1, ..., J$ are related. Therefore, we impose that $\boldsymbol{\theta}_j \in \mathbb{R}^K$ is sparse and $\boldsymbol{\theta}_j, j = 1, .., J$ are drawn from the common prior. We set $\boldsymbol{\theta}_j = \boldsymbol{z}_j \odot \boldsymbol{\omega}_j$, where $\odot$ represents element-wise multiplication of two vectors, $\boldsymbol{z}_j$ is a binary vector defining which members of the dictionary $\boldsymbol{d}_{j,k}$ are used to represent the sample $\boldsymbol{y}_j$, and $\boldsymbol{\omega}_j \sim \mathcal{N}(0, \alpha^{-1} I_K)$ is a weight vector with the precision α. $\boldsymbol{\omega}_j$ is introduced because the reconstruction coefficients of the dictionary are not always binary.

We employ a Beta process to formulate the dictionary D_j and the K-dimensional binary vector $\boldsymbol{z}_j$. The stick-breaking construction of a Beta process is represented as:

$$\begin{aligned} G(\boldsymbol{x}) &= \sum_{k=1}^{K} \pi_k \delta_{\boldsymbol{x}_k}(\boldsymbol{x}) \\ \pi_k &\sim Beta(a_0/K, b_0(K-1)/K) \\ \boldsymbol{x}_k &\sim G_0 \ , \end{aligned} \quad (3)$$

where the $\boldsymbol{x}_k$ is the atom distributed according to G_0, and π_k is the "stick-breaking weight" depending on the parameters a_0 and b_0. For our problem, the candidate members $\boldsymbol{d}_{j,k}$ of our dictionary D_j correspond to the atoms $\boldsymbol{x}_k$, and the kth component of $\boldsymbol{z}_j$ is drawn $z_{j,k} \sim Bernoulli(\pi_k)$. Therefore, the multi-task sparse representation model is expressed as:

$$\begin{aligned} \boldsymbol{y}_j \mid \boldsymbol{\theta}_j, \alpha_0 &\sim \mathcal{N}(D_j \boldsymbol{\theta}_j, \alpha_0^{-1} I_{m_j}), j = 1, ..., J \\ D_j &= [\boldsymbol{d}_{j,1}, ..., \boldsymbol{d}_{j,K}] \\ \boldsymbol{d}_{j,k} &\sim \mathcal{N}(0, m_j^{-1} I_{m_j}) \\ \boldsymbol{\epsilon}_j &\sim \mathcal{N}(0, \alpha_0^{-1} I_{m_j}) \\ \boldsymbol{\theta}_j &= \boldsymbol{z}_j \odot \boldsymbol{\omega}_j \\ \boldsymbol{\omega}_j &\sim \mathcal{N}(0, \alpha^{-1} I_K) \\ \boldsymbol{z}_j \mid \{\pi_k\}_{k=1,K} &\sim \prod_{k=1}^{K} Bernoulli(\pi_k) \\ \pi_k &\sim Beta(a_0/K, b_0(K-1)/K) \\ \alpha &\sim \Gamma(c_0, d_0) \\ \alpha_0 &\sim \Gamma(e_0, f_0) \ . \end{aligned} \quad (4)$$

Non-informative gamma hyper-priors are placed on α and α_0. With these parametric definitions, a graphical model of multi-task sparse representation is illustrated in Figure 1. In this framework, the data from all J tasks are used to jointly infer the priors $\boldsymbol{\pi}$, α, and α_0. The constraint of joint sparsity across different tasks is valuable since different tasks may favor different sparse reconstruction coefficients, yet

the joint sparsity may enforce the robustness in coefficient estimation. Besides, given the priors, each task is learned independently. As a result, the estimation of a task is affected by both its own training data and the other tasks via the common priors.

The consecutive variables in the proposed model are in the conjugate exponential family, and therefore the inference could be implemented via Gibbs sampling analysis.

3.2. Inference

In the proposed model, the variables D, Z, $\boldsymbol{\omega}$, $\boldsymbol{\pi}$, α, and α_0 need to be inferred given the training samples. Gibbs sampling inference is used to update them iteratively. In each step, one of the variables is sampled from its posterior given all other variables and training samples.

In the beginning, we initialize all the variables. Except the dictionary D, all the other variables are initialized randomly. Let D_j denote the dictionary associated with the jth task. We initialize D_j via K-SVD [15]. K-SVD is a method to learn an over-complete dictionary for sparse representation. For every task and every action class, we obtain an initial dictionary $D_{j,c}$ of a large size K_c by K-SVD from the training samples $X_{j,c}$, where $X_{j,c}$ represents the training feature matrix associated with the cth class and the jth task. Then, we obtain an initial dictionary $D_j = [D_{j,1}, ..., D_{j,C}]$, where $\sum_{c=1}^{C} K_c = K$ is the total number of bases.

Let $Y = [\boldsymbol{y}_1, ..., \boldsymbol{y}_J]$ be a sample associated with the J task. The full likelihood of the proposed model is expressed as

$$\begin{aligned} &p(Y, D, Z, \boldsymbol{\omega}, \boldsymbol{\pi}, \alpha, \alpha_0) \\ &= \prod_{j=1}^{J} \mathcal{N}(\boldsymbol{y}_j; D_j(\boldsymbol{z}_j \odot \boldsymbol{\omega}_j), \alpha_0^{-1} I_{m_j}) \mathcal{N}(\boldsymbol{\omega}_j; 0, \alpha^{-1} I_K) \\ &\quad \prod_{j=1}^{J} \prod_{k=1}^{K} \mathcal{N}(\boldsymbol{d}_{j,k}; 0, m_j^{-1} I_{m_j}) Bernoulli(z_{j,k}; \pi_k) \\ &\quad \prod_{k=1}^{K} Beta(\pi_k; a_0, b_0) \\ &\quad \Gamma(\alpha; c_0, d_0) \Gamma(\alpha_0; e_0, f_0) \,. \end{aligned} \tag{5}$$

This model yields a Gibbs sampling scheme for posterior sampling given observations Y. The variables D, Z, $\boldsymbol{\omega}$, $\boldsymbol{\pi}$, α, and α_0 are sampled as follows.

A. *Sampling* $D_j = [\boldsymbol{d}_{j,1}, \boldsymbol{d}_{j,2}, ..., \boldsymbol{d}_{j,K}]$

The posterior probability of $\boldsymbol{d}_{j,k}$ is expressed as

$$p(\boldsymbol{d}_{j,k} \mid -) \propto \mathcal{N}(\boldsymbol{y}_j; D_j(\boldsymbol{z}_j \odot \boldsymbol{\omega}_j), \alpha_0^{-1} I_{m_j}) \mathcal{N}(\boldsymbol{d}_{j,k}; 0, m_j^{-1} I_{m_j}) \tag{6}$$

B. *Sampling* $\boldsymbol{z}_j = [z_{j,1}, z_{j,2}, ..., z_{j,K}]$

Given the observed test data $\boldsymbol{y}_j$, the likelihood function for the variables $\boldsymbol{z}_j$, $\boldsymbol{\omega}_j$ and α_0 is expressed as

$$p(\boldsymbol{y}_j \mid \boldsymbol{z}_j, \boldsymbol{\omega}_j, \alpha_0) = (2\pi/\alpha_0)^{-m_j/2} exp(-\frac{\alpha_0}{2} \|\boldsymbol{y}_j - D_j(\boldsymbol{z}_j \odot \boldsymbol{\omega}_j)\|_2^2) \tag{7}$$

By applying the Bayes' rule, the posterior density function on $z_{j,k}$ is as follow

$$p(z_{j,k} \mid -) \propto \mathcal{N}(\boldsymbol{y}_j; D_j(\boldsymbol{z}_j \odot \boldsymbol{\omega}_j), \alpha_0^{-1} I_{m_j}) Bernoulli(z_{j,k}; \pi_k) \tag{8}$$

C. *Sampling* $\boldsymbol{\omega}_j = [\omega_{j,1}, \omega_{j,2}, ..., \omega_{j,K}]$

The posterior density function on $\omega_{j,k}$ is as follow

$$p(\omega_{j,k} \mid -) \propto \mathcal{N}(\boldsymbol{y}_j; D_j(\boldsymbol{z}_j \odot \boldsymbol{\omega}_j), \alpha_0^{-1} I_{m_j}) \mathcal{N}(\boldsymbol{\omega}_j; 0, \alpha^{-1} I_K) \tag{9}$$

D. *Sampling* π_k

The posterior density function on π_k is as follow

$$p(\pi_k \mid -) \propto Beta(\pi_k; a_0, b_0) \prod_{j=1}^{J} Bernoulli(z_{j,k}; \pi_k) \tag{10}$$

E. *Sampling* α

The posterior density function on α is as follow

$$p(\alpha \mid -) \propto \Gamma(\alpha; c_0, d_0) \prod_{j=1}^{J} \mathcal{N}(\boldsymbol{\omega}_j; 0, \alpha^{-1} I_K) \tag{11}$$

F. *Sampling* α_0

The posterior density function on α_0 is as follow

$$p(\alpha_0 \mid -) \propto \Gamma(\alpha_0; e_0, f_0) \prod_{j=1}^{J} \mathcal{N}(\boldsymbol{y}_j; D_j(\boldsymbol{z}_j \odot \boldsymbol{\omega}_j), \alpha_0^{-1} I_{m_j}) \tag{12}$$

3.3. Classification Rule

Ideally, if a test sample belongs to action class c, the non-zero components in the estimated coefficient vector $\hat{\boldsymbol{\theta}}_j$ will be associated with multiple columns of $\hat{D}_{j,c}$ from individual action class c. Here, $\hat{D}_{j,c}$ is a more compact and discriminative dictionary learned by the above MTSL model from the initial $D_{j,c}$. Therefore, we classify $\boldsymbol{y}_j$ based on how well $\boldsymbol{y}_j$ is reproduced by the coefficients associated with the learned dictionary of each individual class. For each task j, we define

$$\begin{aligned} \hat{\boldsymbol{\theta}}_j &= [\hat{\theta}_{j,1}, ..., \hat{\theta}_{j,K_1}, ..., \hat{\theta}_{j,\sum_{c=1}^{C-1} K_c + 1}, ..., \hat{\theta}_{j,K}] \\ &= [\hat{\boldsymbol{\theta}}_j^1, ..., \hat{\boldsymbol{\theta}}_j^C] \,. \end{aligned} \tag{13}$$

Using the coefficients $\hat{\boldsymbol{\theta}}_j^c$ associated with the action class c, the jth modality $\boldsymbol{y}_j$ of a test sample is reproduced as $\hat{\boldsymbol{y}}_j = \hat{D}_{j,c} \hat{\boldsymbol{\theta}}_j^c$. Then, we classify the test sample to belong to the class with the lowest total reconstruction error accumulated over all the J tasks:

$$c^* = \arg\min_c \sum_{j=1}^{J} \left\| \boldsymbol{y}_j - \hat{D}_{j,c} \hat{\boldsymbol{\theta}}_j^c \right\|_2^2. \tag{14}$$

Figure 2. Sample frames from video sequences of the KTH dataset(top), and the UCF sports dataset (bottom).

Algorithm 1 : Multi-Task Sparse Learning for Action Recognition

Dictionary Learning Phase:
Input: training samples of jth task and cth class $X_{j,c}$
hyper-parameters $a_0, b_0, c_0, d_0, e_0, f_0$
1.initialize dictionary $D_{j,c}$ via the K-SVD of $X_{j,c}$
2.$X_{j,c} = D_{j,c}\boldsymbol{\theta}_{j,c} + \boldsymbol{\epsilon}_{j,c}, j = 1, ..., J$
3.for iteration
  sample $\boldsymbol{d}_{j,k}$
  sample $\boldsymbol{z}_j, \boldsymbol{\omega}_j, \pi_k$
  sample α, α_0
 end iteration
Output: $\hat{D}_j = [\hat{D}_{j,1}, ..., \hat{D}_{j,C}], j = 1, ..., J.$
Classification Phase:
Input:$\hat{D}_j, \boldsymbol{y}_j$
1. $\boldsymbol{y}_j = \hat{D}_j\boldsymbol{\theta}_j + \boldsymbol{\epsilon}_j, j = 1, ..., J$
2.for iteration
  sample $\boldsymbol{z}_j, \boldsymbol{\omega}_j$
 end iteration
3. compute $\hat{\boldsymbol{\theta}}_j = \boldsymbol{z}_j \odot \boldsymbol{\omega}_j$
4. compute the label c^* via Eq.(14).
Output: label c^*

Algorithm 1 summaries the details of the optimization and classification procedure of our multi-task sparse learning model for classification. In the dictionary learning phase, we first obtain an initial dictionary via K-SVD for every task and every action class. Then the proposed MTSL method is used to obtain a compact and discriminative dictionary from the initial dictionary by Gibbs sampling. In the classification phase, the sparse representation of a test sample is achieved by the MTSL model based on the learned dictionary. At last, the test video is classified by Eq.(14).

4. Experiments

We tested our approach on three human action datasets: the KTH [17] and UCF sports [18]. Several samples from these three datasets are shown in Figure 2. We first extracted two features (histogram, and co-occurrence) on every dataset. For the histogram feature and the co-occurrence feature, we fix the number of visual words in the vocabulary of 3D SIFT descriptors to 500 on every dataset. For the Gamma prior on the sparse common prior α ,we set $c_0 = d_0 = 0$ as a default choice which avoids subjective choice and leads to computational simplifications. For the Gamma prior on the noise precision α_0, we also let $e_0 = f_0 = 0$ as a default choice [9].

The KTH video database contains six types of human actions (walking, jogging, running, boxing, hand waving and hand clapping) performed by 25 subjects in four different scenarios. There are totally 599 sequences in the dataset. We performed leave-one-person-out cross-validation to make the performance evaluation. In each run, 24 actors' videos are used as the training set and the remaining one person's videos as the test set. The final results are the average of 25 times runs.

The UCF sports dataset consists of 150 action videos including 10 sport actions, diving, golf swinging, kicking, weightlifting, horseback riding, running, skating, swinging bench, swinging from side angle and walking. It collects a natural pool of actions featured in a wide range of scenes and viewpoints, and in unconstrained environments. The UCF sports database was tested in a leave-one-out manner, cycling each example in as a test video one at a time, following [18] [24] [22].

4.1. Evaluation of the Proposed Method

In order to evaluate the overall performances of our proposed algorithms, we performed two groups of comparison experiments: single feature based methods and feature combination methods. In the first group, the proposed MTSL method handled one single feature by setting the number of tasks $J = 1$, and it is symbolized by "Our-ST" in Tables 1, and 2. We compared it with other three methods:

- The SVM method in which the χ^2 distances between histogram features or co-ocurrent features are incorporated into the radial basis function as the kernel of SVM classifier.

- Sparse representation classification based on the l_1 regularization, symbolized by "L1-SRC" in the tables.

Table 1. The comparison accuracy (%) performance of single features and feature combination methods on the KTH dataset.

Features	SVM	L1-SRC	L1SRC-DL	Our-ST
Histogram	88.89	82.29	94.44	**95.14**
Occurrence	90.63	74.65	**90.70**	**90.70**

(a) Single features

Actions	CF-SVM	ST-SRC	Our-MTSL
boxing	97.92	97.92	**98.96**
hand clapping	98.96	100	**100**
hand waving	94.79	97.92	**100**
jogging	85.42	**91.67**	89.58
running	86.46	90.63	**93.75**
walking	97.92	98.96	**100**
Average	93.58	96.18	**97.05**

(b) Feature combination methods

Table 2. The comparison accuracy (%) performance of single features and feature combination methods on the UCF sports dataset.

Features	SVM	L1SRC-DL	Our-ST
Histogram	80.67	**90.67**	**90.67**
Occurrence	83.3	**85.33**	**85.33**

(a) Single features

Actions	CF-SVM	ST-SRC	Our-MTSL
diving	92.86	**100**	**100**
golf swinging	**89.89**	61.11	77.78
kicking	**100**	90	85
weight lifting	**100**	**100**	**100**
horseback riding	66.67	91.67	**100**
running	61.54	**92.31**	**92.31**
skating	66.67	**91.67**	**91.67**
swinging bench	95	**100**	**100**
swinging from side	84.62	**100**	**100**
walking	86.36	**90.91**	**90.91**
Average	85.33	90.67	**92.67**

(b) Feature combination methods

The bases of dictionary are composed by training samples.

- The l_1 regularization based sparse representation classification. However, the dictionary is learned by the proposed Beta process formulation as summarized in Algorithm 1. This method is symbolized by "L1SRC-DL" in the tables.

In the second group, we compared the propose Multi-Task Sparse Learning method(MTSL) with other three fusion methods:

- Feature combination based on concatenating all the features into a long feature vector. The SVM classification method is employed on the resulting long feature vector. This method is symbolized by "CF-SVM" in the tables.

- Feature combination based on concatenating all the features into a long feature vector. Moreover, our sparse representation classification under $J = 1$ was performed on the resulting long feature vector. This method is symbolized by "CF-SRC" in the tables.

- Feature combination based on independent single task sparse representation. This method can be viewed as a simplification of our method without enforcing the joint sparsity across tasks. For each feature, we still employed the same non-parameter Bayesian to solve the single task sparse representation. The final classification is based on the accumulation of all the single task sparse representation. This method is symbolized by "ST-SRC" in the tables.

Table 3. Comparison of our approach with state-of-the-art approaches on the KTH and UCF sports datasets.

	Years	KTH	UCF
Yeffet *et al.* [22]	2009	90.1	79.2
Wang *et al.* [23]	2009	92.1	85.6
Kovashka *et al.* [24]	2010	94.53	87.27
Le *et al.* [25]	2011	93.9	86.5
Wang *et al.* [16]	2011	94.2	88.2
OHara *et al.* [20]	2012	**97.9**	91.32
Wang *et al.* [21]	2012	79.8	-
Raptis *et al.* [19]	2012	-	79.4
Our approach		97.05	**92.67**

On the two datasets, we performed the two groups of comparison experiments as mentioned above. Accuracies from our proposed method with other methods for single features and feature combination on the KTH dataset are listed in Table 1. Table 1(a) shows the results on single features. It is observed that the l_1 regularized sparse representation method "L1-SRC" performs worse than SVM on both features, but "L1-SRC" combined with our dictionary learning method, namely "L1SRC-DL", high improves the performance and achieves better results than SVM. It proves that our dictionary learning method can obtain a more discriminate dictionary to improve the performance. The feature combination results are listed in Table 1(b), from which we can see that all feature combination methods improve the recognition accuracy and our approach achieves the best results. Our approach achieves the best recognition performances for five actions of six on the KTH dataset, and its average accuracy is 3.45% higher than the feature-level combination method "CF-SVM". It is also better than "ST-SRC", which demonstrates that multi-task sparse represen-

tation by considering joint sparsity improves the performance compared with the single task sparse representation.

Table 2 lists the accuracies of our proposed method compared with other methods on the UCF sports dataset. In Table 2(a), our method and the l_1 sparse representation classification combined with dictionary learning all outperform the SVM classifier method. It sufficiently demonstrates the effectiveness of our classification method and dictionary learning method. As listed in Table 2(b), our method achieves 92.67% accuracy, and best results on eight actions of total ten ones on the UCF sports dataset.

4.2. Comparison to the State of the Art

Table 3 compares our results to the state-of-the-art methods on the KTH and UCF sports datasets. We achieved 97.05% which is comparable to the state of the art, i.e., 97.9% [20]. We report 92.67% on the UCF sports dataset which is an improvement of 1.35% over [20].

5. Conclusion

In this paper we have presented a new multi-task sparse learning algorithm with a non-parametric Beyesian hierarchy for visual feature combination. It has shown the following properties. (i) By Beta process formulation, the vector of reconstruction coefficients is sparse; this imposition of sparseness is distinct from the widely used l_1 regularized sparseness, in which many coefficients are small but not exactly zero. (ii) In this framework, the data from all tasks are used to jointly infer the priors; given the priors, individual task is learned independently; the estimation of a task is affected by both its own training data and the other tasks via the common priors. (iii) In terms of the non-informative gamma hyper-priors, the sparsity level is totally decided on the data; while in the l_1 regularized sparse representation the stopping criteria is defined by assuming the reconstructed residual error or the sparsity level. Experimental results on the KTH and UCF sports datasets have demonstrated that the proposed multi-task sparse learning method is an effective and efficient way to fuse the complementary features for improving the overall classification performance.

Acknowledgement

This work is partly supported by NSFC (Grant No. 60935002, 61100099), the National 863 High-Tech R&D Program of China (Grant No. 2012AA012504), the Natural Science Foundation of Beijing (Grant No. 4121003), and The Project Supported by Guangdong Natural Science Foundation (Grant No. S2012020011081).

References

[1] R. Poppe. A survey on vision-based human action recognition. *Image and Vision Computing*, Vol.28, No.6, pp.976-990, 2010.

[2] A. Argyriou, T. Evgeniou, and M. Pontil. Convex multi-task feature learning. *Machine Learning*, Vol.73, No.3, pp.243-272, 2008.

[3] A. Quattoni, X. Carreras, M. Collins, and T. Darrell. An efficient projection for l 1,infinity regularization. In *ICML*, pp.857C864, 2009.

[4] T. Zhang, B. Ghanem, S. Liu, N. Ahuja. Robust Visual Tracking via Multi-Task Sparse Learning. *CVPR*, 2012.

[5] L. Zhang, P. Zhu, Q. Hu, and D. Zhang. A Linear Subspace Learning Approach via Sparse Coding. *ICCV*, 2011.

[6] T. Guha, and R.K. Ward. Learning Sparse Representations for Human Action Recognition. *PAMI*, Vol.34, No.8, pp.1576-1588, 2012.

[7] K. Guo, P. Ishwar, and J. Konrad. Action Recognition in Video by Sparse Representation on Covariance Manifolds of Silhouette Tunnels. *ICPR* Contests, pp.294-305, 2010.

[8] M. Zhou, H. Chen, J. Paisley, L. Ren, G. Sapiro, and L. Carin. Nonparametric Bayesian Dictionary Learning for Sparse Image Representations. In *NIPS*, 2009.

[9] S. Ji, D. Dunson, and L. Carin. Multi-Task Compressive Sensing. *IEEE Trans. Signal Processing*, Vol. 57, No. 1, pp. 92-106, 2009.

[10] X. Yuan, and S. Yan. Visual Classification with Multi-Task Joint Sparse Representation. In *CVPR*, 2010.

[11] X. Mei and H. Ling. Robust Visual Tracking and Vehicle Classification via Sparse Representation. *PAMI*, Vol.33, No.11, pp.2259-2272, 2011.

[12] K.N. Tran, I.A. Kakadiaris, and S.K. Shah. Part-based motion descriptor image for human action recognition. *Pattern Recognition*, Vol.45, pp.2562-2572, 2012.

[13] I. Laptev. On space-time interest points. *IJCV*, Vol.64, No.2, pp.107-123, 2005.

[14] P. Scovanner, S. Ali, M. Shah, "A 3-dimensional sift descriptor and its application to action recognition," In Proc. *ACM Multimedia*, pp. 357-360, 2007.

[15] Q. Qiu, Z. Jiang, and R. Chellappa. Sparse Dictionary-based Representation and Recognition of Action Attributes. In *ICCV*, 2011.

[16] H. Wang, A. Kläser, I. Laptev, C. Schmid, C. L. Liu. Action Recognition by Dense Trajectories. In *CVPR*, pp. 3169-3176, 2011.

[17] C. Schuldt, I. Laptev, and B. Caputo. Recognizing Human Actions: A Local SVM Approach. In *ICPR*, pp.32-36, 2004.

[18] M. D. Rodriguez, J. Ahmed, and M. Shah. Action MACH: a spatiotemporal maximum average correlation height filter for action recognition. In *CVPR*, 2008.

[19] M. Raptis, I. Kokkinos, and S. Soatto. Discovering Discriminative Action Parts from Mid-Level Video Representations. In *CVPR*, 2012.

[20] S. OHara, and B. A. Draper. Scalable Action Recognition with a Subspace Forest. In *CVPR*, 2012.

[21] S. Wang, Y. Yang, Z. Ma, X. Li, C. Pang, and A. G. Hauptmann. Action Recognition by Exploring Data Distribution and Feature Correlation. In *CVPR*, 2012.

[22] L. Yeffet, and L.Wolf. Local trinary patterns for human action recognition. In *ICCV*, 2009.

[23] H. Wang, M. M. Ullah, A. Kläser, I. Laptev, C. Schmid. Evaluation of local spatio-temporal features for action recognition. In *BMVC*, 2009.

[24] A. Kovashka, and K. Grauman. Learning a Hierarchy of Discriminative Space-Time Neighborhood Features for Human Action Recognition. In *CVPR*, 2010.

[25] Q. V. Le, W. Y. Zou, S. Y. Yeung, and A. Y. Ng. Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis. In *CVPR*, pp. 3361-3368, 2011.