Feature Selection for Multimedia Analysis by Sharing Information among Multiple Tasks

Yi Yang, Zhigang Ma, Alexander G. Hauptmann, and Nicu Sebe

Abstract—While much progress has been made to multi-task classification and subspace learning, multi-task feature selection has long been largely unaddressed. In this paper, we propose a new multi-task feature selection algorithm and apply it to multimedia (*e.g.*, video and image) analysis. Instead of evaluating the importance of each feature individually, our algorithm selects features in batch mode, by which the feature correlation is considered. While feature selection has received much research attention, less effort has been made on improving the performance of feature selection by leveraging the shared knowledge from multiple related tasks. Our algorithm builds upon the assumption that different related tasks have common structures. Multiple feature selection functions of different tasks are simultaneously learned in a joint framework, which enables our algorithm to utilize the common knowledge of multiple tasks as supplementary information to facilitate decision making. An efficient iterative algorithm is proposed to optimize it, whose convergence is guaranteed. Experiments on different databases have demonstrated the effectiveness of the proposed algorithm.

Index Terms—multitask feature selection, action recognition, image classification, 3D motion data annotation.

1 Introduction and Related Work

Multimedia content analysis and understanding is a fundamental research problem. Generally, there are two ways to improve the performance. The first one is designing discriminative multimedia representation methods, such as new features, *e.g.*, SIFT [32], intermediate representation [13], and combining multiple modalities for representation [37]. The other frequently used method is to utilize appropriate machine learning techniques for a better multimedia analysis [6], [34].

Feature selection aims to reduce redundancy and noise in the original feature set. It has been shown to be a powerful tool in a variety of areas, including multimedia analysis, pattern recognition, computer vision, information retrieval, etc. [12], [17], [23], [35]. Previous research efforts have demonstrated that higher accuracy can be obtained provided that a subset of features is appropriately selected from the whole feature set [18], [20], [22], [27], [31]. In some other cases, although the accuracy of multimedia analysis may not be necessarily improved, the computational efficiency can be boosted because only a subset of the original features is used.

According to whether the class labels of training data are available, feature selection algorithms can be roughly grouped into two families, *i.e.*, supervised feature selection and unsupervised feature selection. Generally speaking, supervised feature selection usually yields better and more reliable performance, mainly because of the utilization of class labels. Given sufficient labeled data, it is possible for supervised algorithms to train appropriate feature selection functions. However, labeling a large number of training data is tedious and time-consuming. In many real world applications, the performance of the existing feature selection algorithms is usually restrained by the paucity of labeled training data. Therefore, it turns out to be a great research challenge to design a feature selection algorithm for the cases when only a few labeled data per task are available.

Although multi-task classification and subspace learning have received much research attention [1], [2], [21], [26], very few efforts have been focused on multi-task feature selection. Intuitively, people often adapt the knowledge obtained from previous experience to facilitate new learning tasks. Meanwhile, it has been empirically and theoretically demonstrated that learning multiple related tasks jointly always gains better performance than learning each task independently [1], [2], [4]. In the field of multimedia, some research efforts have also shown that it is beneficial to leverage the knowledge shared by multiple tasks for multimedia analysis [9] [16] [21], [33]. For example, Ma et al. have proposed a knowledge adaptation algorithm for multimedia event detection, which outperforms SVM dramatically [21]. To address the small number of labeled data problem, it is advantageous to borrow the knowledge from some other related tasks for feature selection. However, most of the existing feature selection algorithms select features for each task independently [12],

- *Y. Yang and A. Hauptmann are with the School of Computer Science, Carnegie Mellon University, USA. E-mail: {yiyang, alex}@cs.cmu.edu.*
- *Z. Ma and N. Sebe are with the Department of Information Engineering and Computer Science, University of Trento, Italy. E-mail: {ma, sebe}@disi.unitn.it*

- *This material is based upon work supported in part by the National Science Foundation under Grants No.IIS-0917072, in part by the National Institutes of Health (NIH) Grant No.1RC1MH090021-01, in part by the European Commission under the contract FP7-248984 GLOCAL and in part by the National Program on Key Basic Research Project of China (973 Program) under grant 2010CB327903.*

[18], [27], [23], [35]. Despite of its importance, multitask feature selection has been largely unaddressed. In this work, we propose a new feature selection algorithm, which leverages the knowledge from related multiple tasks to improve the performance of feature selection. In our study, the following lessons have been learned:

- Sharing information among related tasks is beneficial for supervised learning. However, if the multiple tasks are not correlated, the performance is not necessarily improved.
- Compared to single task learning, the advantages of multitask learning are usually more visible when we only have few training examples per task. As we increase the number of positive training data, the intra-task knowledge is sufficient for training, and thus adapting inter-task knowledge does not necessarily help.
- It is not always the case that feature selection improves the performance. However it is still beneficial because it improves the efficiency. Also, feature selection would provide us with better interpretability of the features.
- The improvement of feature selection varies when different classifiers are used. For example, since linear SVM actually has the ability to assign different weights to different features, the performance improvement of SVM is less than KNN, after feature selection.

The rest of this paper is organized as follows. In section 2, we give the objective function. The optimization approach is proposed in section 3, followed by the proof of its convergence. We then show experimental results and conclude the paper.

2 Feature Selection with Shared Information

In this section, we describe in detail the proposed algorithm. Suppose we are going to select features for t tasks. The l-th task contains m_l training data $\{x_l^i\}_{i=1}^{m_l}$ with groundtruth labels $\{y_l^i\}_{i=1}^{m_l}$ from c_l classes. Denote f_l as the feature selection function for the l-th task. As indicated in [2], [1], it is reasonable to assume that there is certain common information shared by the t tasks. How to reveal the shared information among multiple tasks is the key issue in multitask learning. There are different ways to encode the shared information. For example, Argyriou et al. explore the task relatedness by learning a representation in low dimensional subspace shared across multiple tasks [2]. Ando and Zhang suggest that there should be a shared subspace across multiple tasks for classification [1]. Yang et al. have assumed that there is a shared subspace of different labels and proposed a semi-supervised learning algorithm for multi-label image annotation [36]. A maximum entropy discrimination (MED) algorithm is proposed in [14], which produces multiple support vector machines that learn a shared conic kernel combination.

In this paper, we assume that certain components of the feature selection functions are useful across multiple tasks to uncover the information shared by multiple related tasks. Denote $f = \{f_1, ..., f_t\}$. The common components of different feature selection functions can be encoded by a regularization term $\Omega(f)$. The proposed regularized framework for feature selection can be formulated as follows.

$$\min_{f_l} \sum_{l=1}^{t} \left(\sum_{i=1}^{m_l} loss(f_l(x_l^i), y_l^i) + \alpha g(f_l) \right) + \beta \Omega(f) \qquad (1)$$

where $loss(f_l(x_l^i), y_l^i)$ is a loss function evaluating the consistency between labels and features, $g(f_l)$ is a regularization function, α and β are regularization parameters.

For the ease of representation, we define $X_l = [x_l^1, ..., x_l^{m_l}]$ as the data matrix of the l-th task and $Y_l = [y_l^1, ..., y_1^{m_l}]$ as the corresponding label matrix. Given a matrix $A \in \mathbb{R}^{a \times b}$ where a and b are arbitrary numbers, $\|A\|_F$ is its Frobenius norm. The $\ell_{2,1}$-norm of A is defined as

$$\|A\|_{2,1} = \sum_{i=1}^{a} \sqrt{\sum_{j=1}^{b} A_{ij}^2}. \qquad (2)$$

There are many ways to define the loss function in (1). It has been shown in [3], [10], [15], [36] that the least square loss function gains comparable or better performance to other loss functions such as the hinge loss. In our algorithm, we use the least square loss due to its efficiency and simplicity. Nevertheless, the performance of any algorithm is dependent on the loss function used in the objective function. We omit this discussion because the comparison of different loss functions is out of the scope of this paper. For each task, we propose to select the features that are most correlated to labels and rewrite (1) as follows for feature selection.

$$\min_{W_l} \sum_{l=1}^{t} \left(\left\| W_l^T X_l + b_l 1_l^T - Y_l \right\|_F^2 + \alpha \left\| W_l \right\|_{2,1} \right) + \beta \Omega(W),$$

where $b_l \in \mathbb{R}^{c_l \times 1}$ is the bias term for each task and $1_l \in \mathbb{R}^{m_l \times 1}$ is a vector whose elements are all ones, W_l is the feature selection matrix of the l-th task. As indicated in [23], [35], when minimizing the $\ell_{2,1}$ norm of W_l, some rows of W_l shrink to zero, making W_l particularly suitable for feature selection. Denote $W = [W_1, ..., W_t]$ and the regularization term $\Omega(W)$ is added to explore the shared information among the feature selection functions of multiple tasks.

To step further, we first give the definition of trace norm. For an arbitrary matrix A, its trace norm is defined as

$$\|A\|_* = Tr(AA^T)^{\frac{1}{2}}, \qquad (3)$$

where $Tr(\cdot)$ represents the trace operator. As reported in [26], the shared structure of multiple variables can be expressed by a low rank matrix of them. Inspired by this, we assume that the common information of multiple tasks can be shared by them if we restrict W to be a low rank matrix. Minimizing the rank of a matrix is non-convex. In this paper, we propose to minimize the trace norm of W, which is the convex hull of the rank of W, to utilize the shared information of multiple related tasks. The objective function of the proposed multitask feature selection algorithm is given by

$$\min_{W_l} \sum_{l=1}^{t} \left(\left\| W_l^T X_l + b_l 1_l^T - Y_l \right\|_F^2 + \alpha \left\| W_l \right\|_{2,1} \right) + \beta \left\| W \right\|_* \qquad (4)$$

Compared with minimizing the rank of W directly, the objective function shown above is convex. In (4), with the term $\|W_l\|_{2,1}$, our algorithm is able to evaluate the informativeness of all features jointly for each task, by which the correlation of different features is employed. The trace norm term $\|W\|_*$, on the other hand, enables different feature selection functions $\{W_1, ..., W_l\}$ to share the common components/knowledge across multiple tasks. In this way, the information from different tasks can be transferred from one to another. We name the algorithm **F**eature **S**election with **S**hared **I**nformation among multiple tasks (FSSI).

3 Optimization

In this section, we give an iterative approach to optimize the objective function (4). Denote $W_l = [w_l^1, ..., w_l^d]$, where d is the number of features. First, we write the objective function shown in (4) as follows.

$$\min_{W_l} \sum_{l=1}^{t} \left(\left\| W_l^T X_l + b_l 1_l^T - Y_l \right\|_F^2 + \alpha Tr(W_l^T D_l W_l) \right) + \frac{\beta}{2} Tr(W^T (WW^T)^{-\frac{1}{2}} W), \qquad (5)$$

where D_l is a diagonal matrix which is defined as

$$D_l = \begin{bmatrix} \frac{1}{2\|w_l^1\|_2} & & \\ & \ddots & \\ & & \frac{1}{2\|w_l^d\|_2} \end{bmatrix}. \qquad (6)$$

By setting the derivative of (5) *w.r.t.* b_l to 0, we have

$$b_l = \frac{1}{m_l} Y_l 1_l - \frac{1}{m_l} W_l^T X_l 1_l \qquad (7)$$

Substituting (7) into (5) we obtain

$$\begin{aligned} &\min_{W_l} \sum_{l=1}^{t} \left\| W_l^T X_l + (\frac{1}{m_l} Y_l 1_l - \frac{1}{m_l} W_l^T X_l 1_l) 1_l^T - Y_l \right\|_F^2 \\ &+\alpha \sum_{l=1}^{t} Tr(W_l^T D_l W_l) + \frac{\beta}{2} Tr(W^T (WW^T)^{-\frac{1}{2}} W) \\ \Rightarrow &\min_{W_l} \sum_{l=1}^{t} \left\| W_l^T X_l (I_l - \frac{1}{m_l} 1_l 1_l^T) - Y_l (I_l - \frac{1}{m_l} 1_l) \right\|_F^2 \\ &+\alpha \sum_{l=1}^{t} Tr(W_l^T D_l W_l) + \frac{\beta}{2} Tr(W^T (WW^T)^{-\frac{1}{2}} W), \qquad (8) \end{aligned}$$

where I_l is the identity matrix. Denote $H_l = I_l - \frac{1}{m_l} 1_l 1_l^T$ as the centering matrix. (8) can be rewritten as follows.

$$\min_{W_l} \sum_{l=1}^{t} \left\| W_l^T X_l H_l - Y_l H_l \right\|_F^2 + \alpha \sum_{l=1}^{t} Tr(W_l^T D_l W_l) + \frac{\beta}{2} Tr(W^T (WW^T)^{-\frac{1}{2}} W). \qquad (9)$$

By setting the derivative of (9) *w.r.t.* W_l to 0, we have

$$X_l H_l H_l^T X_l^T W_l + \beta \left(\frac{1}{2} (WW^T)^{-\frac{1}{2}} \right) W_l + \alpha D_l W_l = X_l H_l Y_l^T$$

Therefore, we have

$$W_l = (X_l H_l H_l^T X_l^T + \alpha D_l + \beta \tilde{D})^{-1} X_l H_l Y_l^T. \qquad (10)$$

where $\tilde{D} = \frac{1}{2}(WW^T)^{-\frac{1}{2}}$. Based on the above mathematical deduction, we propose an iterative algorithm to optimize the objective function (4), which is summarized in Algorithm 1. Once W_l $(1 \leq l \leq t)$ is obtained, we sort the d features according to $\|w_l^i\|_F$ $(1 \leq i \leq d)$ in descending order and select the top ranked ones for the l-th task.

Algorithm 1: Feature Selection with Shared Information.

```
Input:
    Input data $X_l \in \mathbb{R}^{d \times m_l}$ $(1 \leq l \leq t)$ and labels
    $Y_l \in \mathbb{R}^{c_l \times m_l}$ of the $t$ tasks;
    Regularization parameters $\alpha$ and $\beta$.
Output:
    Feature selection matrix $W_l|_{l=1}^t \in \mathbb{R}^{d \times c_l}$.
Set $r = 0$ and initialize $W_l|_{l=1}^t \in \mathbb{R}^{d \times c_l}$ randomly;
$W_0 = [W_1, ...W_t]$;
repeat
    $l = 1$;
    repeat
        Compute the diagonal matrix $D_l^r$ as:
        $D_l^r = \begin{bmatrix} \frac{1}{2\|(w_l^1)^r\|_2} & & \\ & ... & \\ & & \frac{1}{2\|(w_l^d)^r\|_2} \end{bmatrix}$;
        Compute the diagonal matrix $\tilde{D}^r$ as:
        $\tilde{D}^r = \frac{1}{2}(W_r W_r^T)^{-\frac{1}{2}}$ ;
        Update $W_l$ by
        $W_l = (X_l H_l H_l^T X_l^T + \alpha D_l^r + \beta \tilde{D}^r)^{-1} X_l H_l Y_l^T$;
        Update $b_l$ by $b_l = \frac{1}{m_l} Y_l 1_l - \frac{1}{m_l} W_l^T X_l 1_l$;
        $l = l + 1$;
    until $l > t$;
    $W_{r+1} = [W_1, ..., W_t]$;
    $r = r + 1$;
until Convergence;
Return $W_l$ and $b_l$ for $1 \leq l \leq t$.
```

4 Convergence Analysis

In this section, we theoretically show that Algorithm 1 proposed in this paper converges. We begin with the following lemma.

Lemma 1. *For any invertible matrices A and A_0, the following inequality holds:*

$$\frac{1}{2}Tr(AA_0^{-\frac{1}{2}}) - Tr(A^{\frac{1}{2}}) \geq \frac{1}{2}Tr(A_0A_0^{-\frac{1}{2}}) - Tr(A_0^{\frac{1}{2}}) \quad (11)$$

Proof: Because $(A^{\frac{1}{2}} - A_0^{\frac{1}{2}})^2$ is semi-positive, we have $Tr\left(A_0^{-\frac{1}{4}}(A^{\frac{1}{2}} - A_0^{\frac{1}{2}})^2A_0^{-\frac{1}{4}}\right) \geq 0$. Note that

$$\begin{aligned} &Tr\left(A_0^{-\frac{1}{4}}(A^{\frac{1}{2}} - A_0^{\frac{1}{2}})^2A_0^{-\frac{1}{4}}\right) \\ &= Tr\left((A^{\frac{1}{2}} - A_0^{\frac{1}{2}})^2A_0^{-\frac{1}{4}}A_0^{-\frac{1}{4}}\right) \end{aligned} \quad (12)$$

Therefore, we have:

$$Tr\left((A + A_0 - 2A^{\frac{1}{2}}A_0^{\frac{1}{2}})A_0^{-\frac{1}{2}}\right) \geq 0 \quad (13)$$

$$\Rightarrow \frac{1}{2}Tr(AA_0^{-\frac{1}{2}}) - Tr(A^{\frac{1}{2}}) \geq \frac{1}{2}Tr(A_0A_0^{-\frac{1}{2}}) - Tr(A_0^{\frac{1}{2}})$$

Thus, we have proved this lemma. □

Next, we show that Algorithm 1 converges by the following theorem.

Theorem 1. *Algorithm 1 monotonically decreases the objective function value of Eq (4) in each iteration.*

Proof: For the ease of representation, we denote the updated W_l, b_l in each iteration as $\hat{W}_l$, $\hat{b}_l$ respectively. The inner loop to update $W_l|_{l=1}^t$ in Step 3 of Algorithm 1 corresponds to the optimal $W_l|_{l=1}^t$ of the following problem

$$\begin{aligned} \min_{W_l}\sum_{l=1}^{t}\left(\left\|W_l^TX_lH_l - Y_lH_l\right\|_F^2 + \alpha Tr\left(W_l^TD_l^rW_l\right)\right) \\ +\beta Tr\left(W^T\tilde{D}_rW\right). \end{aligned} \quad (14)$$

According to the definition of D_l and $\tilde{D}$, we thus have:

$$\begin{aligned} &\sum_{l=1}^{t}\left(\left\|\hat{W}_l^TX_l + \hat{b}_l1_l^T - Y_l\right\|_F^2 + \alpha\sum_{j=1}^{d}\frac{\left\|\hat{w}_l^j\right\|_2^2}{2\left\|w_l^j\right\|_2}\right) \\ &\qquad +Tr\left(\hat{W}^T\frac{\beta}{2}(WW^T)^{-\frac{1}{2}}\hat{W}\right) \\ &\leq \sum_{l=1}^{t}\left(\left\|W_l^TX_l + b_l1_l^T - Y_l\right\|_F^2 + \alpha\sum_{j=1}^{d}\frac{\left\|w_l^j\right\|_2^2}{2\left\|w_l^j\right\|_2}\right) \\ &\qquad +Tr\left(W^T\frac{\beta}{2}(WW^T)^{-\frac{1}{2}}W\right). \end{aligned} \quad (15)$$

The same as in [20], [22], [23], we have the following inequality:

$$\begin{aligned} &\sum_{l=1}^{t}\left(\left\|\hat{W}_l^TX_l + \hat{b}_l1_l^T - Y_l\right\|_F^2 + \alpha\sum_{j=1}^{d}\left\|\hat{w}_l^j\right\|\right) \\ &\qquad +\frac{\beta}{2}Tr\left(\hat{W}\hat{W}^T(WW^T)^{-\frac{1}{2}}\right) \\ &\leq \sum_{l=1}^{t}\left(\left\|W_l^TX_l + b_l1_l^T - Y_l\right\|_F^2 + \alpha\sum_{j=1}^{d}\left\|w_l^j\right\|\right) \\ &\qquad +\frac{\beta}{2}Tr\left(WW^T(WW^T)^{-\frac{1}{2}}\right). \end{aligned} \quad (16)$$

Further, (16) can be rewritten as:

$$\begin{aligned} &\sum_{l=1}^{t}\left(\left\|\hat{W}_l^TX_l + \hat{b}_l1_l^T - Y_l\right\|_F^2 + \alpha\sum_{j=1}^{d}\left\|\hat{w}_l^j\right\|\right) \\ &+\frac{\beta}{2}Tr\left((\hat{W}\hat{W}^T)^{\frac{1}{2}}\right) + \frac{\beta}{2}Tr\left(\hat{W}\hat{W}^T(WW^T)^{-\frac{1}{2}}\right) \\ &\qquad -\frac{\beta}{2}Tr\left((\hat{W}\hat{W}^T)^{\frac{1}{2}}\right) \\ &\leq \sum_{l=1}^{t}\left(\left\|W_l^TX_l + b_l1_l^T - Y_l\right\|_F^2 + \alpha\sum_{j=1}^{d}\left\|w_l^j\right\|\right) \\ &+\frac{\beta}{2}Tr\left((WW^T)^{\frac{1}{2}}\right) + \frac{\beta}{2}Tr\left(WW^T(WW^T)^{-\frac{1}{2}}\right) \\ &\qquad -\frac{\beta}{2}Tr\left((WW^T)^{\frac{1}{2}}\right). \end{aligned} \quad (17)$$

According to Lemma 2, we have

$$\begin{aligned} &\frac{\beta}{2}Tr\left(\hat{W}\hat{W}^T(WW^T)^{-\frac{1}{2}}\right) - \beta Tr\left((\hat{W}\hat{W}^T)^{\frac{1}{2}}\right) \\ &\geq \frac{\beta}{2}Tr\left(WW^T(WW^T)^{-\frac{1}{2}}\right) - \beta Tr\left((WW^T)^{\frac{1}{2}}\right) \end{aligned} \quad (18)$$

Subtracting (18) from (17) , we have

$$\begin{aligned} &\sum_{l=1}^{t}\left(\left\|\hat{W}_l^TX_l + \hat{b}_l1_l^T - Y_l\right\|_F^2 + \alpha\left\|\hat{W}_l\right\|_{2,1}\right) + \beta\left\|\hat{W}\right\|_* \\ &\leq \sum_{l=1}^{t}\left(\left\|W_l^TX_l + b_l1_l^T - Y_l\right\|_F^2 + \alpha\left\|W_l\right\|_{2,1}\right) + \beta\left\|W\right\|_* \end{aligned}$$

Therefore, we have proved the theorem. □

Since with the updating rule in Algorithm 1 the objective function shown in (4) monotonically decreases, it is easy to see that the algorithm converges.

5 EXPERIMENTS

In this section, we conduct experiments to test the performance of our algorithm. We first compare our algorithm with other feature selection methods. Then, we study the performance variance *w.r.t.* different classifiers, parameter sensitivity and the convergence of Algorithm 1.

Table 1
Performance comparison of video action recognition (ACC%) of different feature selection algorithms on KTH database. The best results are highlighted in bold.

Action Type	FSSI	All Features	Max Variance	Fisher Score	mRMR	SPEC	FSNM
Walking	**80.25**	72.84	73.35	72.61	72.68	72.90	73.35
Jogging	**82.32**	78.12	78.16	78.15	78.44	73.79	78.73
Running	**76.30**	74.30	74.41	73.81	73.83	69.98	74.70
Boxing	**71.66**	71.25	71.25	71.22	71.22	70.79	71.28
Hand waving	71.36	71.20	71.20	71.50	71.01	**71.86**	71.21
Hand clapping	**74.05**	72.65	72.65	72.19	72.28	70.39	72.16

5.1 Experiment Setup

We apply our algorithm to three different applications, including video action recognition, face recognition and 3D motion data classification. Six datasets are used in the experiment, including two video datasets KTH [29] and CMU-CareMedia, three image datasets ORL [28], JAFFE [19] and YaleB [11], and one 3D motion skeleton dataset HumanEva. We compare our algorithm with the following methods.

1) **All Features**: All original features are used, which is used as baseline in the experiment;
2) **Max Variance**: It chooses the features with maximum variance;
3) **Fisher Score**: It is a classical method which uses discriminative methods, and generative statistical models to accomplish feature selection [7];
4) **mRMR**: It selects the features that correlate the strongest ones with a classification variable. In addition, it selects features that are mutually different from each other while still having a high correlation to obtain a better feature subset [27];
5) **SPEC**: It conducts feature selection by using the spectral graph theory [38]. We adopt the supervised scenario of this method in our experiments;
6) **Feature Selection via Joint $l_{2,1}$-Norms Minimization**(FSNM): It employs $l_{2,1}$-norm minimization on loss function and regularization for feature selection [23].

For each algorithm, all the parameters (if any) are tuned by a "grid-search" strategy from $\{10^{-6}, 10^{-5}, \cdots, 10^{5}, 10^{6}\}$ and we report the best results. Based on the selected features, we train a classifier for different tasks. Unless otherwise specified, we utilize linear SVM as the classifier. Accuracy (ACC) is used to evaluate the performance. As discussed previously, we aim to improve the feature selection performance when the training data are few. In all the experiments, we randomly sample five data per class as the training sets for all the applications. The remaining samples are used as the corresponding testing sets. The experiments are independently repeated 5 times and the average results are reported.

5.2 Video Action Recognition

First, we compare different algorithms in terms of video action recognition. In this experiment, the KTH database [29] and CMU-CareMedia dataset are used. We first use KTH data to test the performance of our algorithm. We take each action type as a separate recognition task, thus resulting in six tasks. Shao and Mattivi [30] compared different features for action classification and showed that the Harris3D + HOG/HOF [32] is a good option. Therefore, we use the Harris3D + HOG/HOF [32] to process the database and a 1000 dimension Bag-of-Words feature is generated to represent each video sequence. We set the numbers of selected features as $\{100, 200, \cdots, 800, 900\}$ for all the algorithms and report the best results.

The experiment results are shown in Table 1. From the table, we observe that our algorithm gains the best performance for 5 tasks out of 6. For the actions *walking*, *jogging*, *running*, and *hand clapping*, our algorithm gains dramatically better performance compared with other algorithms. For *hand waving*, SPEC slightly outperforms our algorithm. Yet, our algorithm is much better than SPEC for all of the other action types. This experiment demonstrates that it is beneficial to utilize shared information among different tasks for feature selection, although for some tasks (e.g., *hand waving*), the shared information might not be useful.

Next, we use CMU-CareMedia dataset collected by Carnegie Mellon University to compare the performance of different feature selection algorithms in action recognition. We adopt the same experiment setting as for the KTH dataset. CMU-CareMedia dataset contains surveillance videos recorded in a nursing home. The data are used for studying patients daily activities, thus providing useful statistics to help doctors' diagnosis. Compared to KTH, CMU-CareMedia is more of a real-world dataset, and it is more complex. The following 5 activity categories are considered: *walking through, standing up, sitting down, object placed on table*, and *object removed from table*, resulting in 5 tasks. In this experiment, we test the action recognition of the video data recorded by a particular camera in the dining room. This camera captures patients' activities during lunch and dinner time, which contains the largest number of labeled data.

Table 2 shows the experiment results of action recognition using CMU-CareMedia dataset. We can see that our algorithm outperforms all of the competitors. In this experiment, our algorithm significantly outperforms others for four out of five categories. This experiment further demonstrates that our

Table 2
Performance comparison of video action recognition (ACC%) of different feature selection algorithms on CareMedia database. The best results are highlighted in bold.

Action Type	FSSI	All Features	Max Variance	Fisher Score	mRMR	SPEC	FSNM
Walking through	**68.18**	65.77	66.01	66.13	66.28	65.55	66.16
Standing up	**78.82**	68.77	69.17	68.35	68.35	68.73	72.22
Sitting down	66.95	63.24	64.38	64.22	64.22	64.07	**67.05**
Object placed on table	**78.63**	75.90	75.80	77.02	77.02	75.66	76.92
Object removed from table	**74.85**	65.20	64.67	67.96	67.59	65.75	67.96

Table 3
Performance comparison of face recognition (ACC%) of different feature selection algorithms on ORL, JAFFE and YaleB databases. The best results are highlighted in bold.

Image Database Name	FSSI	All Features	Max Variance	Fisher Score	mRMR	SPEC	FSNM
ORL	**84.11**	82.70	82.70	82.90	82.90	82.96	82.87
JAFFE	**97.31**	96.32	96.44	96.07	96.07	96.68	96.68
YaleB	**56.10**	55.37	54.60	54.88	54.88	55.29	55.35

algorithm is capable of uncovering the information shared by multiple tasks for feature selection. We also observe that exploiting the common information shared by multiple tasks is not always beneficial. Accordingly, the accuracy of *sitting down* of our algorithm is a bit worse than FSNM.

5.3 Face Recognition

We use three face image databases, namely ORL [28], JAFFE [19] and YaleB [11] as three multi-class classification tasks to test the performance of our algorithm. The images from the three databases are cropped and resized to 32 × 32 pixels. We directly use the gray pixel as the 1024 features because it is simple and has been widely used in previous papers. The number of selected features are tuned from $\{100, 200, \cdots, 800, 900\}$ for all the algorithms and we report the best results.

The experimental results are shown in Table 3. We observe from the table that our algorithm gains the best performance for all of the three databases. Although the three image databases are all face images, they are actually quite different from each other. For example, JAFFE database has a huge variance in expressions, and was originally collected for expression recognition. Differently, the images from YaleB database may be very different from each other regarding lighting directions. On the other hand, the ORL images can be different in terms of facial details, *e.g.*, with or without glasses. The correlations among different tasks are not as tight as video action because the data are from different datasets recorded for different purposes. For example, while JAFFE was recorded mainly for expression recognition, YaleB more focuses on lighting differences. As a result, we observe that the performance gain of our algorithm is not as obvious as in the case of video actions. This observation indicates that 1) our algorithm is able to utilize the shared information among tasks even if they are not tightly related; 2) the more multiple tasks are correlated, the more performance gain can be obtained. How to estimate if multiple tasks are tightly correlated in an automatic way is still an open problem in multi-task learning.

5.4 3D Motion Annotation

Next, we test the performance of our algorithm in terms of 3D motion data annotation, using the 3D motion data from HumanEva database[1]. The HumanEva database contains five types of motions, namely *boxing*, *gesturing*, *jogging*, *throw-catch* and *walking*. As indicated in [5], [24], the data from all the activities are redundant. In this experiment, we use the same dataset as in [34], which contains 10,000 randomly sampled data of two subjects (5,000 per subject). A 3D pose is encoded as a collection of joint coordinates in 3D space and there are 16 joints in the HumanEva data set. Note that different subjects may have different height, different leg/arm length and so forth. Therefore, in this experiment, the two subjects are considered as two different tasks.

Based on the 16 joint coordinates in 3D space, 1590 geometric pose descriptors are extracted using the method proposed in [5] to represent 3D motion data. In this experiment, the number of selected features are set as $\{300, 500, \cdots, 1300, 1500\}$. Table 4 shows the experiment results of 3D motion data classification. We similarly observe that our algorithm gains the best performance for the two subjects. This experiment provides further evidence that our algorithm is advantageous in a variety of applications.

5.5 Performance using Different Classifiers

The performance improvement of feature selection varies when different classifiers are used. Taking KTH dataset as an example, we compare the performance of different feature

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

Table 4
Performance comparison of 3D motion classification (ACC%) of different feature selection algorithms on HumanEva database. The best results are highlighted in bold.

ID	FSSI	All Features	Max Variance	Fisher Score	mRMR	SPEC	FSNM
Subject 1	**82.16**	78.46	78.46	78.47	78.47	79.82	79.56
Subject 2	**74.21**	73.12	73.12	73.55	73.55	72.99	72.81

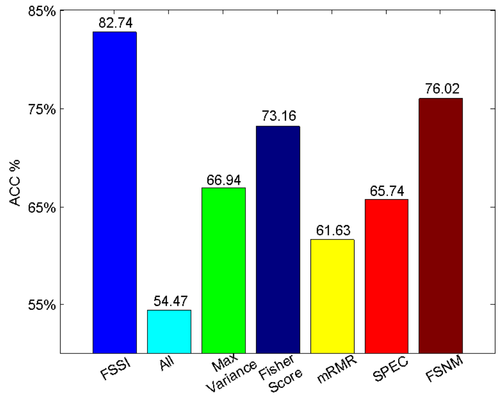


Figure 1. Average recognition Accuracy (ACC) on KTH database using KNN classifier.

selection algorithms when different classifiers are used. In particular, we use KNN as an alternative classifier. Figure 1 shows the average recognition accuracy of the six action types. Comparing Table 1 and Figure 1, we can see that when using all features for action recognition, the accuracy of KNN is lower than that of SVM. However, after feature selection, KNN gains higher accuracy than SVM. One possible explanation is that SVM has the ability to weigh different features, and thus the benefit from feature selection is less.

5.6 Parameter Sensitivity

Using KTH database, we take video action recognition as an example to show the performance variance of our algorithm *w.r.t.* different parameters. We report the results of the first two tasks, *i.e.*, *walking* and *jogging*. First, we fix β and report the performance when α and the number of selected features are changing. The experimental results are shown in Figure 2. We observe that the performance of our algorithm varies when the parameters are different. Generally speaking, the algorithm gains better performance when the number of selected features is 700 to 800 for this database. We also fix α and feature number to test the performance variance when β is changing. Note that the parameter β actually controls the shared components of different feature selection functions $\{W_1, ..., W_t\}$. In the extreme case when $\beta = 0$, the multiple tasks are learned separately, *i.e.*, no common knowledge of multiple tasks is utilized for feature selection. Therefore, Figure 3 clearly demonstrates that the performance can be improved by leveraging the shared knowledge from multiple related tasks for a specific task. However, the optimal parameters are data dependent. It still remains an open problem how to obtain the optimal parameter automatically.

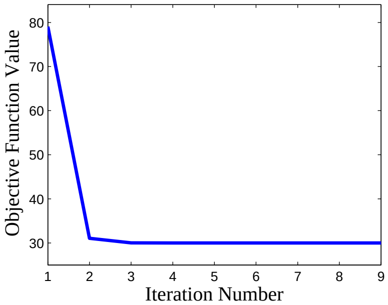


Figure 4. The objective function value of KTH database at each iteration when $\alpha = 1$ and $\beta = 1$.

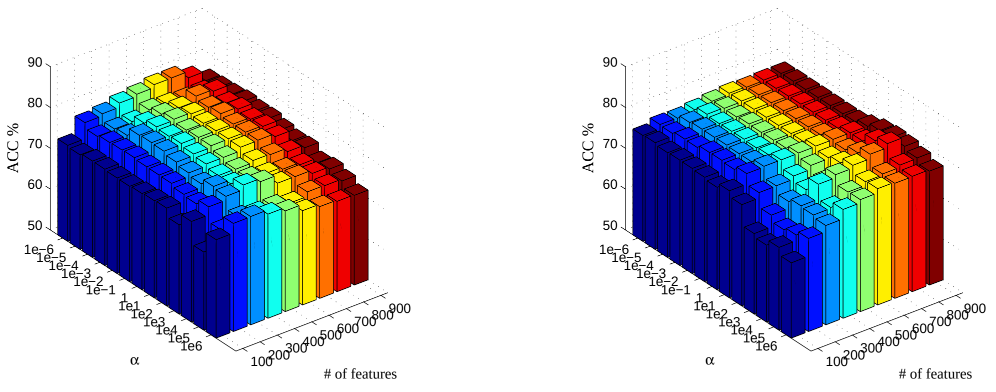
5.7 Convergence Study

In Figure 4, we plot the objective function value of (4) at each iteration of KTH video database. In this figure, the objective function value of (4) monotonically decreases at each iteration, which is consistent with Theorem 1. More specifically, the algorithm converges within 10 iterations for this database, which is more efficient than some of the existing related algorithms for trace norm or $\ell_{2,1}$-norm optimization, such as [8] and [25].

6 Conclusion and Future Work

In this paper, we have proposed a new feature selection algorithm which is able to exploit the information shared by multiple related tasks for multimedia content analysis. In our algorithm, the shared information is transferred between tasks by assuming that feature selection functions of different related tasks have certain common components. Our algorithm evaluates the importance of different features jointly, by which the feature correlation is considered as well. The objective function of our algorithm is convex, and an effective iterative algorithm was proposed to optimize the objective function. We presented a variety of experimental results showing that the performance of our algorithm is superior to the existing feature selection algorithms using different datasets and classifiers.

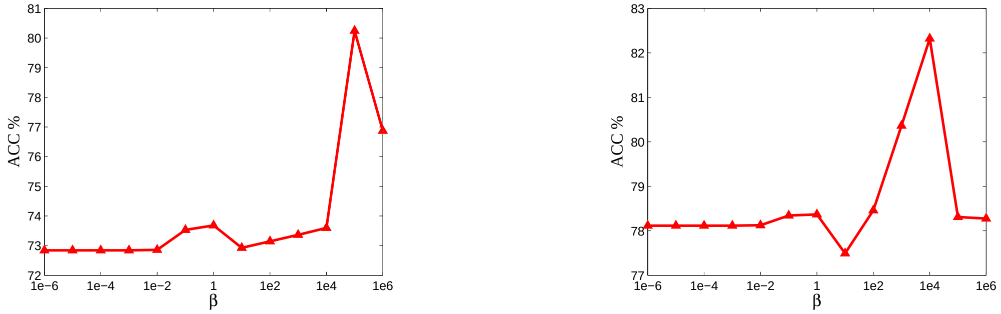
Our work is based on the assumption that the multiple tasks are correlated. Automatic evaluation of the correlation among multiple tasks is an open problem. Practically, it is the human supervisors' job to estimate if the multiple tasks are correlated.



(a) Walking

(b) Jogging

Figure 2. Recognition Accuracy (ACC) with different α and feature numbers while keeping β fixed on KTH database.



(a) Walking

(b) Jogging

Figure 3. Recognition Accuracy (ACC) with different β while keeping α and feature number fixed on KTH database.

However, human estimations are sometimes inaccurate. Even if humans think that the two tasks are correlated, there might not be shared information for training. As the performance of any multitask learning algorithm may drop if we share the information among irrelevant tasks, one interesting direction in the future study is to automatically evaluate if the multiple tasks are correlated to each other.

References

[1] R. K. Ando and T. Zhang. A framework for learning predictive structures from multiple tasks and unlabeled data. *Journal of Machine Learning Research*, 6:1817–1853, 2005.

[2] A. Argyriou, T. Evgeniou, and M. Pontil. Convex multi-task feature learning. *Machine Learning*, 73:243–272, 2008.

[3] L. Bao and et al. Informedia @ Trecvid. *Trecvid Video Retrieval Evaluation Workshop, NIST*, 2011.

[4] R. Caruana. Multitask learning. *Machine Learning*, 28(1):41–75, 1997.

[5] C. Chen, Y. Zhuang, F. Nie, Y. Yang, F. Wu, and J. Xiao. Learning a 3D human pose distance metric from geometric pose descriptor. *IEEE Transactions on Visualization and Computer Graphics*, 17(11):1676–1689, 2011.

[6] X. Chen, A. Hero, and S. Savarese. Multimodal video indexing and retrieval using directed information. *IEEE Transactions on Multimedia*, 14(1):3–16, 2012.

[7] R. Duda, P. Hart, and D. Stork. *Pattern Classification*. John Wiley & Sons, New York, 2nd edition, 2001.

[8] M. Dudík, Z. Harchaoui, and J. Malick. Lifted coordinate descent for learning with trace-norm regularization. *Journal of Machine Learning Research - Proceedings Track*, 22:327–336, 2012.

[9] J. Fan, H. Luo, Y. Gao, and R. Jain. Incorporating concept ontology for hierarchical video classification, annotation, and visualization. *IEEE Transactions on Multimedia*, 9(5):939–957, 2007.

[10] G. Fung and O. L. Mangasarian. Multicategory proximal support vector machine classifiers. *Machine Learning*, 59(1-2):77–97, 2005.

[11] A. Georghiades, P. Belhumeur, and D. Kriegman. From few to many: Illumination cone models for face recognition under variable lighting and pose. *IEEE Transactions on Pattern Analysis and Machine Intelligencee*, 23(6):643–660, 2001.

[12] M. A. Hall and L. A. Smith. Feature selection for machine learning: Comparing a correlation-based filter approach to the wrapper. In *AAAI*, 1999.

[13] A. G. Hauptmann, R. Yan, W.-H. Lin, M. G. Christel, and H. D. Wactlar. Can high-level concepts fill the semantic gap in video retrieval? a case study with broadcast news. *IEEE Transactions on Multimedia*, 9(5):958–966, 2007.

[14] T. Jebara. Multitask sparsity via maximum entropy discrimination. *Journal of Machine Learning Research*, 12:75–110, Feb. 2011.

[15] S. Ji, L. Tang, S. Yu, and J. Ye. A shared-subspace learning framework for multi-label classification. *ACM Transactions on Knowledge Discovery from Data*, 4(2):1–29, 2010.

[16] H. Li, Y. Shi, M. Chen, A. G. Hauptmann, and Z. Xiong. Hybrid active learning for cross-domain video concept detection. In *ACM Multimedia*, pages 1003–1006, 2010.

[17] Z. Li, Y. Yang, J. Liu, X. Zhou, and H. Lu. Unsupervised feature selection using nonnegative spectral analysis. In *AAAI*, 2012.

[18] H. Liu and L. Yu. Feature selection for high-dimensional data: A fast correlation-based filter solution. In *ICML*, 2003.

[19] M. J. Lyons, J. Budynek, and S. Akamatsu. Automatic classification of single facial images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(12):1357–1362, 1999.

[20] Z. Ma, F. Nie, Y. Yang, J. R. R. Uijlings, and N. Sebe. Web image annotation via subspace-sparsity collaborated feature selection. *IEEE Transactions on Multimedia*, 14(4):1021-1030, 2012.

[21] Z. Ma, Y. Yang, Y. Cai, N. Sebe, and A. Hauptmann. Knowledge adaptation for Ad Hoc multimedia event detection with few examplars. In *ACM Multimedia*, 2012.

[22] Z. Ma, Y. Yang, F. Nie, J. R. R. Uijlings, and N. Sebe. Exploiting the entire feature space with sparsity for automatic image annotation. In *ACM Multimedia*, 2011.

[23] F. Nie, H. Huang, X. Cai, and C. Ding. Efficient and robust feature selection via joint $\ell_{2,1}$-norms minimization. In *NIPS*, 2010.

[24] H. Ning, W. Xu, Y. Gong, and T. Huang. Discriminative learning of visual words for 3d human pose estimation. In *CVPR*, 2008.

[25] G. Obozinski and B. Taskar. Multi-task feature selection. In *The workshop of Structural Knowledge Transfer for Machine Learning in ICML*, 2006.

[26] G. Obozinski, B. Taskar, and M. I. Jordan. Joint covariate selection and joint subspace selection for multiple classification problems. *Statistics and Computing*, 20(2):231–252, 2010.

[27] H. Peng, F. Long, and C. Ding. Feature selection based on mutual information: Criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1226–1238, 2005.

[28] F. Samaria and A. Harter. Parameterisation of a stochastic model for human face identification. In *IEEE Workshop on Applications of Computer Vision*, 1994.

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

[30] L. Shao and R. Mattivi. Feature detector and descriptor evaluation in human action recognition. In *ACM CIVR*, 2010.

[31] L. Talavera. An evaluation of filter and wrapper methods for feature selection in categorical clustering. In *IDA*, 2005.

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

[33] J. Yang, R. Yan, and A. G. Hauptmann. Cross-domain video concept detection using adaptive SVMs. In *ACM Multimedia*, pages 188–197, 2007.

[34] Y. Yang, F. Nie, D. Xu, J. Luo, Y. Zhuang, and Y. Pan. A multimedia retrieval framework based on semi-supervised ranking and relevance feedback. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(4):723–742, 2012.

[35] Y. Yang, H. T. Shen, Z. Ma, Z. Huang, and X. Zhou. $\ell_{2,1}$-norm regularized discriminative feature selection for unsupervised learning. In *IJCAI*, 2011.

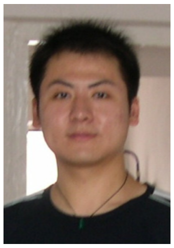
[36] Y. Yang, F. Wu, F. Nie, H. T. Shen, Y. Zhuang, and A. G. Hauptmann. Web and personal image annotation by mining label correlation with relaxed visual graph embedding. *IEEE Transactions on Image Processing*, 21(3):1339–1351, 2012.

[37] Y. Yang, Y. Zhuang, F. Wu, and Y. Pan. Harmonizing hierarchical manifolds for multimedia document semantics understanding and cross-media retrieval. *IEEE Transactions on Multimedia*, 10(3):437–446, 2008.

[38] Z. Zhao and H. Liu. Sprectral feature selection for supervised and unsupervised learning. In *ICML*, 2007.

Yi Yang received the Ph.D degree in Computer Science from Zhejiang University, in 2010. After his graduation, Yi worked in the Data and Knowledge Engineering (DKE) research group at the University of Queensland as a postdoctoral fellow. In May 2011, he joined the Informedia group at the School of Computer Science, Carnegie Mellon University, as a postdoctoral research fellow.

Dr. Yang's research interests include machine learning and its applications to multimedia content analysis and computer vision, e.g. multimedia indexing and retrieval, image annotation, multimedia event detection, etc.



Zhigang Ma received the B.S. and M.S. both from Zhejiang University, Hangzhou, China in 2004 and 2006 respectively. He was a visiting student at the School of Computer Science, Carnegie Mellon University from September 2011 to March 2012. He is currently working toward the PhD degree from the University of Trento, Trento, Italy.

His research interests include machine learning and its application to computer vision and multimedia analysis.

Alexander G. Hauptmann received the B.A. and M.A. degrees in psychology from Johns Hopkins University, the degree in computer science from the Technische Universität Berlin, in 1984, and the Ph.D. degree in computer science from Carnegie Mellon University (CMU), in 1991.

He is currently with the faculty of the Department of Computer Science and the Language Technologies Institute, CMU. His research interests include several different areas: man-machine communication, natural language processing, speech understanding and synthesis, video analysis, and machine learning. From 1984 to 1994, he worked on speech and machine translation, when he joined the Informedia project for digital video analysis and retrieval, and led the development and evaluation of news-on-demand applications.

Nicu Sebe (M'01-SM'11) received the Ph.D. in computer science from Leiden University, Leiden, The Netherlands, in 2001.

Currently, he is with the Department of Information Engineering and Computer Science, University of Trento, Italy, where he is leading the research in the areas of multimedia information retrieval and human-computer interaction in computer vision applications. He was involved in the organization of the major conferences and workshops addressing the computer vision and human-centered aspects of multimedia information retrieval, among which as a General Co-Chair of the IEEE Automatic Face and Gesture Recognition Conference, FG 2008, ACM International Conference on Image and Video Retrieval (CIVR) 2007 and 2010, and WIAMIS 2009 and as one of the initiators and a Program Co-Chair of the Human-Centered Multimedia track of the ACM Multimedia 2007 conference. He is the general chair of ACM Multimedia 2013 and was a program chair of ACM Multimedia 2011. He is a senior member of IEEE and of ACM.