Heterogeneous Feature Selection With Multi-Modal Deep Neural Networks and Sparse Group LASSO

Lei Zhao, Qinghua Hu, *Senior Member, IEEE*, and Wenwu Wang, *Senior Member, IEEE*

***Abstract*—Heterogeneous feature representations are widely used in machine learning and pattern recognition, especially for multimedia analysis. The multi-modal, often also high-dimensional, features may contain redundant and irrelevant information that can deteriorate the performance of modeling in classification. It is a challenging problem to select the informative features for a given task from the redundant and heterogeneous feature groups. In this paper, we propose a novel framework to address this problem. This framework is composed of two modules, namely, multi-modal deep neural networks and feature selection with sparse group LASSO. Given diverse groups of discriminative features, the proposed technique first converts the multi-modal data into a unified representation with different branches of the multi-modal deep neural networks. Then, through solving a sparse group LASSO problem, the feature selection component is used to derive a weight vector to indicate the importance of the feature groups. Finally, the feature groups with large weights are considered more relevant and hence are selected. We evaluate our framework on three image classification datasets. Experimental results show that the proposed approach is effective in selecting the relevant feature groups and achieves competitive classification performance as compared with several recent baseline methods.**

***Index Terms*—Deep learning, feature selection, heterogeneous data, multi-modal, sparse representation.**

I. Introduction

WITH the rapid progress in data acquisition and feature extraction, multi-modal information has been widely used in machine learning, pattern recognition and data mining. For example, in data mining of social media from Twitter and Flikcr, as shown in Fig. 1, the data may contain texts, images, audio, and videos; in medical analysis, various multi-modal information is collected, such as X-ray, CT, MRI, PET, SPECT, and fMRI. To represent these multi-modal data, a great number of feature extraction and description methods have been used, such as FFT, wavelet, HOG, SIFT, and LBP. These facts lead to a challenging task: learning with multi-modal information.

Manuscript received February 07, 2015; revised May 31, 2015; accepted August 23, 2015. Date of publication September 07, 2015; date of current version October 20, 2015. This work was supported in part by the 973 Program under Grant 2013CB329304 and by the National Natural Foundation of China under Grant 61222210 and Grant 61432011. The guest editor coordinating the review of this manuscript and approving it for publication was Dr. Guo-Jun Qi. *(Corresponding author: Qinghua Hu.)*

L. Zhao and Q. Hu are with the School of Computer Science and Technology, Tianjin University, Tianjin 300072, China (e-mail: huqinghua@tju.edu.cn).

W. Wang is with the Centre for Vision, Speech and Signal Processing, University of Surrey, Guildford GU2 7XH, U.K. (e-mail: w.wang@surrey.ac.uk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2015.2477058

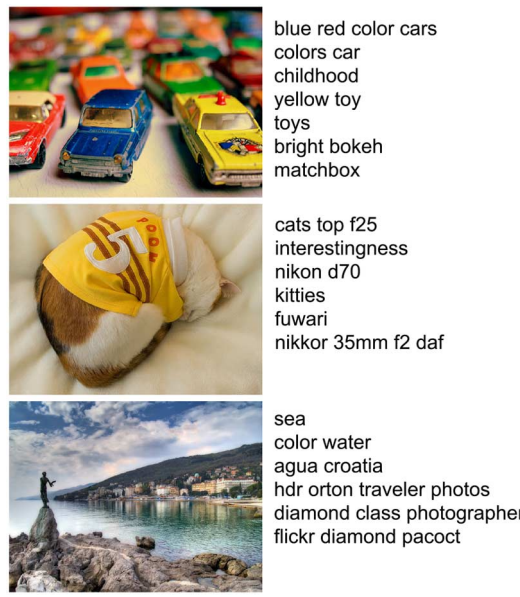


Fig. 1. Example images from Flickr with their associated text description tags. The tags include some relevant words to the image itself and some other information such as the camera settings and author's information.

Learning from multi-modal data introduces some new difficulties. First, as we know, most existing learning algorithms require the data to be represented by feature vectors. It has been shown however that with vectorial representation, some key information hidden in the raw data may be lost. Nevertheless, with other forms of representations, such as bag-of-features [1], high-order tensors [2] or matrices, the original data could be characterized more precisely. These feature descriptors, reflecting different aspects of the original task, may have distinct distributions in a variety of feature spaces. It is therefore important to effectively integrate these heterogenous features. Second, multi-modal data are usually high-dimensional. In high-dimensional feature representations, some features may be redundant or irrelevant to the task under consideration. The irrelevant features, or features corrupted by noise, could even deteriorate the performance of modeling. In some applications, such as medical analysis and bioinformatics, it may become expensive to acquire and extract the features. Hence, it is highly desirable to design an effective approach to evaluating the multi-modal features and selecting the relevant and necessary features.

To make full use of the multi-modal information, several new learning frameworks have been developed in recent years [3]. For example, multiple kernel learning (MKL) based algorithms have been proposed [4]–[6] to address the problem of unifying the representations of heterogenous features. Especially,

Guillaumin *et al.*[7] proposed an MKL based semi-supervised learning method to fuse both modalities of images and tags. Qi *et al.* proposed a unified structured representation called Multimedia Information Networks (MINets), which incorporates multiple information cues in social media and maps different modalities into a latent space [8], [9]. In paper [10], a robust link transfer model is proposed for efficient link knowledge transfer between the networks. This makes it possible for leveraging multi-modal information simultaneously. Yang *et al.*[11] presented a multi-feature model via hierarchical regression to exploit the information derived from various features. In addition, sparse representation based methods have also been proposed to exploit the redundancy in the high-dimensional data. Wang *et al.*[12] presented a sparse multi-modal learning method to integrate heterogeneous image features by solving an optimization problem with joint structured sparsity regularizations. Shekhar *et al.*[13] proposed a method which utilizes observations from multiple modalities to construct the sparse representations. Moreover, deep learning based multi-modal fusion methods have also been proposed recently [14]–[16]. For example, in [17], the distance metric between different modalities is learned by deep neural networks. The methods mentioned above are focused on the problem of how to utilize multiple features more effectively. In these reasearches, however, no attention has been paid to the problem of evaluating the importance of each type of features for the tasks investigated. The objective of this work is on this problem by evaluating the importance of each type of features, selecting the relevant features, and filtering out those irrelevant types of features that may have negative impact on the entire model.

To address the problem of feature selection from heterogeneous features, structured sparsity based techniques have been proposed recently in [18]–[20] where the irrelevant features are filtered out from the multiple heterogeneous feature descriptors. In [21], [22], the problem is addressed by combining the extended $\ell_{2,1}$-norm and unsupervised learning. The feature selection algorithm presented in [23] exploits the information shared by multiple related tasks for multimedia content analysis. Hu *et al.*[24] proposed a method based on neighborhood rough set for heterogeneous feature subset selection. In all these approaches, the original different features are represented by a feature vector and then put into the same feature space, where it is assumed that some association could be found. Nevertheless, when these approaches are applied to the problem of heterogeneous feature selection, they simply neglect the distinctions among the intrinsic structures of various feature representations extracted from different modalities. Intuitively, it is an unreasonable hypothesis. Therefore, a better framework needs to be developed for heterogeneous feature selection. This is the second focus of our work here.

Concentrating on the two main issues mentioned above: 1) how to integrate the discriminative feature representations obtained in different ways into a unified form of feature representation, and 2) how to evaluate each feature group and select the relevant features for the task under consideration. In this paper we propose a novel feature selection framework by combining multi-modal deep neural networks with sparse group lasso. With the multi-modal deep neural networks, the structure of the heterogeneous features which may be hidden in a complicated high dimensional and nonlinear space, can be projected into a new linear space. Then the feature selection is achieved through solving an optimization problem with an L1 regularization together with an additional regularization which encourages sparsity on feature groups. An importance weight for each feature group will be obtained and based on which the irrelevant feature groups are filtered out. We applied our method to three real world datasets with several irrelevant noisy feature groups mixed for image classification tasks. Experimental results show that this framework can discover the relevant feature groups effectively and achieves better classification accuracies compared with several baseline approaches for heterogenous feature selection.

The remainder of this paper is organized as follows. Section II reviews some important and related work on multiple feature integration and heterogeneous feature selection including the MKL method, structured sparse representation, and deep neural networks. Section III presents the proposed framework for grouped feature selection with multi-modal neural networks and sparse group lasso. Experimental results and analysis are given in Section IV. Section V draws the conclusions and gives a discussion on future work.

II. Related Work

Before introducing our heterogeneous feature selection framework, we review some works related to multiple feature integration and feature selection on account of some crucial concepts and key ideas based on which our framework is established.

A. Multiple Feature Integration With MKL

For real world data such as images, the intrinsic structure of most of the feature descriptors extracted is often embeded in a high dimensional and nonlinear space. To reduce the dimensionality of the features, several kernelization based methods have been proposed [25], [26]. Combined with the support vector machine (SVM), these approaches perform well in processing high dimensional features. However, these approaches concentrate on learning single kernel and neglect the distinctions between the different feature groups in the new feature space.

Different from the single kernel methods discussed above, the MKL method learns a combination of multiple kernel functions [27], [28]. Let $\{x_i, y_i\}_{i=1}^N$ be the learning set, where y_i is the target value for sample x_i. Define $\{K_m\}_{m=1}^M$ as a set of base kernel functions. The common MKL problem for binary classification can be formulated as

$$f(x) = \sum_{i=1}^{N} \alpha_i y_i K(x, x_i) + b \tag{1}$$

$$K(x, x') = \sum_{m=1}^{M} \beta_m K_m(x, x'), \beta_m \geq 0 \tag{2}$$

where β_m is the weight of the kernel function K_m, $\{\alpha_i\}_{i=1}^N$ and b are coefficients to be learned from the given training data. For multiple feature groups obtained from the same pattern, each one could be taken as the input of a base kernel function K_m.

How to get an optimal β, in other words, how to obtain an optimal combination of all the kernel functions is an important problem.

A simple combination is to assign each kernel function the same weight. However, this method ignores the different effects of the distrinct features on the entire model. In [4], the MKL problem is addressed with an additional constraint on the weights of the base kernels. This constraint encourages sparsity on the combination of the kernels. In [29], a novel MKL dimensionality reduction framework is presented, where the optimal base kernel ensemble coefficients $\{\beta\}_{i=1}^{M}$ are determinined with graph embedding. These approaches learn an optimal weight for each kernel, however, for heterogeneous feature selection, the base kernels used in these methods need to be predefined manually. In practice, choosing a proper kernel for each feature group is an intractable and challenging problem.

B. Heterogeneous Feature Selection With Structural Sparsity

Many high-dimensional features of real world data could be represented by a subset derived from the set of elemental descriptors. This has led to the development of sparse representation based algorithms for feature selection. Tibshirani [30] presented the popular *lasso* algorithm in 1996. It adds an additional L1-norm penalty on the widely used least squares loss that encourages the sparsity of feature coefficients. Based on lasso, many feature selection methods were proposed in computer vision and multimedia retrieval [18], [21], [22], [31].

The approaches mentioned above, however, concentrate on the sparsity of the single basic element in the feature vector. For the problem of heterogeneous feature selection, they ignore the group property of the concatenated group features. Some studies extended the L1-norm in *lasso* to ℓ_1/ℓ_q-norm which facilitates group sparsity when $q > 1$, [32], [33]. Yuan and Lin [34] proposed group lasso by considering the group structure existing in the entire feature vector. The model yields an optimal solution to the feature selection problem where some feature groups may be dropped according to the sparsity coefficient λ. In [35], Wu *et al.* extended the group lasso with the logistic regression for heterogeneous high dimensional feature selection.

The group lasso has also been further extended to sparse group lasso. For example, Friedman *et al.*[36] presented a group lasso model with an L2-norm regularization which yields sparisty in intra-group and inter-group simultaneously. With sparse group lasso, not only some feature groups will be dropped but also some features within the remaining groups will be removed. Similarly, Wu *et al.*[19] presented a multi-label boosting framework with structural group sparsity, which yields the selection of heterogeneous features. Peng *et al.*[37] employed a similar idea on identifying the primary predictors in integrative genomics study. For all these structural sparsity based methods, the original feature groups are concatenated into a new long feature vector. This may be inappropriate since the different feature groups are derived from distinctive channels of the original data that have different distributions.

C. Feature Transformation With Deep Neural Networks

Recently, deep learning has become a hot spot in machine learning research for its success in many fields such as image or speech recognition and information retrieval. Given different data, instead of designing a handcraft feature representation, a deep learning algorithm tends to learn a good abstract representation for the current task with a series of nonlinear transformations. A typical deep architecture consists of several hidden layers, and a hierarchical representation can be learned from the original inputs with these hidden layers.

Hinton and Salakhutdinov [38] developed effective algorithms for deep learning in 2006. In the following years, deep learning has attracted much attention thanks to its strong ability in feature learning. There are also some works on multi-modal information integration using deep models. Ngiam *et al.*[14] present a bi-modal deep auto-encoder which learns a joint feature representation from audio and video simultaneously. They apply their model to cross modality learning and bi-modal fusion. Srivastava and Salakhutdinov [15] propose a model of multi-modal Restricted Boltzmann Machines (RBM). Similar to the work of [14], with this model, a joint representation could be obtained from the two given modalities: image and text. First, the two branches of their networks are pre-trained separately in a completely unsupervised fashion. An additional layer is added on the top of the two pre-trained branches and then a RBM is constructed to fine-tune all the layers with back-propagation. In this way, a joint distribution over images and text is learned. In [39], a multi-source deep network is constructed to integrate multiple information and applied to human pose estimation. In this work, multiple less abstract conventional representations for human pose estimation are refined with deep networks for extracting more abstract representation on the concept level. A fusion representation is learned simultaneously and used for the final prediction. Wu *et al.*[17] use mutlimodal deep neural networks to learn a combined non-linear similarity function. They trained multiple deep denoising autoencoders for different low-level features in an unsupervised manner. In the fine-tuning stage, an optimal combination of modality independent non-linear similarity functions is learned. Zhou *et al.*[16] combine multi-modal deep neural networks with conditional random fields (CRF) and applied it to dialogue act recognition by using multiple features simultaneously. Similarly, by treating different low-level features as different modalities, the deep networks are used for learning better latent representations. Then a CRF model is used for discovering the correlations across labels.

However, all these approaches exploit additional hidden layers or other shallow models for integrating multiple latent features learned by the base multi-modal networks. They concentrate on how to take advantage of multiple features effectively and care little for the various impact of different modalities on the performance of the final recognition tasks.The key difference between our approach and the approaches mentioned above is that we train a unique sub-network for every feature group (modality) while all these sub-networks share the same optimization objective in the back-propagation period of the fine-tuning stage. Through the multiple nonlinear transformation with these sub-networks, we aim to obtain a unified high-level abstract representation on the concept level for each type of original feature representations. These sub-networks are combined to construct a multi-modal neural networks.

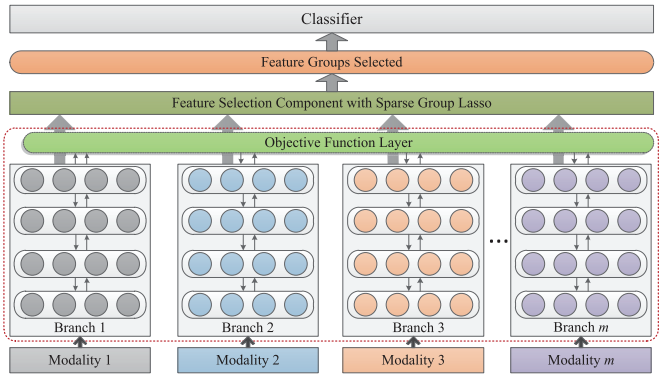


Fig. 2. Architecture of the proposed feature selection framework, composed of multi-modal neural networks and sparse group LASSO. The multi-modal neural networks is shown in the red dashed box.

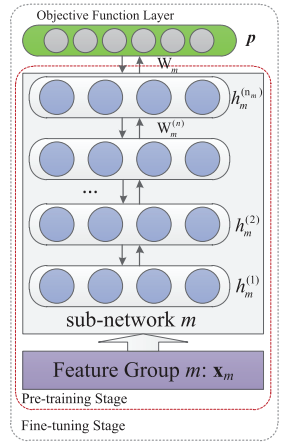


Fig. 3. Illustration of the structure of the sub-networks. In the pre-training stage, we train the branch sub-networks (including those layers in the red box) as stacked denoising auto-encoders layer-wisely. In the fine-tuning stage, we train the branches and the shared objective function layer overall as multilayer perceptron with back-propagation.

Different from some fusion networks [14], [15], we do not set any fusion layers on the top of entire networks.

In addition, the major barrier to handling multi-modal information for conventional feature selection with regularization is the heterogeneity existing among different modalities. However we eliminate this negative impact by mapping the heterogeneous modalities into a latent concept space with the elaborate multi-modal deep networks. This is another key difference between our framework and the exclusive feature learning methods with sparse representation and regularization, such as $\ell_{2,1}$-norm. In the proposed framework, we utilize these multi-modal networks and the sparse group lasso jointly to select the feature groups that are relevant to classification tasks.

III. Proposed Grouped Feature Selection Framework

In this section, we present our framework combining multi-modal deep neural networks with sparse group lasso (MMNNSGL) for grouped feature selection.

A. Model Architecture

Fig. 2 illustrates the architecture of the proposed feature integration and selection framework. The framework consists of two main modules: Multi-Modal Neural Networks and Feature Selection Component. In addition, a classifier is attached on the top for classification tasks in this paper. The core of the entire framework is the module of Multi-Modal Neural Networks which is responsible for extracting abstract feature representations. As shown in Fig. 2, this module includes multiple sub-networks, i.e. the low-level branches in the whole architecture. Similar to the MKL method where every modality is allocated a unique kernel, we assign heterogeneous sub-networks to different modalities These branch sub-networks differ from each other in the construction of hidden layers, whereas they share the same optimization criterion in the objective function layer. The Feature Selection Component aims to find the optimal weights for all the feature groups by solving the optimization problem with sparse group lasso. As a result, the features with small weights are dropped out. The top of the framework is the module of classifier, here SVM and logistic regression are often used.

Each independent modality is characterized by a single feature group, and then these different modalities are sent to different branches of the Multi-modal Neural Networks, yielding refined feature representations with multiple nonlinear transformations based upon the given original modalities. When all the feature groups are transformed by the multi-modal neural networks, the outputs of the refined features extracted from the top layer of each branch are concatenated into a new feature vector. Then the Feature Selection Component takes this concatenation as its input and derives an optimal solution of the weight vector. According to this weight vector, the most relevant feature groups with respect to the current task are picked out. Finally, we use these selected features in the final recognition task. In the following sections, we will describe and analyze the Multi-Modal Neural Networks and the Feature Selection Component in detail.

B. Heterogeneous Sub-Networks for Extracting Homogeneous Feature Representation

For many approaches of heterogeneous feature selection and multiple feature integration mentioned in previous sections, the primary obstacle that hinders them from getting better performance is the heterogeneity of the discriminative feature groups. Recently, deep learning has been widely applied in machine learning for its attractive ability in feature extraction and transformation. A deep learning algorithm is usually composed of multiple nonlinear transformations for projecting the original inputs into a new feature space. In our proposed framework, we use deep learning to extract homogeneous features with the heterogeneous sub-networks. Fig. 3 illustrates the intrinsic structure of the sub-networks. The Multi-Modal Neural Networks are composed of an Objective Function Layer at the top level, where a loss function is defined, and multiple sub-networks at the lower levels i.e. branches. In this way, the heterogeneous feature groups are cast into a unified representation where the heterogeneity across these groups is eliminated.

To process different data, several architectures have been developed to construct the internal structure of the deep neural networks, including deep neural networks (DNN) [40], deep belief networks (DBN) [41], stacked denoising autoencoders (SDA) [42], and convolutional neural networks (CNN) [43].

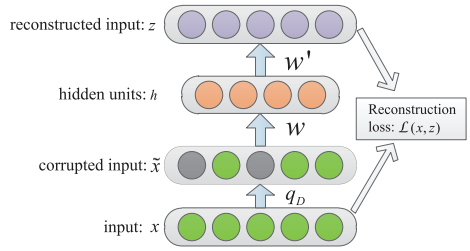


Fig. 4. Illustration of the denoising autoencoder. The inputs are corrupted by a corruption map q_D, and we reconstruct the original input from this corrupted $\tilde{x}$. We train all the hidden layer by minimizing the reconstruction error $\mathcal{L}$ between the uncorrupted input x and the reconstructed z.

With these deep architectures, different performance can be achieved for a variety of data sources. In addition, the final performance of these models is affected by the choice of the number of hidden layers and hidden nodes. Clearly, considering the intrinsic distribution of different modalities, it would be desirable to construct heterogeneous neural networks for the heterogeneous modalities. In our model, we regard each feature group as an independent modality and assign it one of the sub-networks (branches). The structures of the hidden layers and the number of hidden nodes are different. Moreover, in the sub-networks constructed for different modalities, we can exploit different basic deep architectures. For example, the CNN performs well in processing image raw data, while the SDA produces good performance for numerical data with noise. With these heterogeneous sub-networks, we can deal with multiple discriminative feature groups and train appropriate artificial neural networks for different modalities.

In this paper, we choose SDA as the base deep architecture for the sub-networks as the inputs are numerical vectors. There are two stages for training the networks: unsupervised pre-training stage and supervised fine-tuning stage. As shown in Fig. 3, denoising autoencoder is used to pre-train the sub-networks in the pre-training stage. The denoising autoencoder is a variant of autoencoder. Fig. 4 illustrates the basic idea of a denoising autoencoder. Denoising autoencoder corrupts the given original input vector $x \in \mathbb{R}^p$ into a noisy version $\tilde{x}$ by a corruption mapping of $\tilde{x} \sim q_D(\tilde{x}|x)$. Then the hidden units are encoded as $h = s(W\tilde{x} + b)$, where $s(\cdot)$ denotes a nonlinear transformation function, W is the weight matrix and b is the bias vector. The nonlinear function is often set as $s(t) = 1/(1 + e^{-t})$ or $s(t) = \tanh(t)$. Finally, the hidden unit h is decoded into the reconstruction of $z = s(W'h + b')$. The hidden layer is trained by optimizing the parameters of the model (W, W', b, b') such that the reconstruction error $\mathcal{L}(x, z)$ is minimized. According to the distribution assumption of the input, the reconstruction error $\mathcal{L}(x, z)$ is computed as either the traditional squared error

$$\mathcal{L}(x, z) = \|x - z\|^2 \tag{3}$$

or the cross-entropy function

$$\mathcal{L}(x, z) = -\sum_{i=1}^{p}[x_i \log z_i + (1 - x_i)\log(1 - z_i)]. \tag{4}$$

In this way, all the hidden layers are trained layer-wisely in the pre-training stage and the outputs of each trained layer are used as the inputs of the next layer in the training period.

However, our objective is to transform the discriminative feature groups into homogeneous feature representations. These low-level features convey different information of the same concept. It is not trivial to find the connection between them directly from a relatively low semantic level. Nevertheless, these heterogeneous modalities could be associated with each other easily from the higher concept level. Actually, the concept prior is contained within supervised information, such as labels, pair-wise similarity constraints. Therefore, an objective function layer, shared by all the sub-networks, is set at the top level of the Multi-Modal Neural Networks. We introduce this auxiliary layer to utilize the given labels and establish the intrinsic link among multiple modalities, which is expressed in the form of objective function to be optimized according to the current pattern recognition task. In the fine-tuning stage, this top layer is added into the sub-networks and all the parameters are fine-tuned with the back-propagation algorithm to minimize the loss function. Depending on the given recognition task, a variety of loss functions can be adopted. In this paper, the prediction error defined on the multi-class classification tasks is considered. Given a k-class classification task, we suppose the input sample x has p features totally. The top layer has the following parameters: the weight matrix of $W \in \mathbb{R}^{k \times p}$ and the bias vector of $b \in \mathbb{R}^k$. The loss function of the negative log-likelihood of the softmax regression is calculated as

$$\mathcal{L} = -\sum_{i=1}^{n}\sum_{j=1}^{k} 1\{y_j^{(i)} = 1\} \log \frac{e^{W_j x^{(i)} + b_j}}{\sum_{\ell=1}^{k} e^{W_\ell x^{(i)} + b_\ell}} \tag{5}$$

where $x^{(i)}$ denotes the i-th sample and $y^{(i)}$ is its label indicator. If the i-th sample belongs to class j, the corresponding indicator $y_j^{(i)} = 1$. W_j denotes the j-th row of the weight matrix W and $1\{\cdot\}$ is an indicator function whose value is 1 if the i-th sample belongs to the j-th class; otherwise, 0. For convenience, we slightly abuse the notation for x to denote the input of the objective function layer. The actual input is the latent representation $h_m^{n_m}$ extracted from the top layer of the sub-network, as shown in Fig. 3. We use gradient descent to minimize the loss and fine-tune every sub-network with back-propagation. Specifically, to avoid the interference across modalities, we connect each sub-network to the objective function layer with part of the nodes in this layer. In terms of implementation, we can pre-train and fine-tune different sub-networks separately. The only connection across modalities is the same concept prior (i.e. label information). This auxiliary layer is used only for fine-tuning all the networks and it is discarded once all the networks are well trained.

In this way, we fine-tune the whole sub-networks to yield high-level abstract feature representations for the classification task. After a series of non-linear transformations, these abstract features are able to express complex patterns. With this additional auxiliary layer in the fine-tuning stage, we combine the concept prior with deep generative learning. Meanwhile, we obtain the refined feature representations from the top layer of each branch sub-network, on a group-by-group basis. These new feature representations are concatenated as the input of the feature selection component.

C. Feature Group Evaluation and Selection

Since the module of Multi-Modal Neural Networks has transformed the original feature representations into multiple homogeneous vectors, the feature selection component will evaluate these feature groups and select the most relevant subsets for the current pattern recognition task. Accounting for the grouped property in the refined feature descriptors, we exploit the sparse group lasso for grouped feature selection. It should be noted that for presentation clarity, in this subsection, we abuse the notation for x to denote the refined features obtained in previous section.

Given a training set of $\{(x^{(i)}, y^{(i)} \in \mathbb{R}^p \times \{+1, -1\}^k; i = 1, 2, \ldots, n\}$ consisting of n samples of k classes, where $x^{(i)} = (x_1^{(i)}, x_2^{(i)}, \cdots, x_p^{(i)})^{\mathrm{T}} \in \mathbb{R}^p$ denotes the p-dimensional feature vector refined previously by the multi-modal networks for the i-th sample, $y^{(i)} = (y_1^{(i)}, y_2^{(i)}, \cdots, y_k^{(i)})^{\mathrm{T}} \in \{+1, -1\}^k$ is the corresponding label indicator, $y_k^{(i)} = +1$ if sample $x^{(i)}$ belongs to class k; otherwise, -1. Let $\mathrm{X} = (x^{(1)}, x^{(2)}, \cdots, x^{(n)})^{\mathrm{T}} \in \mathbb{R}^{n \times p}$ denote the training data matrix, and $\mathrm{Y} = (y^{(1)}, y^{(2)}, \cdots, y^{(n)})^{\mathrm{T}} \in \{+1, -1\}^{n \times k}$ be the label indicator matrix. Suppose the p-dimensional feature vector is divided into g non-overlapping groups and G_ℓ denotes the size of the ℓ-th feature group. Define $\beta_j = (\beta_{j1}^{\mathrm{T}}, \beta_{j2}^{\mathrm{T}}, \cdots, \beta_{j\ell}^{\mathrm{T}})^{\mathrm{T}} \in \mathbb{R}^p$ as the coefficient vector for label j, where $\beta_{j\ell}$ is the corresponding coefficient subvector of group ℓ, and $\mathrm{X}_\ell \in \mathbb{R}^{n \times G_\ell}$ as the features of the training data corresponding to the ℓ-th group. The grouped feature selection problem for the j-th label indicator can be formulated as the following optimization task:

$$\mathcal{S}(\beta_j) = \min_{\beta_j} \mathcal{L}(\beta_j) + \mathcal{R}(\beta_j) \tag{6}$$

where $\mathcal{L}(\beta_j)$ is the loss function, and $\mathcal{R}(\beta_j)$ is the regularization. According to the training data and the specific task, the loss function could take different forms. In this paper we consider the task of image classification and therefore the logistic loss is applied

$$\mathcal{L}(\beta_j) = \sum_{i=1}^{n} \log(1 + \exp[-\mathrm{Y}_{(i,j)}(\beta_j^{\mathrm{T}} x^{(i)} + c)]) \tag{7}$$

where c is the intercept. The regularization $\mathcal{R}$ in (6) is formulated as

$$\mathcal{R}(\beta_j) = \lambda_1 \|\beta_j\|_1 + \lambda_2 \sum_{\ell=1}^{g} w_\ell \|\beta_{j\ell}\|_2 \tag{8}$$

where λ_1 and λ_2 are regularization parameters, and the hyper-parameter w_ℓ is the weight of feature group ℓ and always set as the squared root of the feature group size G_ℓ.

The regularization of (8) includes two parts: common L1-norm penalty and an additional penalty which encourages sparsity on the group level of features. In other words, the regularization in (8) leads to sparsity in both inter-group and intra-group features. Not only some feature groups but also some features within the same group are discarded if their weights are zero. The features whose weights are nonzeros are selected.

Fig. 5. Sample images from the adopted datasets.

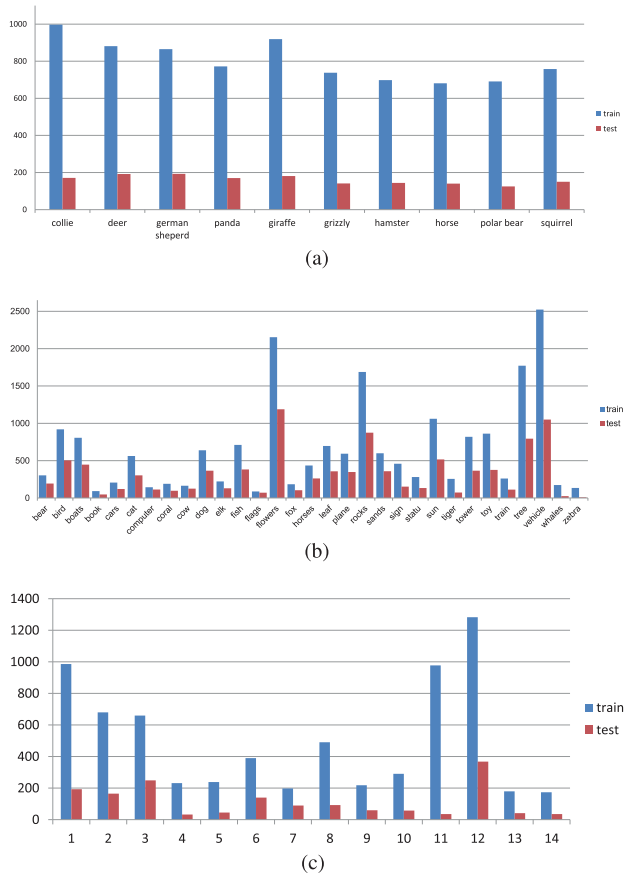


Fig. 6. Number of images in different classes in the adopted datasets. (a) *Animal-10*. (b) *NUS-WIDE-Object*. (c) *MSRA-MM*.

Let $f(\beta, c)$ denote the logistic loss in (7), and define $\phi(\beta)$ as the penalty terms of (8). The optimization problem could be defined as a new form

$$\min_{\beta} f(\beta, c) + \lambda\phi(\beta). \tag{9}$$

Treating the penalty $\phi(\beta)$ as a Moreau-Yosida regularization, Liu and Ye [44] proposed an efficient algorithm to solve the optimization above and provided related lemmas and detailed proofs in their paper. We exploit the implementation of this method provided in the toolbox of SLEP.[1] At each iteration, it only needs to evaluate the function value and the gradient. The algorithm converges with a linear time complexity, thus it could process large-scale data efficiently.

[1][Online]. Available: http://www.public.asu.edu/jye02/Software/SLEP/

TABLE I
FEATURE GROUPS OF ANIMAL-10 DATASET

	Feature Group	Description	Dimensionality
1	LSS	Local Self-Similarity features [45].	2000
2	RGSIFT	rgSIFT descriptors [46].	2000
3	SIFT	SIFT descriptors [47].	2000
4	SURF	SURF descriptors [48].	2000
5	DeCAF	Deep Convolutional Activation Feature [49].	4096
6	CQ	Color Histogram features.	2688
7	PHOG	Pyramid Histogram of Oriented Gradients [50].	252
8	Gaussian	Random Gaussian noise obeys distribution of $N(0,1)$.	100
9	Uniform	Random noise obeys uniform distribution $U(0,1)$.	100
10	Chi2	Random noise obeys Chi Square distribution $\chi^2(1)$.	100
11	F-dist	Random noise obeys F-distribution $F(4,4)$.	100
12	Beta	Random noise obeys Beta-distribution $Be(0.5,0.5)$.	100
13	LSS+N	Some Gaussian noise is added to the original LSS features.	2000
14	RGSIFT+N	Some Gaussian noise is added to the original RGSIFT features.	2000

TABLE II
FEATURE GROUPS OF NUS-WIDE-OBJECT DATASET

	Feature Group	Description	Dimensionality
1	CH	Color Histogram features.	64
2	CORR	Color auto-correlogram [51].	144
3	EDH	Edge Direction Histogram [52].	73
4	WT	Wavelet Texture [53].	128
5	CM	Block-wise Color Moments [54].	225
6	TextLDA1	Doc-Topic distribution of LDA Topic model with 31 topics.	31
7	TextLDA2	Doc-Topic distribution of LDA Topic model with 81 topics.	81
8	Gaussian	Random Gaussian noise obeys distribution of $N(0,1)$.	100
9	Uniform	Random noise obeys uniform distribution $U(0,1)$.	100
10	Chi2	Random noise obeys Chi Square distribution $\chi^2(1)$.	100
11	F-dist	Random noise obeys F-distribution $F(4,4)$.	100
12	Beta	Random noise obeys Beta-distribution $Be(0.5,0.5)$.	100
13	CH+N	Some Gaussian noise is added to the original CH features.	64
14	CORR+N	Some Gaussian noise is added to the original CORR features.	144

Algorithm 1 Feature Group Evaluation with Refined Feature Representations

Input: Training set $\mathrm{X} = [x^{(1)}, x^{(2)}, \cdots, x^{(n)}]^{\mathrm{T}} \in \mathbb{R}^{n \times p}$,
Label descriptors matrix
$\mathrm{Y} = [y^{(1)}, y^{(2)}, \cdots, y^{(n)}]^{\mathrm{T}} \in \{+1, -1\}^{n \times k}$.
Output: Weight vector of all features $\beta \in \mathbb{R}^p$,
Importance vector of feature groups
$\sigma = [\sigma_1, \sigma_2, \cdots, \sigma_g] \in \mathbb{R}^g$.

For label $j = 1 : k$
$\beta_j := \arg\min_{\beta_j} \mathcal{S}(\beta_j)$
End For
$\beta := \sum_{i=1}^{k} |\beta_i| \cdot p(x = i)$
For each feature group i
$\sigma_i = (\sum_{j=1}^{G_i} \beta_{G_j^i})/G_i$
End For

Algorithm 1 describes the procedure for evaluating the importance of feature groups output from the multi-modal neural networks. In Algorithm 1, $p(x = i)$ denotes the probability that the sample x belongs to class i. Each time we get one weight vector β_j for the corresponding label indicator j by solving the optimization problem. However, there exist multiple label indicators. To evaluate the relevance of the feature to the current task for all the labels, we introduce another weight $p(x = i)$ for each label and obtain the final weight vector β for the features. Then we obtain the importance for each feature group

$$\sigma_i = \left(\sum_{j=1}^{G_i} \beta_{G_j^i} \right) / G_i \tag{10}$$

where $\beta_{G_j^i}$ denotes the j-th feature in the i-th group and G_i is the size of Group i. According to this importance vector, the feature groups with nonzero weights are selected and they are considered more relevant to the current task. These features are used for the final recognition task. At the same time, if the sparsity parameter $\lambda_1 \neq 0$, some features within the same group are also left out in order to improve the efficiency of the model.

IV. EXPERIMENTS

In this section, we demonstrate the effectiveness of the proposed framework for classification tasks with three real-world image recognition dataset.

A. Data Description

The three datasets we used are: Animal with Attributes, NUS-WIDE-Object and MSRA-MM 2.0. Fig. 5 presents some sample

TABLE III
FEATURE GROUPS OF MSRA-MM DATASET

	Feature Group	Description	Dimensionality
1	CH	RGB Color Histogram features.	256
2	CORR	Color auto-correlogram.	144
3	EDH	Edge Direction Histogram.	75
4	WT	Wavelet Texture.	128
5	CM	Block-wise Color Moments.	225
6	HSV	HSV color histogram.	64
7	Gaussian	Random Gaussian noise obeys distribution of $N(0,1)$.	100
8	Uniform	Random noise obeys uniform distribution $U(0,1)$.	100
9	Chi2	Random noise obeys Chi Square distribution $\chi^2(1)$.	100
10	F-dist	Random noise obeys F-distribution $F(4,4)$.	100
11	Beta	Random noise obeys Beta-distribution $Be(0.5,0.5)$.	100
12	CH+N	Some Gaussian noise is added to the original CH features.	256
13	CORR+N	Some Gaussian noise is added to the original CORR features.	144

TABLE IV
CLASSIFICATION ACCURACIES WITH INDIVIDUAL FEATURE GROUP OF ANIMAL-10

Features/Methods	SVM	DNN	MMNN+SVM
LSS	0.4941	0.5413	**0.5507**
RGSIFT	0.5022	**0.5331**	0.5314
SIFT	0.4088	**0.4450**	0.4381
SURF	0.5271	**0.5669**	0.5644
DeCAF	0.8034	0.8388	**0.8401**
CQ	0.3920	0.4738	**0.4773**
PHOG	0.3765	**0.3819**	0.3684
Gaussian	0.1064	0.1213	0.1070
Uniform	0.1120	0.1244	0.1101
Chi2	0.1058	0.1288	0.1164
F-dist	0.1064	0.1206	0.1089
Beta	0.1120	0.1244	0.1070
LSS+N	0.3161	0.3956	**0.4026**
RGSIFT+N	0.4070	0.4000	**0.4138**

TABLE V
CLASSIFICATION ACCURACIES WITH INDIVIDUAL FEATURE GROUP OF NUS-WIDE-OBJECT

Features/Methods	SVM	DNN	MMNN+SVM
CH	0.2426	0.3018	**0.3032**
CORR	0.3066	0.3600	**0.3738**
EDH	0.2894	**0.3077**	0.3074
WT	0.3063	0.3684	**0.3733**
CM	0.2857	0.3331	**0.3394**
TextLDA1	0.5062	0.5403	**0.5433**
TextLDA2	0.4564	0.5495	**0.5553**
Gaussian	0.1041	0.1205	0.1050
Uniform	0.1044	0.1199	0.1050
Chi2	0.1021	0.1188	0.1050
F-dist	0.1067	0.1200	0.1050
Beta	0.1052	0.1188	0.1050
CH+N	0.2221	0.2235	**0.2238**
CORR+N	0.2803	**0.2953**	0.2940

TABLE VI
CLASSIFICATION ACCURACIES WITH INDIVIDUAL FEATURE GROUP OF MSRA-MM

Features/Methods	SVM	DNN	MMNN+SVM
CH	0.2744	**0.3157**	0.2910
CORR	0.3236	**0.3648**	0.3404
EDH	0.2570	0.2700	**0.2719**
WT	0.3030	0.3316	**0.3329**
CM	0.2844	**0.3308**	0.3018
HSV	0.2775	0.2811	**0.2974**
Gaussian	0.1886	0.2327	0.2289
Uniform	0.1780	0.2321	0.2280
Chi2	0.1879	0.2314	0.2289
F-dist	0.1730	0.2333	0.2289
Beta	0.1680	0.2327	0.2290
CH+N	0.2321	**0.2931**	0.2489
CORR+N	0.2626	**0.3176**	0.2825

images of those datasets. Note that, for MSRA-MM 2.0, we are unable to provide the raw images here, as it is already closed and we have only got its image feature matrix.

- *Animal-10 dataset*
  The Animal dataset[2] contains 30475 images of 50 animal classes from Flickr and Google Picasa. We select 9607 images of 10 classes from the 50 animal classes for image classification and rename this subset as Animal-10. The number of each class in Animal-10 is presented in Fig. 6(a). We randomly take 8000 images for training and the remaining 1607 images for testing. Table I lists all the feature groups we adopted in our experiments including seven commonly used meaningful feature groups and additional seven noisy feature groups. We obtain seven types of commonly used feature descriptors for every image of this dataset (1-7 in Table I). To demonstrate the effectiveness of the proposed framework for filtering irrelevant features, we add another five different types of noise groups (8-12 in Table I). We also add some Gaussian noise to two original feature groups, with the noise added here following a normal distribution of $N(0,0.2)$.

- *NUS-WIDE-Object Dataset*
  The NUS-WIDE-Object dataset[3] consists of 30000 images from Flickr. Text description tags are attached to every image by the authors of the photos. These 30 thousands images are classified into 31 classes and the number of images

[2][Online]. Available: http://attributes.kyb.tuebingen.mpg.de/

[3][Online]. Available: http://lms.comp.nus.edu.sg/research/NUS-WIDE.htm

TABLE VII
CLASSIFICATION ACCURACIES COMPARISION ON THREE IMAGE CLASSIFICATION TASKS

Datasets/Methods	SVM	MKL	MtBGS	GLLR+SVM	MMNN+SVM	MMNNSGL+SVM
Animal-10	0.6671	0.8419	0.8451	0.6416	0.8544	**0.8675**
NUS-WIDE-Object	0.3654	0.5954	0.6064	0.3755	0.6381	**0.6482**
MSRA-MM	0.3373	0.3945	0.3118	0.3385	0.3951	**0.4095**

in every class is presented in Fig. 6(b). We take 20000 images randomly as the training set and the remaining 10000 images as the test set. Table II lists all 14 feature groups we adopted for this dataset. Five commonly used feature descriptors are extracted from this dataset (1-5 in Table II). Besides, we extract two document feature representations [55] from the text photo descriptions (6,7 in Table II). Same as the Animal-10 dataset, we get extra seven noisy feature groups for this dataset and adopt them in our experiments (8-14 in Table II).

- *MSRA-MM Dataset*
  The MSRA-MM dataset[4] includes 1 million images collected from Microsoft Live Search. There are 50000 labelled images categorized into 100 concepts. We choose 8607 images of 14 classes from the 100 classes for image classification. Fig. 6(c) shows the number of images from each class in this subset. We randomly take 7000 images for training and use the remaining in test. This dataset provides six types of visual features for each image. Table III lists its feature groups. Similarly, we add extra seven noisy feature groups into this dataset in our experiments (7-13 in Table III).

B. Experimental Setup

We apply our MMNNSGL framework to the three datasets mentioned above and compare the proposed method against four other methods: (1) SVM with the concatenation of all original multiple feature groups, (2) MKL method [4], (3) MtBGS [19], (4) Group Lasso with Logistic Regression (GLLR) [20]. We denote the method of using support vector machine classifier with the original feature concatenation by SVM and take its performance as a baseline. For method of GLLR, we utilize group lasso for logistic regression to select grouped features from the original features and use SVM to classify the test set with the selected features. Similarly, we take SVM as the basic classifier of our framework. We denote these two methods by GLLR+SVM and MMNNSGL+SVM, respectively.

In addition, to demonstrate the feature extraction ability of the multi-modal neural networks, we present the classification performance of three basic methods with individual feature group as baselines: (1) SVM with the original individual feature group, (2) SVM with the refined features, (3) the logistic regression classifier with refined features. We denote them by SVM, MMNN+SVM and DNN, respectively. The logistic regression classifier is attached to the objective function layer of the multi-modal neural networks in our proposed framework. As to SVM, we randomly take a small validation set and seek the optimal kernel and corresponding parameters according to the classification accuracy on the validation set. This procedure is similar to cross-validation. We have tested the RBF kernel and linear kernel for every modality. We search the optimal parameters σ and c in the range of [0.01, 5] and [0.1, 10] respectively.

We implemented the MKL learning algorithm for multi-class classification on the foundation of simpleMKL. We also implemented the key algorithm of MtBGS and applied it to single label multi-class image classification tasks. For MKL method, an independent kernel was set for each individual feature group. We have tried some different kernels including RBF kernel, Polynomial kernel and linear kernel. We selected the kernel for every modality according to the classification performance on a small validation set. Then a relatively optimal kernel was allocated to each modality. In the method of MtBGS, the parameters (λ_1, λ_2) are optimized in the range of 10^{-4} to 1 with a step of 0.005. Similarly, the parameters of sparse group lasso exploited in our feature selection component are tuned in the same range of values.

The GLLR and MtBGS are implemented with the sparse learning package of SLEP. The SVM in all our experiments is implemented with the LIBSVM[5] software package. We implemented the multi-modal neural networks with the deep learning library of Theano.[6] Considering the computational demand for training the multi-modal neural networks, we run our algorithm on GPU to accelerate the training procedure.

C. Experimental Results

First we evaluate the proposed framework using different single feature groups to verify the capability of multi-modal neural networks in feature extraction. Tables IV,V and VI show the classification accuracies with different features on three adopted datasets. The bold numbers denote the best accuracy of the compared methods.

Obviously, for all the different feature groups except for the random noise groups, using refined features achieve better performance than using the original features. For those noisy feature groups obtained by mixing the original features with Gaussian noise, using the refined features extracted by the sub-networks of our multi-modal neural networks gives much better classification accuracies than using the un-refined features. This also demonstrates the denoising effectiveness of the deep neural networks. The results also confirm that through the sub-networks we have obtained better feature representations for each individual feature group. We also notice that with the same refined feature representation, using SVM usually gets

[4][Online]. Available: http://research.microsoft.com/en-us/projects/msrammdata/

[5][Online]. Available: http://www.csie.ntu.edu.tw/%7ecjlin/libsvm/

[6][Online]. Available: http://deeplearning.net/software/theano/

better classification performance than the method of logistic regression. For this reason, we set the SVM as the basic classifier for our framework finally.

On the other hand, from the results we can see that these noisy features can have negative impact on the final classification accuracy if we use multiple feature groups jointly for the task. To demonstrate the ability of the proposed framework for filtering the irrelevant features, we conduct a series of experiments and compare our method with other methods. The results are shown in Table VII. We also give the results of the method that uses SVM to classify the test set with the refined representations of multi-modal networks without feature selection. The method is denoted as MMNN+SVM in Table VII.

Empirical evidence shows that our framework outperforms other methods significantly in all the three datasets. At the same time, we can also find that those noisy feature groups deteriorate the performance of the models to certain degrees. For the same SVM, using some single type of feature is better than using all the feature groups.

To show the effectiveness of the feature selection in the compared methods, we investigate the weights of the features obtained by different methods. For the MKL method, the algorithm yields a weight for the kernel of each feature group. However, the methods of MtBGS, GLLR and our MMNNSGL yield a weight corresponding to each single dimensionality in the feature groups. We calculate the weight for each feature group defined as σ in Algorithm 1 to compare the effectiveness of these methods using feature selection. Fig. 7 shows the weight for each feature group obtained by different methods on the three datasets, respectively.

It can be observed that our method can effectively filter the noisy feature groups that are deemed to be irrelevant to the final classification task. All the random noise and the noisy original feature groups are weighted zero in all the three datasets. It endows only those feature groups that are relevant and informative to the classification task with a proper value. In contrast, the other three methods assign incorrect weights to those irrelevant features because of the distinction of the heterogeneous features. For example, with the GLLR method using the original features, high weights have been assigned to the group of random noise. Even though the MKL method assigns every feature group with a different weight, it cannot select those feature groups that are more relevant to the classification task. We notice that the results of MtBGS are close to ours. For the dataset of Animal-10, MtBGS assigns a zero weight to the random noise group but a relatively high weight to the noisy feature group. For the NUS-WIDE-Object dataset, MtBGS sets the weights for all the feature groups to nonzeros. With higher sparsity coefficients, MtBGS can filter out most of the feature groups, nevertheless it gives a poor classification performance.

After we obtain the importance weights of the individual feature groups, a feature group is selected if its weight is larger than a predefined threshold (10^{-4} in our experiments). We believe that the importance weight has the potential to quantify the relevance of each feature group to the given task. To verify this hypothesis, we conduct extra experiments. For each dataset, we sort the feature groups by the importance value in a descending order. We then add each single feature group into a concatenated group feature vector one by one. The test set is then used to evaluate the classification performance of using the concatenated feature groups with the SVM classifier. The results are illustrated in Fig. 8. The red triangle denotes the position where

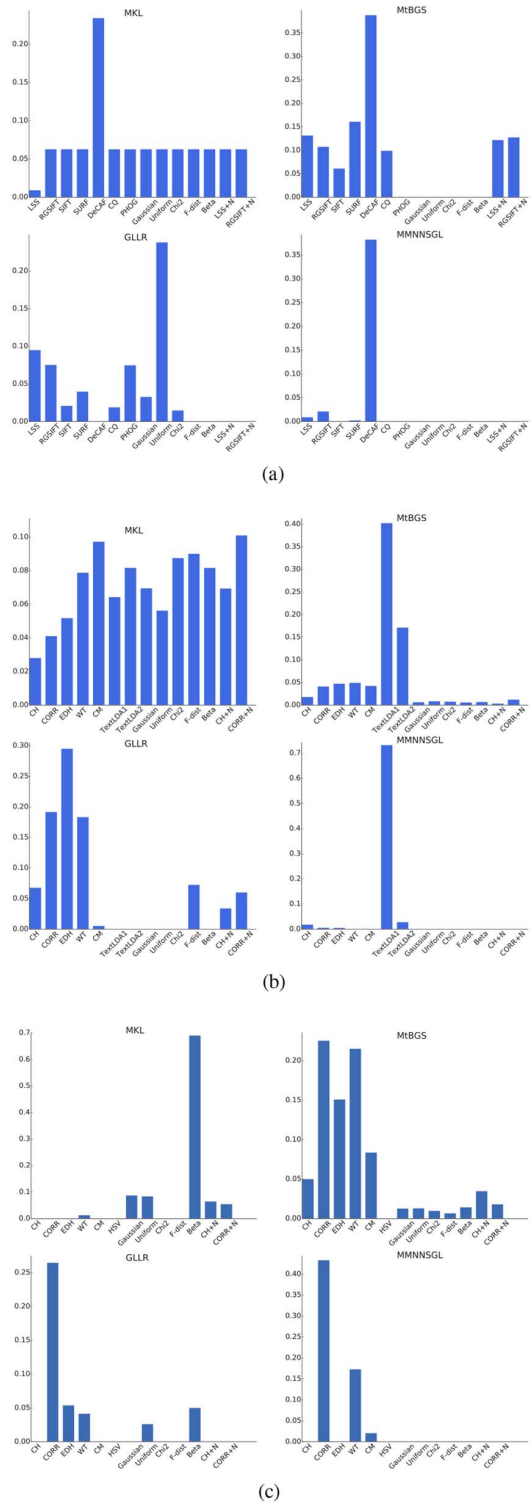


Fig. 7. Weights of different feature groups obtained by the methods compared, which, from top left to bottom right, are the MKL, MtBGS, GLLR, and the proposed method, respectively. (a) *Animal-10*. (b) *NUS-WIDE-Object*. (c) *MSRA-MM*.

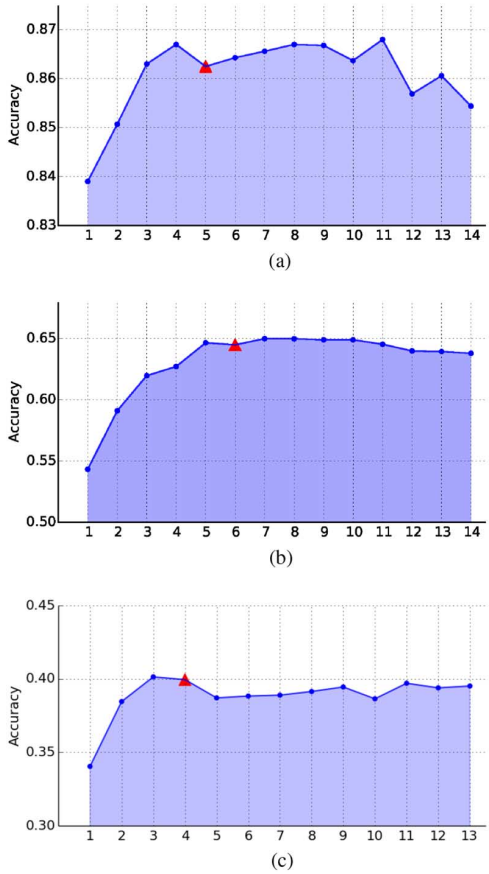


Fig. 8. Classification accuracy changes with respect to the inclusion of the feature group with lower weights. We sort the feature groups shown in Tables I, II, and III, respectively, by their importance values derived from our MMNNSGL. The number on the horizontal axis shows the indices of the sorted refined feature groups. The red triangle denotes the position from which the feature group with zero weights is included. (a) *Animal-10*. (b) *NUS-WIDE-Object*. (c) *MSRA-MM*.

the feature groups with zero weights are added. It is evident that we could even achieve a relatively better performance by using the feature descriptors from only a few feature groups. Those groups with zero weights have little effect on the classification performance of our model. We can also see that with the help of deep networks in our model, the overall performance drops only slightly in spite of adding irrelevant feature groups. This further confirms the effectiveness of feature refinement with our multi-modal neural networks.

All the empirical results show that the proposed framework can transform the original heterogeneous features into a new form that possesses a better discrimination ability. The MMNNSGL method has the ability to select those features that are more relevant to the image classification tasks considered.

V. Conclusion and Future Work

We have presented a method for combining deep neural networks with sparse representation and proposed the Multi-Modal Neural Networks with Sparse Group Lasso framework for grouped heterogeneous feature selection. Different from some existing methods applied to multiple feature integration, such as MKL and Group Lasso based methods, the proposed framework exploits the distinction among the heterogeneous features and their different importance for the considered recognition tasks. The main advantage of the proposed framework lies in the powerful ability in feature transformation. With the multi-modal neural networks, a new unified representation is extracted from each original feature group where the heterogeneity across the groups is eliminated. An extended method of sparse group lasso is used to learn the weight or importance of each feature with the unified feature representations. Finally, the most relevant features are picked out for the given recognition tasks. We have evaluated our framework on three real world datasets for image classification. Experimental results have demonstrated the improved performance of our approach in grouped feature selection, multiple feature integration, and classification accuracy, as compared with several baseline methods.

Despite the fact that we only applied the proposed MMNNSGL framework to the single-label multi-class classification problem, it could be further extended to other tasks such as multi-label categorization or retrieval tasks. For these tasks, the loss function may have to be defined in a different way according to their properties. On the other hand, we have obtained the weights of each feature group and selected the feature groups with weights of high values. However, the information of their importance has not yet been exploited to improve the classification performance, which is a question worth further studying.

Acknowledgment

The authors gratefully acknowledge the support of NVIDIA Corporation with the donation of the Tesla K40 GPU used for this research.

References

[1] A. C. Berg, T. L. Berg, and J. Malik, "Shape matching and object recognition using low distortion correspondences," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recog.*, Jun. 2005, vol. 1, pp. 26–33.

[2] O. Duchenne, F. Bach, I.-S. Kweon, and J. Ponce, "A tensor-based algorithm for high-order graph matching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 12, pp. 2383–2395, Dec. 2011.

[3] J. Yang, J.-Y. Yang, D. Zhang, and J.-F. Lu, "Feature fusion: Parallel strategy vs. serial strategy," *Pattern Recog.*, vol. 36, no. 6, pp. 1369–1381, 2003.

[4] A. Rakotomamonjy *et al.*, "Simplemkl," *J. Mach. Learn. Res.*, vol. 9, pp. 2491–2521, 2008.

[5] Y.-Y. Lin, T.-L. Liu, and C.-S. Fuh, "Local ensemble kernel learning for object category recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2007, pp. 1–8.

[6] A. Vedaldi, V. Gulshan, M. Varma, and A. Zisserman, "Multiple kernels for object detection," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Sep.–Oct. 2009, pp. 606–613.

[7] M. Guillaumin, J. Verbeek, and C. Schmid, "Multimodal semi-supervised learning for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2010, pp. 902–909.

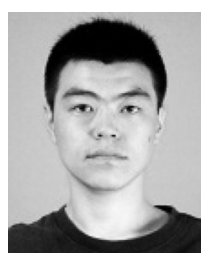
[8] G.-J. Qi, C. Aggarwal, Q. Tian, H. Ji, and T. S. Huang, "Exploring context and content links in social media: A latent space method," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 5, pp. 850–862, May 2012.

[9] G.-J. Qi, M.-H. Tsai, S.-F. Tsai, L. Cao, and T. S. Huang, "Web-scale multimedia information networks," *Proc. IEEE*, vol. 100, no. 9, pp. 2688–2704, Sep. 2012.

[10] G.-J. Qi, C. Aggarwal, and T. S. Huang, "Breaking the barrier to transferring link information across networks," *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 7, pp. 1741–1753, Jul. 1, 2015.

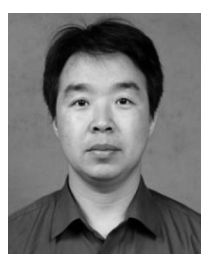
[11] Y. Yang *et al.*, "Multi-feature fusion via hierarchical regression for multimedia analysis," *IEEE Trans. Multimedia*, vol. 15, no. 3, pp. 572–581, Apr. 2013.

[12] H. Wang, F. Nie, H. Huang, and C. Ding, "Heterogeneous visual features fusion via sparse multimodal machine," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2013, pp. 3097–3102.
[13] S. Shekhar, V. M. Patel, N. M. Nasrabadi, and R. Chellappa, "Joint sparse representation for robust multimodal biometrics recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 1, pp. 113–126, Jan. 2014.
[14] J. Ngiam *et al.*, "Multimodal deep learning," in *Proc. 28th Int. Conf. Mach. Learn.*, 2011, pp. 689–696.
[15] N. Srivastava and R. Salakhutdinov, "Multimodal learning with deep boltzmann machines," *Adv. Neural Inf. Process. Syst.*, vol. 25, pp. 2222–2230, 2012.
[16] Y. Zhou, Q. Hu, J. Liu, and Y. Jia, "Combining multi-modal deep neural networks with conditional random fields for Chinese dialogue act recognition," *Neurocomput.*, vol. 168, pp. 408–417, 2015.
[17] P. Wu *et al.*, "Online multimodal deep similarity learning with application to image retrieval," in *Proc. 21st ACM Int. Conf. Multimedia*, 2013, pp. 153–162.
[18] Y. Han, F. Wu, J. Jia, Y. Zhuang, and B. Yu, "Multi-task sparse discriminant analysis (MtSDA) with overlapping categories," in *Proc. AAAI*, 2010, pp. 469–474.
[19] F. Wu, Y. Han, Q. Tian, and Y. Zhuang, "Multi-label boosting for image annotation by structural grouping sparsity," in *Proc. Int. Conf. on Multimedia*, 2010, pp. 15–24.
[20] L. Meier, S. Van De Geer, and P. Bühlmann, "The group lasso for logistic regression," *J. Roy. Statist. Soc., Ser. B (Statist. Methodol.)*, vol. 70, no. 1, pp. 53–71, 2008.
[21] Z. Ma, Y. Yang, F. Nie, J. Uijlings, and N. Sebe, "Exploiting the entire feature space with sparsity for automatic image annotation," in *Proc. 19th ACM Int. Conf. Multimedia*, 2011, pp. 283–292.
[22] Y. Yang, H. T. Shen, Z. Ma, Z. Huang, and X. Zhou, "l2, 1-norm regularized discriminative feature selection for unsupervised learning," in *Proc. Int. Joint Conf. Artificial Intell.*, 2011, vol. 22, no. 1, p. 1589.
[23] Y. Yang, Z. Ma, A. G. Hauptmann, and N. Sebe, "Feature selection for multimedia analysis by sharing information among multiple tasks," *IEEE Trans. Multimedia*, vol. 15, no. 3, pp. 661–669, Apr. 2013.
[24] Q. Hu, D. Yu, J. Liu, and C. Wu, "Neighborhood rough set based heterogeneous feature subset selection," *Inf. Sci.*, vol. 178, no. 18, pp. 3577–3594, 2008.
[25] D. Cai, X. He, and J. Han, "Semi-supervised discriminant analysis," in *Proc. IEEE 11th Int. Conf. Comput. Vis.*, Oct. 2007, pp. 1–7.
[26] X. Niyogi, "Locality preserving projections," *Adv. Neural Inf. Process. Syst.*, vol. 16, pp. 153–160, 2004.
[27] F. R. Bach, G. R. Lanckriet, and M. I. Jordan, "Multiple kernel learning, conic duality, and the smo algorithm," in *Proc. 21st Int. Conf. Mach. Learn.*, 2004, pp. 6–13.
[28] A. Rakotomamonjy, F. Bach, S. Canu, and Y. Grandvalet, "More efficiency in multiple kernel learning," in *Proc. 24th Int. Conf. Mach. Learn.*, 2007, pp. 775–782.
[29] Y.-Y. Lin, T.-L. Liu, and C.-S. Fuh, "Multiple kernel learning for dimensionality reduction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 6, pp. 1147–1160, Jun. 2011.
[30] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Roy. Statist. Soc., Ser. B (Methodological)*, vol. 58, pp. 267–288, 1996.
[31] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210–227, Feb. 2009.
[32] A. Argyriou, T. Evgeniou, and M. Pontil, "Convex multi-task feature learning," *Mach. Learn.*, vol. 73, no. 3, pp. 243–272, 2008.
[33] J. Liu, S. Ji, and J. Ye, "Multi-task feature learning via efficient l2,1-norm minimization," in *Proc. 25th Conf. Uncertainty Artificial Intell.*, 2009, pp. 339–348.
[34] M. Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables," *J. Roy. Statist. Soc., Ser. B (Statist. Methodol.)*, vol. 68, no. 1, pp. 49–67, 2006.
[35] F. Wu, Y. Yuan, and Y. Zhuang, "Heterogeneous feature selection by group lasso with logistic regression," in *Proc. Int. Conf. Multimedia*, 2010, pp. 983–986.
[36] J. Friedman, T. Hastie, and R. Tibshirani, "A note on the group lasso and a sparse group lasso," *CoRR*, vol. arXiv:1001.0736, 2010 [Online]. Available: http://arxiv.org/abs/1001.0736
[37] J. Peng *et al.*, "Regularized multivariate regression for identifying master predictors with application to integrative genomics study of breast cancer," *Ann. Appl. Statist.*, vol. 4, no. 1, pp. 53–77, 2010.
[38] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.
[39] W. Ouyang, X. Chu, and X. Wang, "Multi-source deep learning for human pose estimation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2014, pp. 2337–2344.
[40] H. Larochelle, Y. Bengio, J. Louradour, and P. Lamblin, "Exploring strategies for training deep neural networks," *J. Mach. Learn. Res.*, vol. 10, pp. 1–40, 2009.
[41] G. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, 2006.
[42] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *J. Mach. Learn. Res.*, vol. 11, pp. 3371–3408, 2010.
[43] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Adv. Neural Inf. Process. Syst.*, pp. 1097–1105, 2012.
[44] J. Liu and J. Ye, "Moreau-Yosida regularization for grouped tree structure learning," *Adv. Neural Inf. Process. Syst.*, pp. 1459–1467, 2010.
[45] S. Eli and I. Michal, "Matching local self-similarities across images and videos," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2007, pp. 1–8.
[46] K. E. Van De Sande, T. Gevers, and C. G. Snoek, "Evaluating color descriptors for object and scene recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1582–1596, Sep. 2010.
[47] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.
[48] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *Proc. Eur. Conf. Comput. Vis.*, 2006, pp. 404–417.
[49] J. Donahue *et al.*, "Decaf: A deep convolutional activation feature for generic visual recognition," in *Proc. 31st Int. Conf. Mach. Learn.*, 2014, pp. 647–655.
[50] A. Bosch, A. Zisserman, and X. Munoz, "Representing shape with a spatial pyramid kernel," in *Proc. 6th ACM Int. Conf. Image Video Retrieval*, 2007, pp. 401–408.
[51] J. Huang, S. R. Kumar, M. Mitra, W.-J. Zhu, and R. Zabih, "Image indexing using color correlograms," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 1997, pp. 762–768.
[52] D. K. Park, Y. S. Jeon, and C. S. Won, "Efficient use of local edge histogram descriptor," in *Proc. ACM Workshops Multimedia*, 2000, pp. 51–54.
[53] B. S. Manjunath and W.-Y. Ma, "Texture features for browsing and retrieval of image data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 8, pp. 837–842, Aug. 1996.
[54] M. A. Stricker and M. Orengo, "Similarity of color images," in *IS&T/SPIE Symp. Electron. Imaging: Sci. Technol. Int. Soc. Opt. Photon.*, 1995, pp. 381–392.
[55] L. Yang *et al.*, "Cqarank: Jointly model topics and expertise in community question answering," in *Proc. 22nd ACM Int. Conf. Conf. Inf. Knowl. Manage.*, 2013, pp. 99–108.



Lei Zhao received the B.S. degree in computer science and technology from Tianjin University, Tianjin, China, in 2012, and is currently working toward the M.E. degree in computer technology engineering at Tianjin University.

His current research interests include pattern recognition, machine learning, neural networks, and computer vision.



Qinghua Hu (M'10–SM'13) received the B.S., M.S., and Ph.D. degrees from the Harbin Institute of Technology, Harbin, China, in 1999, 2002, and 2008, respectively.

He was a Post-Doctoral Fellow with the Department of Computing, Hong Kong Polytechnic University, Hong Kong, from 2009 to 2011. He is currently a Full Professor and the Vice Dean of the School of Computer Science and Technology with Tianjin University, Tianjin, China. He has authored over 100 journal and conference papers in the areas of granular computing-based machine learning, reasoning with uncertainty, pattern recognition, and fault diagnosis. His research interests include rough sets, granular computing, and data mining for classification and regression.

Prof. Hu was the Program Committee Co-Chair of the International Conference on Rough Sets and Current Trends in Computing in 2010, the Chinese Rough Set and Soft Computing Society in 2012 and 2014, the International Conference on Rough Sets and Knowledge Technology in 2014, and the International Conference on Machine Learning and Cybernetics in 2014. He was the General Co-Chair of IJCRS 2015, and currently serves as a Referee for a great number of journals and conferences.

Wenwu Wang (M'02–SM'11) received the B.Sc. degree in automatic control, M.E. degree in control science and control engineering, and Ph.D. degree in navigation guidance and control from Harbin Engineering University, Harbin, China, in 1997, 2000, and 2002, respectively.

He joined Kings College, London, U.K., in May 2002, as a Postdoctoral Research Associate and transferred to Cardiff University, Cardiff, U.K., in January 2004. In May 2005, he joined the Tao Group Ltd. (now Antix Labs Ltd.), Reading, U.K., as a DSP Engineer. In September 2006, he joined Creative Labs Ltd., Egham, U.K., as an R&D Engineer. Since May 2007, he has been with the Centre for Vision Speech and Signal Processing, University of Surrey, Guildford, U.K., where he is currently a Reader in Signal Processing and a Co-Director of the Machine Audition Laboratory. In 2008, he was a Visiting Scholar with the Perception and Neurodynamics Laboratory and the Center for Cognitive Science, The Ohio State University, Columbus, OH, USA. He has authored or coauthored over 150 publications in these areas, including *Machine Audition: Principles, Algorithms and Systems* (IGI Global, 2010) and *Blind Source Separation: Advances in Theory, Algorithms and Applications* (Springer, 2014). His current research interests include blind signal processing, sparse signal processing, audio-visual signal processing, machine learning and perception, machine audition (listening), and statistical anomaly detection.

Dr. Wang is currently an Associate Editor for the IEEE TRANSACTIONS ON SIGNAL PROCESSING. He is a Member of the Ministry of Defence University Defence Research Collaboration in Signal Processing (since 2009), a Member of the BBC Audio Research Partnership (since 2011), and an Associate Member of the Surrey Centre for Cyber Security (since 2014). He is (or has been) a Chair, Session Chair, or Technical/Program Committee Member for a number of international conferences, including Publication Co-Chair of ICASSP 2019, Session Chair of ISP 2015, Local Arrangement Co-Chair of MLSP 2013, Session Chair of ICASSP 2012, Area and Session Chair of EUSIPCO 2012, and Track Chair and Publicity Co-Chair of SSP 2009. He was a Tutorial Speaker for ICASSP 2013 and UDRC Summer School 2014 and 2015.