Discriminative manifold extreme learning machine and applications to image and EEG signal classification

Yong Peng[a], Bao-Liang Lu[a,b,*]

[a] *Center for Brain-like Computing and Machine Intelligence, Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China*

[b] *Key Laboratory of Shanghai Education Commission for Intelligent Interaction and Cognitive Engineering, Shanghai Jiao Tong University, Shanghai 200240, China*

Abstract

Extreme learning machine (ELM) uses a non-iterative method to train single-hidden-layer feed-forward networks (SLFNs), which has been proven to be an efficient and effective learning model for both classification and regression. The main advantage of ELM lies in that the input weights as well as the hidden layer biases can be randomly generated, which contributes to the analytical solution of output weights. In this paper, we propose a discriminative manifold ELM (DMELM) by simultaneously considering the discriminative information and geometric structure of data; specifically, we exploit the discriminative information in the local neighborhood around each data point. To this end, a graph regularizer based on a newly designed graph Laplacian to characterize both properties is formulated and incorporated into the ELM objective. In DMELM, the output weights can also be obtained in analytical form. Extensive experiments are conducted on image and EEG signal classification to evaluate the effectiveness of DMELM. The results show that DMELM consistently achieves better performance than original ELM and yields promising results in comparison with several state-of-the-art algorithms, which suggests that the discriminative as well as manifold information are beneficial to classification.

Keywords: Extreme learning machine, Discriminative information, Manifold information, Image classification, EEG, Emotion recognition

1. Introduction

SLFNs have been extensively studied during the past several decades. The most popular algorithm used for training SLFNs is the back-propagation algorithm [1], which adopts the gradient descent methods to optimize the weights in neural networks. However, the gradient-based methods cannot guarantee the global optima and they are often time-consuming due to the iterative process in weight tuning.

As an alternate, ELM was proposed by Huang *et al.* [2, 3] as a new paradigm to train SLFNs in which only the output weights between the hidden layer and output layer need to be optimized. The main difference between ELM and existing approaches is that the input weights and biases of the hidden neurons in ELM can be randomly generated. The original ELM adopts the least square loss to measure the prediction error, which causes that the output weights can be solved analytically. Therefore, ELM can attain much faster learning speed than gradient-based methods. The universal approximation capacity is also maintained by ELM with fixed hidden neurons and tunable output weights [4, 5]. ELM provides us a unified model for binary classification, multiclass classification and regression [6], which can achieve comparable or even better prediction error than support vector machine (SVM) [6, 7]. ELM shares many similarities as well as several differences with SVM, which were reviewed in detail by [8, 9, 10].

With the advance of ELM research, much efforts have been made from both theoretical and applications perspectives. Inspired by the great success of deep learning models, Kasun *et al.* introduced a building block, ELM autoencoder (ELM-AE), to represent features based on singular values [11]. Several ELM-AEs can be stacked together to form a deep architecture, namely multilayer neural network. The elastic net regularized ELM was proposed by [12] and put into EEG-based drivers' vigilance estimation. Wang *et al.* proposed a parallelized ELM ensemble framework based on the min-max modular network [13], which has great capacity to process big and imbalanced data [14]. To emphasize the label consistency of training examples, Peng *et al.* presented the discriminative graph regularized ELM (GELM) [15], which enforces the ELM network outputs of training samples from the same class to be similar. Though most of existing ELM variants focused on supervised learning tasks, Huang and colleagues extend-

*Corresponding author

Email addresses: `stany.peng@gmail.com` (Yong Peng), `bllu@sjtu.edu.cn` (Bao-Liang Lu)

ed ELM into semi-supervised and unsupervised learning based on the manifold regularization [16], which greatly expands the applicability of ELM. Various improvements have been applied to the original ELM, rendering it more effective or suitable for specific applications such as ELM for sequential online learning [17, 7, 18, 19], security assessment in power systems [20], no-reference image quality assessment [21], remote sensing image classification [22], medical related applications [23, 24, 25], data privacy [26]. ELM has been implemented by parallel techniques [27, 28]. The hardware technique-based implementation [29] makes ELM efficiently deal with large data sets and real time reasoning. Detailed review on ELM can be found in [30, 31].

Though ELMs have become increasingly popular in diverse fields, the objective of ELMs in least square form mainly pays attention to the discriminative information of data. Recently, various researchers [32, 33, 34] have considered the case when the data is sampled from a probability distribution that has support on or near to a *submanifold* of the ambient space. Here, a d-dimensional submanifold of an Euclidean space $\mathbb{R}^m$ is a subset $\mathcal{M} \subset \mathbb{R}^m$ which locally looks like a flat d-dimensional Euclidean space [35]. In order to detect the underlying manifold structure, various *manifold learning* algorithms have been proposed such as locally linear embedding [32], ISOMAP [33], Laplacian eigenmap [34] and local tangent space alignment [36]. One of the key ideas in manifold learning is the so-called locally invariant idea [37], *i.e.*, the nearby points are likely to have similar transformed representations.

The earlier research on manifold learning mainly focused on nonlinear dimensionality reduction. In recent studies, manifold assumption or locally invariant idea was extensively applied to some popular learning models such as non-negative matrix factorization [38, 39, 40], concept factorization [41], sparse coding [42], low-rank representation [43], and Gaussian mixture model [44]. All these studies demonstrated that learning performance can be significantly enhanced if the geometric structure of data is exploited and the local invariance is considered.

In this paper, we propose to improve the performance of ELM by emphasizing both discriminative information and geometric structure of data. Accordingly, a discriminative manifold extreme learning machine is formulated, which can exploit the discriminative information in the neighborhood around each data point. Different from the existing several linear models which employed the maximum margin criterion [45] and local manifold information [40], the proposed DMELM has two different characteristics: (1) random feature mapping from input layer to hidden layer; and (2) the output weights can be more efficiently obtained by solving a regularized least square problem. As pointed by [16], generating feature mapping randomly enables ELM the capacity of nonlinear feature learning and alleviates the risk of overfitting.

The remainder of this paper is organized as follows. In Section 2, we briefly review the ordinary ELM and the discriminative graph regularized ELM. The model formulation as well as some discussions of the proposed DMELM are introduced in Section 3. Experiments to show the effectiveness of DMELM on image and EEG signal classification are presented in Section 4. Concluding remarks are given in Section 5.

2. Preliminaries

2.1. Extreme learning machine

ELM was originally proposed for training SLFNs and was then extended for training the generalized SLFNs where the hidden layer need not to be neuron alike. Considering the supervised learning task, we are provided N training samples $\{\mathbf{x}_i, \mathbf{t}_i\}_{i=1,\ldots,N}$ from C classes, where each sample and its corresponding network target vector are respectively as $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{iD})^T$ and $\mathbf{t}_i = (t_{i1}, t_{i2}, \ldots, t_{iC})$. In ELM, the network input weights $\mathbf{W} \in \mathbb{R}^{L\times D}$ and the hidden layer biases $\mathbf{b} \in \mathbb{R}^L$ are randomly generated. Assuming that the number of hidden neurons is L, the output function of ELM for SLFNs is

$$f_L(\mathbf{x}) = \sum_{i=1}^{L} \beta_i h_i(\mathbf{x}) = \mathbf{h}(\mathbf{x})\boldsymbol{\beta}, \tag{1}$$

where $\boldsymbol{\beta} = [\beta_1, \beta_2, \ldots, \beta_L]^T \in \mathbb{R}^{L\times C}$ is the output weights between the hidden layer and the output layer, $\mathbf{h}(\mathbf{x}) = [h_1(\mathbf{x}), \ldots, h_L(\mathbf{x})]$ is the output row vector of the hidden layer w.r.t. the input $\mathbf{x}$. $\mathbf{h}(\mathbf{x})$ actually maps the data from the D-dimensional input space to the L-dimensional hidden layer feature space, that is, ELM feature space $\mathcal{H}$. Therefore, $\mathbf{h}(\mathbf{x})$ is indeed a feature mapping.

The ordinary ELM aims to minimize the objective

$$\min_{\boldsymbol{\beta}} \|\mathbf{H}\boldsymbol{\beta} - \mathbf{T}\|^2, \tag{2}$$

where $\mathbf{H}$ is the hidden layer output matrix as

$$\mathbf{H} = \begin{bmatrix} \mathbf{h}(\mathbf{x}_1) \\ \mathbf{h}(\mathbf{x}_2) \\ \vdots \\ \mathbf{h}(\mathbf{x}_N) \end{bmatrix} = \begin{bmatrix} h_1(\mathbf{x}_1) & h_2(\mathbf{x}_1) & \ldots & h_L(\mathbf{x}_1) \\ h_1(\mathbf{x}_2) & h_2(\mathbf{x}_2) & \ldots & h_L(\mathbf{x}_2) \\ \vdots & \vdots & \vdots & \vdots \\ h_1(\mathbf{x}_N) & h_2(\mathbf{x}_N) & \ldots & h_L(\mathbf{x}_N) \end{bmatrix}.$$

Therefore, the output weight matrix $\boldsymbol{\beta}$ can be estimated analytically by

$$\hat{\boldsymbol{\beta}} = \arg\min_{\beta} ||\mathbf{H}\boldsymbol{\beta} - \mathbf{T}||_2^2 = \mathbf{H}^\dagger \mathbf{T}, \tag{3}$$

where $\mathbf{H}^\dagger$ is the Moore-Penrose generalized inverse of $\mathbf{H}$. If $\mathbf{H}^T\mathbf{H}$ is nonsingular, $\mathbf{H}^\dagger = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T$; or when $\mathbf{H}\mathbf{H}^T$ is nonsingular, $\mathbf{H}^\dagger = \mathbf{H}^T(\mathbf{H}\mathbf{H}^T)^{-1}$ [6].

In order to improve the stability and generalization performance of the ordinary ELM, a small positive value can be added to the diagonal of $\mathbf{H}^T\mathbf{H}$ or $\mathbf{H}\mathbf{H}^T$. In this method, the solution of regularized ELM can be expressed as

$$\hat{\boldsymbol{\beta}} = \left(\mathbf{H}^T\mathbf{H} + \frac{\mathbf{I}}{\lambda}\right)^{-1} \mathbf{H}^T\mathbf{T}. \tag{4}$$

The solution shown in (4) can be obtained by solving the following optimization problem

$$\min_{\boldsymbol{\beta}} \mathcal{J}_{RLEM} = \frac{1}{\lambda}\|\boldsymbol{\beta}\|^2 + \sum_{i=1}^{N}\|\xi_i\|_2^2, \quad s.t., \xi_i = \mathbf{t}_i - \mathbf{h}(\mathbf{x}_i)\boldsymbol{\beta}, \, i = 1, \ldots, N \tag{5}$$

where $||\boldsymbol{\beta}||_2^2 = \sum_{j=1}^{L} ||\beta_j||_2^2$ is regarded as the regularization term and $||\beta_j||_2^2$ denotes the ℓ_2-norm of vector β_j. Moreover, λ denotes the regularization parameter to balance the influence of error term and the model complexity. It is a general method to make the least square regression stable, which is called "ridge regression" [46] in statistics.

As a whole, training a SLFN based on ELM rule can be summarized in Algorithm 1.

Algorithm 1 Extreme learning machine

Input: training set $\mathcal{X} = \{\mathbf{x}_i, \mathbf{t}_i\}_{i=1,\ldots,N}$, activation function $g(\cdot)$, number of hidden neurons L and regularization parameter λ;

Output: Output weight matrix $\boldsymbol{\beta}$;

1: Randomly assign input weights $\mathbf{W}$ and hidden biases $\mathbf{b}$;
2: Calculate the hidden layer output matrix $\mathbf{H}$;
3: Calculate the output weight matrix $\hat{\boldsymbol{\beta}}$ by (3) or (4).

2.2. Discriminative graph regularized ELM

As the label consistency property of training samples is not considered in ELM, GELM [15] was proposed to enforce the output of training samples from the same class to be similar. In GELM, label information of training samples was used to construct an adjacent graph and the graph regularizer was formulated to constrain the output. This constraint is imposed on the ELM objective. In GELM, the output weights can be solved analytically.

In GELM, supposing that we have a training set with N samples from C classes in which the c-th class has N_c samples, then the adjacent matrix $\mathbf{W}$ would be defined as

$$W_{ij} = \begin{cases} \frac{1}{N_c}, & \text{if both } \mathbf{h}(\mathbf{x}_i) \text{ and } \mathbf{h}(\mathbf{x}_j) \text{ belong to the } c\text{-th class,} \\ 0, & \text{otherwise,} \end{cases}$$

where $\mathbf{h}(\mathbf{x}_i) = [h_1(\mathbf{x}_i), \ldots, h_L(\mathbf{x}_i)] \triangleq \mathbf{h}_i$ and $\mathbf{h}(\mathbf{x}_j) = [h_1(\mathbf{x}_j), \ldots, h_L(\mathbf{x}_j)] \triangleq \mathbf{h}_j$ are hidden layer representations w.r.t. two input samples $\mathbf{x}_i$ and $\mathbf{x}_j$, respectively. If we define a diagonal matrix $\mathbf{D}$ with column sums of $\mathbf{W}$ as its entries, the graph Laplacian can be calculated by $\mathbf{L}_{\mathrm{GELM}} = \mathbf{D} - \mathbf{W}$. Denote the outputs w.r.t. $\mathbf{h}_i$ and $\mathbf{h}_j$ respectively by $\mathbf{y}_i$ and $\mathbf{y}_j$. On the basis of label consistency that when $\mathbf{h}_i$ and $\mathbf{h}_j$ are from the same class, $\mathbf{y}_i$ and $\mathbf{y}_j$ should share similar properties, we minimize the following objective

$$\sum_{i=1}^{N}\sum_{j=1}^{N}\|\mathbf{y}_i - \mathbf{y}_j\|^2 W_{ij} = \mathrm{Tr}(\mathbf{Y}^T\mathbf{L}_{\mathrm{GELM}}\mathbf{Y}), \tag{6}$$

where $\mathbf{Y} = \mathbf{H}\boldsymbol{\beta}$ is the output of ELM. Therefore, the objective function of GELM is defined as follows

$$\min_{\boldsymbol{\beta}} \; \|\mathbf{H}\boldsymbol{\beta} - \mathbf{T}\|_2^2 + \lambda_1 \mathbf{Tr}\left((\mathbf{H}\boldsymbol{\beta})^T\mathbf{L}_{\mathrm{GELM}}(\mathbf{H}\boldsymbol{\beta})\right) + \frac{1}{\lambda_2}\|\boldsymbol{\beta}\|_2^2, \tag{7}$$

where $\mathrm{Tr}\left((\mathbf{H}\boldsymbol{\beta})^T\mathbf{L}_{\mathrm{GELM}}(\mathbf{H}\boldsymbol{\beta})\right)$ is the graph regularizer.

3. Discriminative manifold ELM

3.1. DMELM model formulation

The graph regularizer in GELM tried to preserve the label consistency of training samples. Roughly, GELM assumes the samples from each class as one manifold, which considers the manifold structure of data on the class level. However, in real world applications, taking face recognition as an example, face images with similar variations, such as illumination or expression, often have higher correlation than those from the same subject. This means that mining the discriminative information in a local area is beneficial for classification. Therefore, in this section we will present a new regularizer into ELM to let its output layer (1) preserve the geometric structure of data by applying manifold regularization and (2) maximize the margins between different classes to incorporate the discriminative information. Specifically, both properties can be attained by exploiting the discriminative information in the local neighborhood around each data point.

Before introducing the regularizer, we first review the general manifold regularization method [47]. Generally, manifold regularization exploits the geometry of the marginal distribution $\mathcal{P}_X$, which ensures that the solution is smooth w.r.t. both ambient space and the marginal distribution $\mathcal{P}_X$, resulting in the following objective

$$\min_{f \in \mathcal{H}_K} \frac{1}{N}\sum_{i=1}^{N}\ell(\mathbf{x}_i, y_i, f(\mathbf{x}_i)) + \gamma_A\|f\|_K^2 + \gamma_I\|f\|_I^2, \tag{8}$$

where the regularizer $\|f\|_K^2$ controls the model complexity, $\|f\|_I^2$ is the manifold regularizer to control the complexity measured by the manifold geometry of the sample distribution, and ℓ is the loss function. In ELM, the specific form of objective (8) becomes

$$\min_{\boldsymbol{\beta}} \|\mathbf{H}\boldsymbol{\beta} - \mathbf{T}\|_2^2 + \lambda_1\mathcal{R}_{dm} + \frac{1}{\lambda_2}\|\boldsymbol{\beta}\|_2^2. \tag{9}$$

The $\mathcal{R}_{dm}$ in (9) is expected to reflect the local discriminative structure of data. Then, in the output layer of discriminative manifold ELM, the learned representation can well preserve the neighboring relationship of samples from the same class while separate the nearby samples from different classes far from each other. As a result, DMELM can further maximize the margins among samples from different classes in local neighborhood around each data point.

Based on the spectral graph theory [48] and the general graph embedding framework [49], the geometric structure of data can be characterized by a graph $G(V, E, \mathbf{W})$, where V is a set of vertices in which each vertex represents a data point, $E \subseteq V \times V$ is a set of edges connecting related vertices and $\mathbf{W}$ is an adjacency matrix recording the pairwise weights between vertices. To depict local geometric structure, G is usually a sparse graph which means that $\mathbf{W}$ only gives the nearest neighbors information of each data point. In our discriminative manifold formulation of ELM, two graphs, within-class graph G_w and between-class graph G_b, are constructed in the ELM input layer because the discriminative as well as manifold information of data are fully given in the original data space.

Concretely, for each data point $\mathbf{x}_i$, we first divide its k nearest neighbors into two non-overlapping subsets according to their labels. Then, we can construct graphs G_w and G_b for $\mathbf{x}_i$ as

$$W_{w,ij} = \begin{cases} 1, & \text{if } \mathbf{x}_j \in \mathcal{N}_k(\mathbf{x}_i) \text{ or } \mathbf{x}_i \in \mathcal{N}_k(\mathbf{x}_j) \\ & \quad \mathbf{x}_i \text{ and } \mathbf{x}_j \text{ are from the same class,} \\ 0, & \text{otherwise.} \end{cases}$$

$$W_{b,ij} = \begin{cases} 1, & \text{if } \mathbf{x}_j \in \mathcal{N}_k(\mathbf{x}_i) \text{ or } \mathbf{x}_i \in \mathcal{N}_k(\mathbf{x}_j) \\ & \quad \mathbf{x}_i \text{ and } \mathbf{x}_j \text{ are from different classes,} \\ 0, & \text{otherwise.} \end{cases}$$

where $\mathcal{N}_k(\mathbf{x}_i)$ denotes the set of k nearest neighbors of $\mathbf{x}_i$. Obviously, in DMELM output layer, we need to (1) enforce the output representations of neighboring samples on G_w to stay as close as possible and (2) enforce the output representations of connected samples on G_b stay as far as possible. Denote these two objectives respectively by $\mathcal{O}_1$ and $\mathcal{O}_2$ and we can simply define them as

$$\mathcal{O}_1 = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} W_{w,ij}\|\mathbf{h}_i\boldsymbol{\beta} - \mathbf{h}_j\boldsymbol{\beta}\|_2^2, \tag{10}$$

and

$$\mathcal{O}_2 = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} W_{b,ij}\|\mathbf{h}_i\boldsymbol{\beta} - \mathbf{h}_j\boldsymbol{\beta}\|_2^2, \tag{11}$$

where $\mathbf{h}_i$ and $\mathbf{h}_j \in \mathbb{R}^{1\times L}$ are two rows in $\mathbf{H}$, corresponding to the two hidden representations of samples $\mathbf{x}_i$ and $\mathbf{x}_j$.

The compact forms of $\mathcal{O}_1$ and $\mathcal{O}_2$ can be reached by respectively imposing linear transformations on (10) and (11). Therefore, we have

$$\begin{aligned}
\mathcal{O}_1 &= \frac{1}{2}\sum_{i,j=1}^{N} W_{w,ij}\|\mathbf{h}_i\boldsymbol{\beta} - \mathbf{h}_j\boldsymbol{\beta}\|_2^2 \\
&= \frac{1}{2}\sum_{i,j=1}^{N} W_{w,ij}\,\mathrm{Tr}\left((\mathbf{h}_i\boldsymbol{\beta} - \mathbf{h}_j\boldsymbol{\beta})^T(\mathbf{h}_i\boldsymbol{\beta} - \mathbf{h}_j\boldsymbol{\beta})\right) \\
&= \mathrm{Tr}\left(\sum_{i=1}^{N}(\mathbf{h}_i\boldsymbol{\beta})^T\left[\sum_j W_{w,ij}\right]\mathbf{h}_i\boldsymbol{\beta} - \sum_{i,j=1}^{N}(\mathbf{h}_i\boldsymbol{\beta})^T W_{w,ij}\mathbf{h}_j\boldsymbol{\beta}\right) \\
&= \mathrm{Tr}\left((\mathbf{H}\boldsymbol{\beta})^T(\mathbf{D}_w - \mathbf{W}_w)(\mathbf{H}\boldsymbol{\beta})\right) \\
&= \mathrm{Tr}\left((\mathbf{H}\boldsymbol{\beta})^T\mathbf{L}_w(\mathbf{H}\boldsymbol{\beta})\right),
\end{aligned}$$

where $\mathbf{D}_w$ is a diagonal degree matrix with entries $D_{w,ii} = \sum_j W_{w,ij}$ or $D_{w,ii} = \sum_i W_{w,ij}$ since $\mathbf{W}_w$ is symmetric, $\mathbf{L}_w = \mathbf{D}_w - \mathbf{W}_w$ is the Laplacian matrix of graph G_w. Similarly, we have

$$\mathcal{O}_2 = \mathrm{Tr}\left((\mathbf{H}\boldsymbol{\beta})^T\mathbf{L}_b(\mathbf{H}\boldsymbol{\beta})\right),$$

where $\mathbf{L}_b = \mathbf{D}_b - \mathbf{W}_b$ is the Laplacian matrix of graph G_b. Similar to $\mathbf{D}_w$, $\mathbf{D}_b$ is also a degree matrix which has each diagonal entry defined as $D_{b,ii} = \sum_j W_{b,ij}$ or $D_{b,ii} = \sum_i W_{b,ij}$ since $\mathbf{W}_b$ is symmetric.

Define $\mathbf{F} \triangleq \mathbf{H}\boldsymbol{\beta}$, simultaneously minimizing $\mathcal{O}_1$ and maximizing $\mathcal{O}_2$ lead to the following problem

$$\min_{\mathbf{F}} \frac{\mathrm{Tr}(\mathbf{F}^T\mathbf{L}_w\mathbf{F})}{\mathrm{Tr}(\mathbf{F}^T\mathbf{L}_b\mathbf{F})}. \tag{12}$$

Based on the connection between Rayleigh quotient and eigen-value decomposition, the above objective can be optimized by solving the following eigenvalue decomposition problem

$$\mathbf{L}_w\mathbf{v} = \eta\mathbf{L}_b\mathbf{v}, \tag{13}$$

which is equivalent to

$$\mathbf{L}_w\mathbf{L}_b^{-\frac{1}{2}}\mathbf{u} = \eta\mathbf{u} \tag{14}$$

by setting $\mathbf{u} = \mathbf{L}_b^{-\frac{1}{2}}\mathbf{v}$. Therefore, we have the transformed form as

$$\mathbf{L}_b^{-\frac{1}{2}}\mathbf{L}_w\mathbf{L}_b^{-\frac{1}{2}}\mathbf{u} = \eta\mathbf{u}, \tag{15}$$

which is corresponding to the objective as

$$\min_{\mathbf{F}} \mathrm{Tr}\left(\mathbf{F}^T\mathbf{L}_b^{-\frac{1}{2}}\mathbf{L}_w\mathbf{L}_b^{-\frac{1}{2}}\mathbf{F}\right). \tag{16}$$

Accordingly, the $\mathcal{R}_{dm}$ in (9) has the following expression

$$\mathcal{R}_{dm} = \mathrm{Tr}\left((\mathbf{H}\boldsymbol{\beta})^T(\mathbf{L}_b^{-\frac{1}{2}})^T\mathbf{L}_w(\mathbf{L}_b^{-\frac{1}{2}})(\mathbf{H}\boldsymbol{\beta})\right). \tag{17}$$

We add a tiny perturbation to the diagonal of the graph Laplacian matrix $\mathbf{L}_b$, *i.e.*, $\tilde{\mathbf{L}}_b = \mathbf{L}_b + \zeta\mathbf{I}$, to make it always invertible. In all experiments, we empirically set ζ as a fixed small value $10^{-6}\mathrm{Tr}(\mathbf{L}_b)$. In the rest of this paper, we still use the notation $\mathbf{L}_b$ other than the perturbed matrix $\tilde{\mathbf{L}}_b$ for simplicity.

We define a unified graph Laplacian matrix as $\mathbf{L}_{\mathrm{DMELM}} \triangleq (\mathbf{L}_b^{-\frac{1}{2}})^T\mathbf{L}_w(\mathbf{L}_b^{-\frac{1}{2}})$ for graphs G_w and G_b instead of individually using two matrices $\mathbf{L}_w$ and $\mathbf{L}_b$ following the lines in [40]. As a result, we can formulate the objective of DMELM as

$$\min_{\boldsymbol{\beta}} \|\mathbf{H}\boldsymbol{\beta} - \mathbf{T}\|_2^2 + \lambda_1\mathrm{Tr}\left((\mathbf{H}\boldsymbol{\beta})^T\mathbf{L}_{\mathrm{DMELM}}(\mathbf{H}\boldsymbol{\beta})\right) + \frac{1}{\lambda_2}\|\boldsymbol{\beta}\|_2^2. \tag{18}$$

We can easily find that the objective of DMELM shares the same form as that of GELM [15]. However, the difference between them is obvious; the Laplacian matrix $\mathbf{L}_{\mathrm{DMELM}}$ characterizes manifold as well as discriminative information of data, which contains more information than

$\mathbf{L}_{\text{GELM}}$ in GELM. Objective (18) is a quadratic form w.r.t. $\boldsymbol{\beta}$. By setting its derivative w.r.t. $\boldsymbol{\beta}$ to be zero, we can obtain the estimated output weight matrix of DMELM as

$$\hat{\boldsymbol{\beta}} = (\mathbf{H}^T\mathbf{H} + \lambda_1\mathbf{H}^T\mathbf{L}_{\text{DMELM}}\mathbf{H} + \frac{1}{\lambda_2}\mathbf{I})^{-1}\mathbf{H}^T\mathbf{T}. \quad (19)$$

3.2. Discussion

We give some discussions on the connection between D-MELM and related works.

Yan and colleagues [49] proposed a general framework for dimensionality reduction based on graph embedding in which the statistical or geometric properties of a data set was characterized by constructing different graphs. This work is closely related to DMELM in constructing the two different types of graphs G_w and G_b. However, there are several differences between DMELM and Yan's work. Firstly, Yan's work directly operates sample in the raw feature space; in DMELM, we use the representation in ELM feature space, whose rationality has been extensively studied in [50, 51, 52]. Secondly, Yan's work mainly works on dimensionality reduction which can be seen as feature transformation. In DMELM, we aim to let its output layer (1) preserve the geometric structure of data by applying manifold regularization, and (2) maximize the margins between different classes to incorporate the discriminative information.

The motivation of the GELM [15] model is actually to preserve the local consistency of data; however, such geometric property is hard to explore after the nonlinear mapping of ELM hidden layer. Therefore, GELM tried to preserve the label consistency of training samples. Generally, GELM assumes the samples from each class as one manifold, which considers the manifold structure of data on class level. In DMELM, we try to exploit the discriminative information in local neighborhood around each data point, which explicitly considers the local manifold structure and discriminative information of data. We can view DMELM as a refinement of GELM by emphasizing the local geometric property.

4. Experimental studies

In this section, we evaluate the performance of DMELM on two types of classification tasks, image classification and EEG-based emotion recognition. In both experiments, the activation function of the hidden layer is the 'sigmoid' function. To help reproducing the experimental results described in this work, the source code will be available from `http://bcmi.sjtu.edu.cn/~pengyong`.

4.1. Image classification

Four representative data sets, ORL, PIE, COIL20 and USPS, are used in image classification. The properties of these four data sets are briefly described below (see also Table 1).

4.1.1. Data sets

- **ORL**[1]. There are 40 subjects and each subject has 10 different face images in ORL database. For some subjects, the images were taken at different times, varying the lighting, facial expressions (open/closed eyes, smiling/not smiling) and facial details (glasses/no glasses). All the images were taken against a dark homogeneous background with the subjects in an upright, frontal position (with tolerance for some side movement). Each image was normalized to 32×32 pixel array and reshaped to a long vector.

- **PIE**[2]. It contains 41,368 face images of 68 subjects, each subject under 13 different poses, 43 different illumination conditions and with 4 different expressions. We choose the five near frontal poses (C05, C07, C09, C27, C29) and use all 11,544 images under different illuminations and expressions where each person has about 170 images except for a few bad images.

- **COIL20**[3]. It is a dataset of gray-scale images of 20 objects. The objects were placed on a motorized turntable against a background. The turntable was rotated through 360° to vary the object poses with respect to a fixed camera. Images of the objects were taken at pose intervals of 5°, which corresponds to 72 images per object. For experiments, we have resized each of the original 1440 images down to 32×32 pixels.

- **USPS**. It consists of gray-scale handwritten digit images. We use a popular subset which contains 9298 handwritten digit images in total provided by Deng Cai[4]. The size of each image is 16×16 pixels with 256 gray levels.

Figure 1 shows some sample images from above data sets.

Table 1: Statistics of the four data sets.

Dataset	Size(N)	Dimensionality(D)	#Class (C)
ORL	400	1024	40
PIE	11544	1024	68
COIL20	1440	1024	20
USPS	9298	256	10

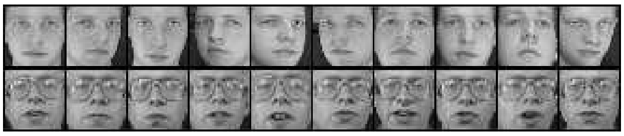
In this experiment, we compare DMELM with ordinary ELM, the ℓ_2-norm regularized ELM (RELM) and discriminative graph regularized ELM. Each image data set is partitioned into the different gallery and probe sets, and for these data sets we randomly select $l_{\text{ORL}} = \{2, 3, 4, 5\}$, $l_{\text{PIE}} = \{5, 10, 15, 20\}$, $l_{\text{COIL20}} = \{2, 4, 6, 8\}$ and $l_{\text{USPS}} = \{3, 5, 10, 15\}$ samples per class for training and the rest for

[1]http://www.uk.research.att.com/facedatabase.html

[2]http://www.ri.cmu.edu/projects/project_418.html

[3]http://www.cs.columbia.edu/CAVE/software/softlib/coil-20.php

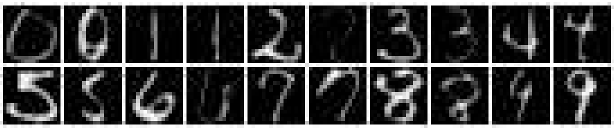
[4]http://www.cad.zju.edu.cn/home/dengcai/Data/MLData.html



(a) Sample images of 2 subjects in ORL.

(b) Sample images of 2 subjects in PIE.

(c) Sample images of 20 objects in COIL20.



(d) Sample images of 10 digits in USPS.

Figure 1: Sample images of ORL, PIE, COIL20 and USPS.

testing. Though the training and testing sets are randomly generated, they are kept the same for all comparing algorithms for fair comparison. Before classification, samples are projected to $N_{tr} - 1$ (N_{tr} is the number of training examples) dimensional PCA subspace for all ELMs. The setting of specific parameters in DMELM will be described in section 4.1.3.

4.1.2. Experimental results

Tables 2, 3, 4 and 5 show the experimental results of different ELMs on these four data sets, respectively. It can be found that DMELM consistently achieves the best performance over all the data sets.

From the results, we can see that all ELMs can be effectively trained when given more training samples and thus the accuracy gap between them is minor. However, when given a small amount of training samples, DMELM can obtain better generalization performance than the other ELMs. For example, in the ORL classification experiment, DMELM and ELM has significant difference in accuracy (10%), which is caused by that DMELM explores more side information from the data set such as the discriminative and manifold structure than ELM.

Table 2: Results (%) of ELM variants on ORL.

ORL	2 Train	3 Train	4 Train	5 Train
ELM	79.69	84.64	89.17	94.50
RELM	83.44	87.86	95.83	96.50
GELM	87.19	90.71	96.25	96.50
DMELM	**89.38**	**91.79**	**97.50**	**97.50**

Table 3: Results (%) of ELM variants on PIE.

PIE	5 Train	10 Train	15 Train	20 Train
ELM	69.27	78.93	83.90	87.49
RELM	73.85	86.32	90.72	92.82
GELM	78.10	88.47	92.11	93.83
DMELM	**79.19**	**88.90**	**92.41**	**94.01**

Table 4: Results (%) of ELM variants on COIL20.

COIL20	2 Train	4 Train	6 Train	8 Train
ELM	71.50	84.34	87.05	89.77
RELM	72.43	84.71	87.12	89.84
GELM	73.64	85.29	87.65	91.33
DMELM	**75.29**	**87.35**	**89.77**	**92.89**

Table 5: Results (%) of ELM variants on USPS.

USPS	3 Train	5 Train	10 Train	15 Train
ELM	71.47	81.08	84.22	88.15
RELM	72.31	82.29	84.95	88.61
GELM	72.32	82.58	85.01	88.87
DMELM	**73.94**	**84.22**	**86.82**	**89.60**

These experimental results reveal a number of interesting points:

(1) The stability of learning algorithm is important. The ordinary ELM may encounter the singularity problem which can be avoided by introducing the ℓ_2-norm regularization. The ℓ_2-norm constraint can shrink values of output weight matrix, which yields better generalization performance. Thus, the performance of RELM is better than that of ELM;

(2) The label consistency is important besides the training error. Actually, the graph regularization in GELM depicts the manifold information on class level. By enforcing the label consistency property that samples from the same class have similar outputs, GELM obtains obvious accuracy improvement w.r.t. ELM and RELM;

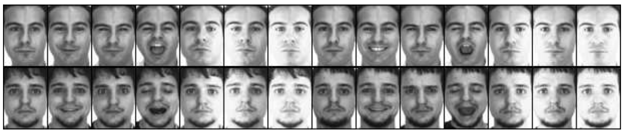
(3) Both discriminative information and manifold structure in data are important for classification. Our experimental results demonstrated that the unified graph Laplacian defined in DMELM which simultaneously considers the discriminative and manifold information is much more effective than that in GELM. The learned output weights can obtain the strong discriminative ability and vary smoothly along the data manifold to some extent.

Further, we show the effectiveness of DMELM by comparing them with some state-of-the-art classification methods by following the pipeline in [15]. For fair comparison, the experimental paradigm is the same as that in [53] and the data sets are Extended Yale B and AR face data sets. These classification method widely used in face recognition are nearest neighbor classifier (NN), linear regression classifier (LRC), support vector machine (SVM), sparse representation-based classification (SRC) [54], collaborative representation-based regularized least square (CRC_RLS) [53]. The characteristics of these two data sets are stated as follows:

- **Extended Yale B**[5]. The Extended Yale B contains contains 2414 frontal face images of 38 subjects. We used the cropped and normalized face images of size 54×48, which were taken under varying illumination conditions. We randomly split the data set into two halves. One half, which contains 32 images for each subject, was used as training set, and the other half was used for testing.
- **AR**[6]. It contains 100 subjects and each subject has 26 face images taken in two sessions. For each session, there are 13 face images. In our experiment, a subset (with only illumination and expression changes) was chosen. For each subject, 7 images from session 1 were used for training, with the other 7 images from session 2 for testing. The images were cropped to 60×43.

Some sample images from Extended Yale B and AR data sets are shown in Figure 2.

(a) Sample images of 2 subjects in Extended Yale B.



(b) Sample images of 2 subjects in AR.

Figure 2: Sample images from Extended Yale B and AR.

Table 6 demonstrates the results versus feature dimension by NN, LRC, SVM, SRC, CRC_RLS and DMELM on the Extended Yale B and AR data sets, respectively. It can be seen that regardless of different dimension settings, DMELM always results in the best performance over these state-of-the-art classification methods. Even the accuracy is nearly saturated, DMELM still can obtain the superiority to GELM. Especially for result when dimension is 54 on AR, DMELM gets approximately 3% improvement. This shows that by leveraging the power of exploiting the two properties, the learned ELM output mapping can yield better generalization performance.

4.1.3. Parameter sensitivity analysis

There are five parameters in the proposed DMELM model: the number of hidden neurons L, the parameters λ_1 for discriminative manifold regularizer, λ_2 for ℓ_2-norm regularizer, parameters $k1$ and $k2$ for the sizes of within-class and between-class graphs. In this section, we analyze the sensitivity of DMELM w.r.t. these parameters.

Based on the results in [6], the performance of ELM is not very sensitive to the number of hidden neurons, which is still an open problem in ELM research. We also conduct experiments on the four data sets used in section 4.1.1 and Figure 3 shows the sensitivity of ELM versus different number of hidden neurons. We can easily find that the performance of ELM is very stable w.r.t. different number of hidden neurons (only slight fluctuation when the size of training set is pretty small). Therefore, similar to [15], we simply set the number of hidden neurons a near optimal value as $5 \times numDim$ for ORL, PIE, COIL20, Extended Yale B and AR and $10 \times numDim$ for USPS. Therefore, if the dimension of input data is 10, the number of hidden neurons will be 50.

For the remaining four parameters, we divide them into two groups based on their different properties in DMELM: λ_1 and λ_2 are in group 1, $k1$ and $k2$ are in group 2. We evaluate the sensitivity of DMELM w.r.t. these two groups on PIE data set. We vary λ_1 and γ_2 in candidates $\{2^{-10}, \ldots, 2^{10}\}$, $k1$ in candidates $\{1, 2, \ldots, l_{\text{PIE}} - 1\}$ and $k2$ in $\{5, 15, \ldots, 95\}$.

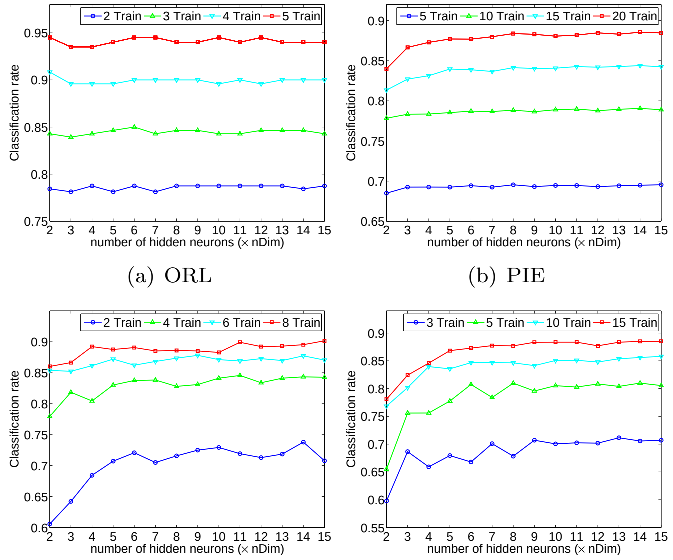
Figure 4 shows the sensitivity of DMELM w.r.t. different combinations of λ_1 and λ_2 with different number of training samples per subject. As we can see, for each setting of training and testing data, there is a large flat area near the optimal value on the landscape, which means DMELM is insensitive to the combination of parameters λ_1 and λ_2. For example, DMELM consistently achieves good performance for $\lambda_1 = \{2^4, 2^5, \ldots, 2^{10}\}$ and $\lambda_2 = \{2^3, 2^4, \ldots, 2^{10}\}$ when $l_{\text{PIE}} = 20$ and we can select parameter combination (λ_1, λ_2) from these candidate values. Generally, large λ_1 values are encouraged to emphasize the local discriminative information in data.

Table 6: The classification results (%) of different classification methods on Extended Yale B and AR.

Extended Yale B	#dim=84	#dim=150	#dim=300
NN	85.8	90.0	91.6
LRC	94.5	95.1	95.9
SVM	94.9	96.4	97.0
SRC	95.5	96.8	97.9
CRC_RLS	95.0	96.3	97.9
GELM	95.6	97.8	98.8
DMELM	**96.0**	**98.1**	**99.2**
AR	#dim=54	#dim=120	#dim=300
NN	68.0	70.1	71.3
LRC	71.0	75.4	76.0
SVM	69.4	74.5	75.4
SRC	83.3	89.5	93.3
CRC_RLS	80.5	90.0	93.7
GELM	83.0	90.3	93.6
DMELM	**85.7**	**91.3**	**94.1**

* The accuracies of the first five methods are from [53].

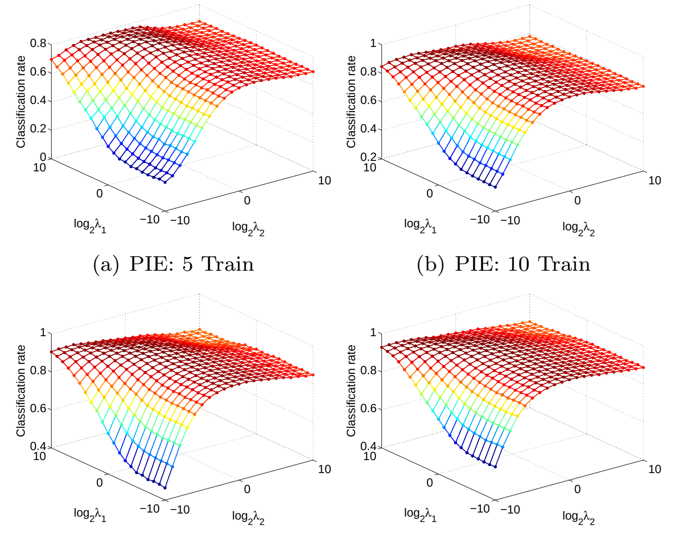
5 http://vision.ucsd.edu/∼leekc/ExtYaleDatabase/ExtYaleB.html
6 http://www2.ece.ohio-state.edu/ aleix/ARdatabase.html



(a) ORL (b) PIE

(c) COIL20 (d) USPS

Figure 3: Performance of ELM to different number of hidden neurons.



(a) PIE: 5 Train (b) PIE: 10 Train

(c) PIE: 15 Train (d) PIE: 20 Train

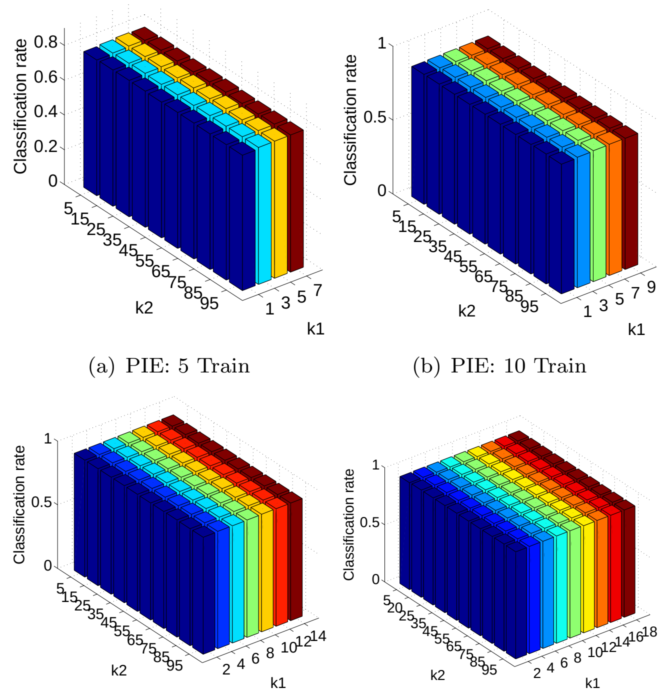
Figure 4: Performance of DMELM to different combinations of (λ_1, λ_2) on PIE.

Figure 5 shows the sensitivity of DMELM w.r.t. different combinations of $k1$ and $k2$ with different number of training samples per subject. It is obvious that the performance of DMELM is very stable w.r.t. different combinations of ($k1$, $k2$).

Thus, we fixed (λ_1, λ_2) as (10^0, 10^4), $k1 = \min(l, 3)$ and $k2$=20 for all the image data sets in previous experiments.

4.2. EEG-based emotion recognition

EEG signal, which record the electrical activities along the scalp, can provide researchers a reliable channel to investigate human emotional states. In this experiment, the proposed DMELM will be evaluated on EEG-based emotion recognition was compared with linear kernelized SVM, ELM, RELM and GELM.



(a) PIE: 5 Train (b) PIE: 10 Train

(c) PIE: 15 Train (d) PIE: 20 Train

Figure 5: Performance of DMELM to different combinations of ($k1$, $k2$) on PIE.

4.2.1. Data sets

The EEG data consists of three types of emotional states (positive, neutral and negative), which were previously evoked by watching corresponding types of movie clips. The stimuli are popular movies in Chinese, which are *Just Another Pandora's Box*, *Lost in Thailand*, *World Heritage in China*, *After Shock* and *Back to 1942*. Posters of these movies are shown in Figure 6.

Figure 6: Movie clips to evoke different types of emotional states. (from [55])

Three men and three women aged between 20 and 27 were involved in the EEG collection experiment. Each subject had three sessions experiment, with about one week interval. There are 15 movie clips in each session and 5 clips for each state. Each movie clip lasts about 4 minutes to show an vivid and relatively complete story.

A 62-channel electrode cap according to the extended international 10-20 system and ESI NeuroScan system were used to record the EEG data with sampling rate 1000Hz. Movie clips were played with a 10s rest and 15s hint between consecutive clips. During the rest, subjects were asked to fill a form as feedback to show whether the emotional states were successfully evoked. Figure 7 is the experimental procedure.

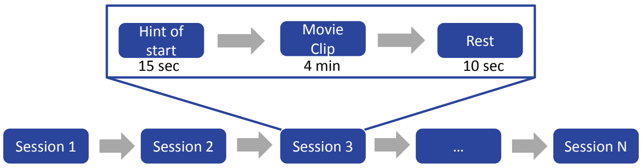


Figure 7: Procedure of stimuli playing.

The differential entropy (DE) [56], which is defined as

$$
\begin{aligned}
h(X) &= -\int_X f(x)\log(f(x))dx \\
&= \int_{-\infty}^{+\infty} \frac{-1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \log\left(\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}\right) dx \\
&= \frac{1}{2}\log(2\pi e\sigma^2),
\end{aligned}
$$

was extracted on the five frequency bands of EEG. They are δ(1-3Hz), θ(4-7Hz), α(8-13Hz), β(14-30Hz) and γ(31-50Hz). Short-time Fourier transform with 1s non-overlapping Hanning window was used to calculate the average DE features of each channel on these bands. Each band has 62 channels and thus 310 dimensional features were obtained for each sample. Since the effective experimental time lasted for 57 minutes, we finally got 3400 samples for each session. Linear dynamic system was used to remove the rapid changes of EEG features and get more reliable samples [57]. We chose 2000 samples as training set and the remainder in the same session as test set.

4.2.2. Experimental results

According to our previous research [55, 58], β and γ band features are more relevant to the emotion than the others. Therefore, we only report the results of different algorithms on β, γ and all frequency bands features to avoid a too large table. The number of hidden neurons in ELMs is set as three times of input dimension. The combination of (k_1, k_2) in DMELM is (20,20). The other involved parameters (C in SVM, λ in RELM, (λ_1, λ_2) in GELM and DMELM) are searched from $\{2^{-10}, 2^{-9}, \ldots, 2^{10}\}$ and then the best results are reported. Table 7 shows the EEG-based emotion recognition results of different algorithms on the six subjects. The best results across different algorithms with each frequency band feature are shown in boldface. Obviously, DMELM consistently performs better the other algorithms in most cases. The average results of different algorithms are presented in Table 8. When using all frequency band features, the average accuracy across all subjects of DMELM (81.01%) gets nearly 1% improvement w.r.t. GELM (80.25%), which suggests the effectiveness of exploiting local discriminative information. As an effective and efficient algorithm, RELM (78.10%) obtains 1.5% improvement w.r.t. SVM (76.62%) but with much less time cost. The performance of ordinary ELM is inferior to that of SVM which may be caused by the singularity problem in calculating the matrix inverse. Similar results can be found when using β and γ frequency bands features.

Table 8: Average results (%) of different algorithms on EEG-based emotion recognition.

Freq. Band	Mean±Std		
	β	γ	total
SVM	75.24±14.00	76.84±12.76	76.62±13.12
ELM	72.96±12.61	73.51±12.02	72.71±12.23
RELM	77.79±12.79	78.17±13.02	78.10±12.72
GELM	79.07±12.94	79.93±13.24	80.25±11.92
DMELM	**80.59±12.17**	**80.82±12.66**	**81.01±12.24**

Figure 8 show the average confusion matrices of the five algorithms based on 310 DE features. We can see that the positive and neutral states are much easier to be recognized while the negative state is difficult to estimate. The DMELM can respectively obtain 5% and 7% accuracy improvements when estimating the negative state w.r.t. GELM and SVM.

5. Conclusion

In this paper, we have proposed a discriminative manifold extreme learning machine, termed DMELM, which simultaneously takes the discriminative information and manifold structure of data into account. We constructed the within-class graph and between-class graph to depict the discriminative information in local neighborhood around each data point. DMELM was formulated by incorporating a graph regularizer into ELM objective, which is based on a unified graph Laplacian matrix of both graphs. Our experimental results demonstrated that our proposed DMELM achieves excellent performance in both image classification and EEG-based emotion recognition.

Most existing ELM models are focusing on supervised learning scenarios while little effort was made to extend ELM into unsupervised learning field. Thus, for our future work, it is of great significance to put ELM into learning applications with only unlabeled data.

Table 7: EEG-based emotion recognition results (%) of different models on six subjects.

Subject A	Session 1			Session 2			Session 3		
	β	γ	total	β	γ	total	β	γ	total
SVM	84.10	81.50	82.59	65.46	67.27	**75.65**	57.15	59.54	59.90
ELM	80.71	79.12	81.50	63.15	63.29	65.90	59.39	58.09	57.37
RELM	84.39	82.23	83.96	66.47	69.51	70.16	64.96	61.56	61.78
GELM	**85.19**	86.64	84.39	66.18	75.07	70.09	66.26	61.92	63.95
DMELM	**85.19**	**88.01**	**85.26**	**68.71**	**75.87**	72.40	**68.93**	**65.32**	**65.39**
Subject B	Session 1			Session 2			Session 3		
	β	γ	total	β	γ	total	β	γ	total
SVM	**90.17**	89.52	88.15	69.44	**70.66**	65.82	78.97	77.24	71.82
ELM	84.61	86.63	82.59	68.42	65.25	65.39	80.20	72.11	69.94
RELM	88.08	90.17	88.15	69.73	67.77	68.28	81.65	77.46	73.92
GELM	88.08	90.90	89.45	69.65	69.22	69.15	82.30	77.75	**79.48**
DMELM	89.96	**91.19**	**92.63**	**71.89**	69.73	**72.47**	**84.39**	**79.55**	79.33
Subject C	Session 1			Session 2			Session 3		
	β	γ	total	β	γ	total	β	γ	total
SVM	77.24	76.37	76.52	90.03	89.45	91.11	58.60	59.18	61.20
ELM	74.93	71.46	71.97	86.56	82.73	81.79	51.81	57.15	55.35
RELM	77.67	76.81	79.34	90.46	90.32	91.04	52.53	58.82	59.54
GELM	**79.19**	**80.92**	82.37	90.75	89.96	92.99	54.62	58.45	**67.85**
DMELM	78.25	77.82	**83.53**	**92.34**	**90.46**	**93.14**	**59.61**	**60.26**	60.48
Subject D	Session 1			Session 2			Session 3		
	β	γ	total	β	γ	total	β	γ	total
SVM	92.99	90.68	96.68	88.09	91.98	91.04	**97.18**	**96.32**	97.25
ELM	92.34	91.91	89.67	86.78	90.03	89.02	87.64	87.93	92.70
RELM	95.30	94.08	96.60	92.70	93.35	95.88	95.16	95.59	97.11
GELM	96.89	96.60	96.68	95.30	96.89	**96.89**	96.82	95.74	96.53
DMELM	**97.18**	**96.89**	**97.11**	**95.74**	**97.25**	96.82	96.82	**96.32**	**97.54**
Subject E	Session 1			Session 2			Session 3		
	β	γ	total	β	γ	total	β	γ	total
SVM	67.12	76.89	70.01	53.90	70.66	60.19	63.08	63.29	73.99
ELM	67.05	75.79	68.14	57.95	68.35	61.85	61.99	61.85	66.84
RELM	72.54	78.18	71.39	72.25	72.54	73.05	70.52	64.02	70.09
GELM	74.64	80.35	73.19	74.35	73.92	73.19	73.77	66.98	**74.57**
DMELM	**76.37**	**81.36**	**75.94**	**75.07**	**76.66**	**75.43**	**75.65**	**68.14**	71.10
Subject F	Session 1			Session 2			Session 3		
	β	γ	total	β	γ	total	β	γ	total
SVM	73.19	69.80	73.19	59.25	58.82	56.50	88.29	93.86	87.50
ELM	73.27	68.35	73.48	55.78	56.36	52.46	80.78	86.71	82.80
RELM	78.90	86.05	77.17	57.95	56.58	58.82	88.95	91.98	89.60
GELM	80.20	85.98	84.32	57.88	57.08	59.25	91.18	94.29	90.10
DMELM	**81.72**	**87.14**	**85.55**	**59.32**	**59.39**	**62.28**	**93.43**	**94.08**	**91.76**

Acknowledgement

This work was partially supported by the National Basic Research Program of China (No.2013CB329401), the National Natural Science Foundation of China (No.61272248), and the Science and Technology Commission of Shanghai Municipality (No.13511500200). The first author was supported by China Scholarship Council (No.201206230012).

References

[1] D. E. Rumelhart, G. E. Hinton, R. J. Williams, Learning representations by back-propagating errors, Nature 323 (6088) (1986) 53–536.

[2] G.-B. Huang, Q.-Y. Zhu, C.-K. Siew, Extreme learning machine: a new learning scheme of feedforward neural networks, in: Proceedings of IEEE International Joint Conference on Neural Networks, vol. 2, 985–990, 2004.

[3] G.-B. Huang, Q.-Y. Zhu, C.-K. Siew, Extreme learning machine: theory and applications, Neurocomputing 70 (1) (2006) 489–501.

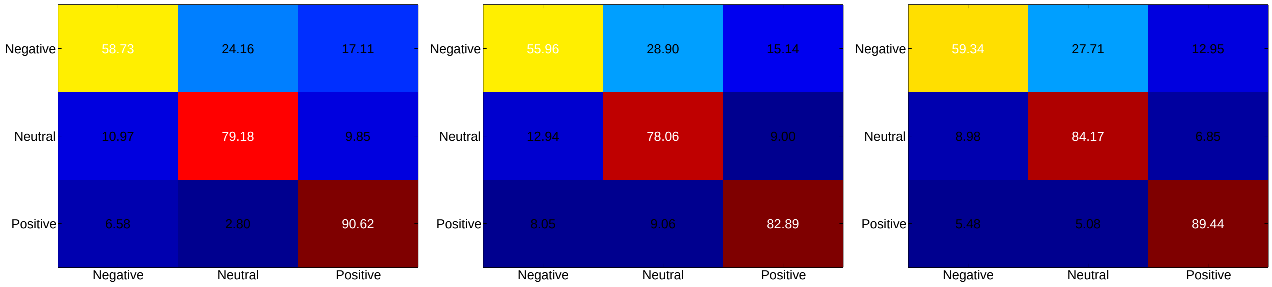
[4] G.-B. Huang, L. Chen, C.-K. Siew, Universal approximation using incremental constructive feedforward networks with random hidden nodes, IEEE Transactions on Neural Networks 17 (4) (2006) 879–892.

[5] R. Zhang, Y. Lan, G.-B. Huang, Z.-B. Xu, Universal approximation of extreme learning machine with adaptive growth of hidden nodes, IEEE Transactions on Neural Networks and Learning Systems 23 (2) (2012) 365–371.

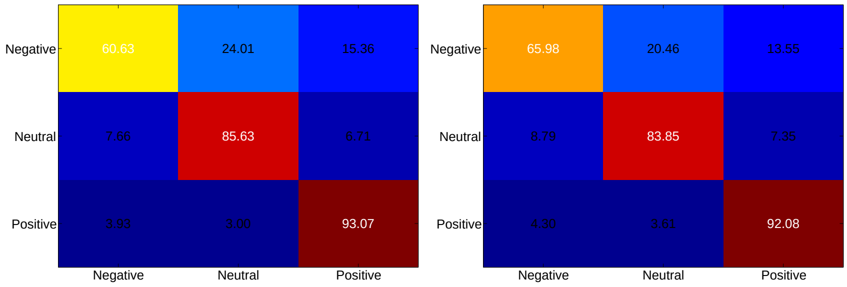
[6] G.-B. Huang, H. Zhou, X. Ding, R. Zhang, Extreme learning machine for regression and multiclass classification, IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 42 (2) (2012) 513–529.

[7] H.-J. Rong, G.-B. Huang, N. Sundararajan, P. Saratchandran, Online sequential fuzzy extreme learning machine for function approximation and classification problems, IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 39 (4) (2009) 1067–1072.

[8] Q. Liu, Q. He, Z. Shi, Extreme support vector machine clas-



(a) Confusion matrix of SVM (b) Confusion matrix of ELM (c) Confusion matrix of RELM



(d) Confusion matrix of GELM (e) Confusion matrix of DMELM

Figure 8: Confusion matrices of different algorithms on EEG-based emotion recognition.

sifier, in: Proceedings of the 12th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining, 222–233, 2008.

[9] B. Frénay, M. Verleysen, Using SVMs with randomised feature spaces: an extreme learning approach, in: Proceedings of the 18th European Symposium on Artificial Neural Networks, 315–320, 2010.

[10] X. Liu, C. Gao, P. Li, A comparative analysis of support vector machines and extreme learning machines, Neural Networks 33 (2012) 58–66.

[11] L. L. C. Kasun, H. Zhou, G.-B. Huang, C. M. Vong, Representational learning with extreme learning machine for big data, IEEE Intelligent Systems 28 (6) (2013) 31–34.

[12] L.-C. Shi, B.-L. Lu, EEG-based vigilance estimation using extreme learning machines, Neurocomputing 102 (2013) 135–143.

[13] B.-L. Lu, M. Ito, Task decomposition and module combination based on class relations: a modular neural network for pattern classification, IEEE Transactions on Neural Networks 10 (5) (1999) 1244–1256.

[14] X.-L. Wang, Y.-Y. Chen, H. Zhao, B.-L. Lu, Parallelized extreme learning machine ensemble based on min–max modular network, Neurocomputing 128 (2014) 31–41.

[15] Y. Peng, S. Wang, X. Long, B.-L. Lu, Discriminative graph regularized extreme learning machine and its application to face recognition, Neurocomputing 149 (2015) 340–353.

[16] G. Huang, S. Song, J. N. Gupta, C. Wu, Semi-supervised and unsupervised extreme learning machines, IEEE Transactions on Cybernetics 44 (12) (2014) 2405–2417.

[17] N.-Y. Liang, G.-B. Huang, P. Saratchandran, N. Sundararajan, A fast and accurate online sequential learning algorithm for feedforward networks, IEEE Transactions on Neural Networks 17 (6) (2006) 1411–1423.

[18] Y. Lan, Y. C. Soh, G.-B. Huang, Ensemble of online sequential extreme learning machine, Neurocomputing 72 (13) (2009) 3391–3395.

[19] J. Zhao, Z. Wang, D. S. Park, Online sequential extreme learning machine with forgetting mechanism, Neurocomputing 87 (2012) 79–89.

[20] Y. Xu, Z. Y. Dong, J. H. Zhao, P. Zhang, K. P. Wong, A reliable intelligent system for real-time dynamic security assessment of power systems, IEEE Transactions on Power Systems 27 (3) (2012) 1253–1263.

[21] S. Suresh, R. Venkatesh Babu, H. Kim, No-reference image quality assessment using modified extreme learning machine classifier, Applied Soft Computing 9 (2) (2009) 541–552.

[22] M. Pal, A. E. Maxwell, T. A. Warner, Kernel-based extreme learning machine for remote-sensing image classification, Remote Sensing Letters 4 (9) (2013) 853–862.

[23] Y. Song, J. Crowcroft, J. Zhang, Automatic epileptic seizure detection in EEGs based on optimized sample entropy and extreme learning machine, Journal of Neuroscience Methods 210 (2) (2012) 132–146.

[24] M. Termenon, M. Graña, A. Barrós-Loscertales, C. Ávila, Extreme learning machines for feature selection and classification of cocaine dependent patients on structural MRI data, Neural processing letters 38 (3) (2013) 375–387.

[25] Y. Kaya, M. Uyar, A hybrid decision support system based on rough set and extreme learning machine for diagnosis of hepatitis disease, Applied Soft Computing 13 (8) (2013) 3429–3438.

[26] S. Samet, A. Miri, Privacy-preserving back-propagation and extreme learning machine algorithms, Data & Knowledge Engineering 79 (2012) 40–61.

[27] Q. He, C. Du, Q. Wang, F. Zhuang, Z. Shi, A parallel incremental extreme SVM classifier, Neurocomputing 74 (16) (2011) 2532–2540.

[28] Q. He, T. Shang, F. Zhuang, Z. Shi, Parallel extreme learning machine for regression based on MapReduce, Neurocomputing 102 (2013) 52–58.

[29] S. Decherchi, P. Gastaldo, A. Leoncini, R. Zunino, Efficient digital implementation of extreme learning machines for classification, IEEE Transactions on Circuits and Systems II: Express Briefs 59 (8) (2012) 496–500.

[30] G.-B. Huang, D. H. Wang, Y. Lan, Extreme learning machines: a survey, International Journal of Machine Learning and Cybernetics 2 (2) (2011) 107–122.

[31] G.-B. Huang, An insight into extreme learning machines: random neurons, random features and kernels, Cognitive Computation 6 (2014) 376–390.

[32] S. T. Roweis, L. K. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science 290 (5500) (2000) 2323–2326.

[33] J. B. Tenenbaum, V. De Silva, J. C. Langford, A global geomet-

ric framework for nonlinear dimensionality reduction, Science 290 (5500) (2000) 2319–2323.

[34] M. Belkin, P. Niyogi, Laplacian eigenmaps and spectral techniques for embedding and clustering, in: Proceeding of Advances in Neural Information Processing Systems, 585–591, 2001.

[35] J. M. Lee, Introduction to smooth manifolds, 2001.

[36] Z. Zhang, H. Zha, Principal manifolds and nonlinear dimensionality reduction via tangent space alignment, SIAM Journal on Scientific Computing 26 (1) (2005) 313–338.

[37] R. Hadsell, S. Chopra, Y. LeCun, Dimensionality reduction by learning an invariant mapping, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, 1735–1742, 2006.

[38] D. Cai, X. He, X. Wu, J. Han, Non-negative matrix factorization on manifold, in: Proceedings of IEEE International Conference on Data Mining, 63–72, 2008.

[39] D. Cai, X. He, J. Han, T. S. Huang, Graph regularized nonnegative matrix factorization for data representation, IEEE Transactions on Pattern Analysis and Machine Intelligence 33 (8) (2011) 1548–1560.

[40] N. Guan, D. Tao, Z. Luo, B. Yuan, Manifold regularized discriminative nonnegative matrix factorization with fast gradient descent, IEEE Transactions on Image Processing 20 (7) (2011) 2030–2048.

[41] D. Cai, X. He, J. Han, Locally consistent concept factorization for document clustering, IEEE Transactions on Knowledge and Data Engineering 23 (6) (2011) 902–913.

[42] M. Zheng, J. Bu, C. Chen, C. Wang, L. Zhang, G. Qiu, D. Cai, Graph regularized sparse coding for image representation, IEEE Transactions on Image Processing 20 (5) (2011) 1327–1336.

[43] X. Lu, Y. Wang, Y. Yuan, Graph-regularized low-rank representation for destriping of hyperspectral images, IEEE Transactions on Geoscience and Remote Sensing 51 (7) (2013) 4009–4018.

[44] X. He, D. Cai, Y. Shao, H. Bao, J. Han, Laplacian regularized Gaussian mixture model for data clustering, IEEE Transactions on Knowledge and Data Engineering 23 (9) (2011) 1406–1418.

[45] X. Li, T. Jiang, K. Zhang, Efficient and robust feature extraction by maximum margin criterion, IEEE Transactions on Neural Networks 17 (1) (2006) 157–165.

[46] A. E. Hoerl, R. W. Kennard, Ridge regression: biased estimation for nonorthogonal problems, Technometrics 42 (1) (2000) 80–86.

[47] M. Belkin, P. Niyogi, V. Sindhwani, Manifold regularization: A geometric framework for learning from labeled and unlabeled examples, The Journal of Machine Learning Research 7 (2006) 2399–2434.

[48] F. R. Chung, Spectral graph theory, vol. 92, American Mathematical Society, 1997.

[49] S. Yan, D. Xu, B. Zhang, H.-J. Zhang, Q. Yang, S. Lin, Graph embedding and extensions: a general framework for dimensionality reduction, IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (1) (2007) 40–51.

[50] Q. He, X. Jin, C. Du, F. Zhuang, Z. Shi, Clustering in extreme learning machine feature space, Neurocomputing 128 (2014) 88–95.

[51] S. Lin, X. Liu, J. Fang, Z. Xu, Is extreme learning machine feasible? A theoretical assessment (Part I), IEEE Transactions on Neural Networks and Learning Systems 26 (1) (2015) 7–20.

[52] X. Liu, S. Lin, J. Fang, Z. Xu, Is extreme learning machine feasible? A theoretical assessment (Part II), IEEE Transactions on Neural Networks and Learning Systems 26 (1) (2015) 21–34.

[53] L. Zhang, M. Yang, X. Feng, Sparse representation or collaborative representation: Which helps face recognition?, in: Proceedings of IEEE International Conference on Computer Vision, 471–478, 2011.

[54] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, Y. Ma, Robust face recognition via sparse representation, IEEE Transactions on Pattern Analysis and Machine Intelligence 31 (2) (2009) 210–227.

[55] J.-Y. Zhu, W.-L. Zheng, Y. Peng, B.-L. Lu, EEG-based emotion recognition using discriminative graph regularized extreme learning machine, in: Proceedings of IEEE International Joint Conference on Neural Networks, 525–532, 2014.

[56] L.-C. Shi, Y.-Y. Jiao, B.-L. Lu, Differential entropy feature for EEG-based vigilance estimation, in: Proceedings of 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 6627–6630, 2013.

[57] L.-C. Shi, B.-L. Lu, Off-line and on-line vigilance estimation based on linear dynamical system and manifold learning, in: Proceedings of 32nd Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 6587–6590, 2010.

[58] Y. Peng, J.-Y. Zhu, W.-L. Zheng, B.-L. Lu, EEG-based emotion recognition with manifold regularized extreme learning machine, in: Proceedings of 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 974–977, 2014.