Face Recognition using Graph Extreme Learning Machine with L_{21}-norm Regularization

Mohanad Abd Shehab[1], Nihan Kahraman[2*], Gokhan Bilgin[3]

[1,2]Yildiz Technical University, Department of Electronics and Communication Engineering, Davutpasa, Istanbul, Turkey
[1]f0414039@std.yildiz.edu.tr, [2]corresponding author, nicoskun@yildiz.edu.tr
[3] Yildiz Technical University, Department of Computer Engineering, Davutpasa, Istanbul, Turkey
[3]gbilgin@yildiz.edu.tr

Abstract

Mathematical approaches including subspace learning (SL), graph Laplacian, and L_{21}-norm regularization can be combined effectively with baseline ELM to produce a novel extension of ELM termed Face Recognition using Graph ELM with L_{21}-norm Regularization (GELML21) that can improve the robustness and compactness of ELM.
In this paper, subspace learnings that exploit the geometrical structure of face data were utilized to produce discriminative features.
ELM activation functions and some SL approaches have almost nonlinear characteristics that enhance features extraction performance while destroying the local consistency properties. For this reason, the graph Laplacian was used to regulate the samples in the same class to similar outputs. After that, L_{21} -norm algorithm with proved convergence was introduced to solve the resultant optimization problem and yield distinct hidden layer that clearly enhanced the accuracy.
Our experimental results have demonstrated that GELML21 possessed excellent performance in face recognition in comparison with ELM variants and several popular state-of-the-art classification methods.

Keywords: Extreme learning machines; Subspace learning; Graph Laplacian; L_{21}-norm regularization; Face and object recognition.

1. Introduction

Extreme learning machine (ELM) is a single hidden layer feedforward neural network (SLFN) with a sufficient number of hidden neurons and almost any nonlinear activation function. It uses arbitrarily chosen input weights and biases without any tuning and can universally approximate any continuous functions or any compact input sets with zero or randomly small error [1]. ELM has been proved as an efficient and effective learning algorithm for classification, regression and many other tasks [1-6].

However, there are still some limitations in ELM. For example, although learning efficiency increases with randomly selected parameters, the ELM algorithm may appear less stable. In order to tackle this issue, regularization approaches like Regularized Extreme Learning Machine (RELM) with ridge regression [7] have been applied in the ELM algorithms. Moreover, better generalization performance and higher stability have been reported with Graph Extreme Learning Machine (GELM) [8, 9]. In GELM, constructing an adjacent graph depending on the label information of training samples is involved to formulate the graph regularization term which is used to learn similar outputs for samples from the same class. RELM and GELM model have a closed form solution as the standard ELM and thus the output weights can be obtained efficiently, but slower learning efficiency is expected.

2. Previous and Proposed ELM Works

2.1 Baseline, Regularized, and Graph ELM (ELM, RELM and GELM)

The baseline ELM [1] was firstly proposed for classification and regression. It employs nonlinear piecewise continuous activation functions which are used for approximating any continuous target function. Consider a data set containing N-training samples $(x_j, t_j) \in R^n \times R^m$, $j = 1,2,\dots,N$ with n, m : number of input attributes and output classes, $G(x)$ is the activation function, the hidden layer output function for N input and L hidden nodes can be written as Eq.1 and the estimated output y_j is mathematically modeled as Eq.2

$$h(x) = [G(w_1, b_1, x_1), \dots, G(w_L, b_L, x_N)] \quad (1)$$

$$y_j = \sum_{i=1}^{L} \beta_i . G(w_i x_j + b_i) = \beta . h(x) = t_j + \epsilon_j \quad (2)$$
$$j = 1, \dots, N$$

where β_i is the output weight vector connecting the ith hidden nodes and the output nodes; w_i , b_i are the randomly chosen input weight and bias vectors connecting the ith hidden nodes; x_j is the input attribute and y_j is the actual output.

ELM aims to minimize the relative error as Eq.3,

$$min \sum_{j=1}^{L} \|y_j - t_j\| \quad (3)$$

The desired output targets t_j can be expressed as in Eq.4.

$$\sum_{i=1}^{L} \beta_i . G(w_i x_j + b_i,) = t_j \quad j = 1, \dots, N \quad (4)$$

In matrix form, Eq.4 is equivalent to $H.\beta = T$ (5)

$$H(w_1, \cdots, w_L, b_1, \cdots, b_L, x_1, \cdots, x_N) = \begin{bmatrix} G(w_1, b_1, x_1) & \cdots & G(w_L, b_L, x_1) \\ \vdots & \cdots & \vdots \\ G(w_1, b_1, x_N) & \cdots & G(w_L, b_L, x_N) \end{bmatrix}_{N \times L} \quad (6)$$

$$\beta = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_L^T \end{bmatrix}_{L \times m} \text{ and } T = \begin{bmatrix} t_1^T \\ \vdots \\ t_N^T \end{bmatrix}_{N \times m} \text{ where}$$

$$T_{ij} = \begin{cases} 1 \ for\ vectors\ of\ class_i = j \\ -1\ for\ vectors\ of\ class_i \neq j \end{cases} \quad (7)$$

The ith column of T is the output target of the ith hidden nodes with respect to input $x_1,\ldots, x_N$.

2.1.1. Baseline ELM

Based on the minimization of the Least Square error, β can be solved as in Eq.8.

$$\beta = \arg min_\beta \|H.\beta - T\|_2^2 = H^\dagger T \quad (8)$$

where $H^\dagger = (H^T H)^{-1} H^T$ is Moore-Penrose generalized pseudo inverse.

2.1.2. Regularized ELM

The matrix $H^T H$ may be close to singular and the pseudo inverse, $H^\dagger$ is prone to numerical instabilities. As a result, a small regularization term (λ) should be included. The optimization function for RELM will be as in Eq.9.

$$min_{\beta,\lambda} \frac{1}{2}\|H.\beta - T\|_2^2 + \frac{\lambda}{2}\|\beta\|_2^2 \quad (9)$$

where $\|\beta\|_2^2 = \sum_{i=1}^{L}\|\beta_i\|_2^2$ is $L2$-ridge regularization term of vectors β_i and λ is the regularization parameter that balance the influence of error term and the model complexity.

Solving Eq.9 leads to $\beta = H^\dagger T$ where

$$H^\dagger = (H^T H + \lambda I)^{-1} H^T \quad \text{for } N > L$$

and $$H^\dagger = H^T (HH^T + \lambda I)^{-1} \quad \text{for } L > N \quad (10)$$

2.1.3. Graph ELM

Almost all activation functions in ELM are of non-linear characteristics that can destroy the local consistency properties in the data set. It can utilize the graph Laplacian within ELM optimization model through exploiting the class similarity of intrinsic construction in order to enhance the overall model accuracy as:

$$min \sum_{ij}\|y_i - y_j\|_2^2 W_{ij} = Tr(Y Ł Y^T) \text{ where } Y = H.\beta \quad (11)$$

By combining the previous L_2 regularization term and the above mentioned graph regularization term with suitable regularization parameters (λ_1 and λ_2), it can formulate the objective function as:

$$min_\beta \|H.\beta - T\|_2^2 + \lambda_1 Tr(H\beta Ł \beta^T H^T) + \lambda_2\|\beta\|_2^2 \quad (12)$$

The graph Laplacian matrix Ł is constructed through

$$Ł = V - W \quad (13.a)$$

where V is a diagonal matrix with

$$V_{ii} = \sum_{j=1}^{N} W_{ij} \quad (13.b)$$

and W_{ij} reflects the similarity between x_i and x_j as:

$$W_{ij} = \begin{cases} 1 & x_i \text{ and } x_j \text{ are of same class} \\ 0 & \text{otherwise} \end{cases} \quad (13.c)$$

Solving Eq.12 yields:

$$H^\dagger = (H^T H + \lambda_1 H^T Ł H + \lambda_2 I)^{-1} H^T \quad (14)$$

and hence, $\beta = H^\dagger T$

2.2. The Proposed ELM with L_{21}-norm

The objective function of the ELM classification problem with graph Laplacian and L_{21}-norm minimization can be represented as in Eq.15.

$$min_\beta \|H.\beta - T\|_2^2 + \lambda_1 Tr(Y Ł Y^T) + \lambda_2\|\beta\|_{21}^2 \quad (15)$$

where λ_1 and λ_2 are the penalty terms used to trade-off between the parameters

As $Y = H.\beta \Longrightarrow$

$$min_\beta \|H.\beta - T\|_2^2 + \lambda_1 Tr(H\beta.Ł.\beta^T H^T) + \lambda_2\|\beta\|_{21}^2 \quad (16)$$

Derive Eq.16 with respect to β yields

$$2\beta(H^T H) - 2(H^T T) + 2\lambda_1\beta(H^T Ł H) + 2\lambda_2\beta D = 0 \Longrightarrow$$

$$\beta = (H^T H + \lambda_1 H^T Ł H + \lambda_2 D)^{-1}.H^T T \quad (17)$$

Where $D \in R^{L\times L}$ is a diagonal matrix whose ith-diagonal element is $D_{ii} = \frac{1}{2\|\beta_i\|_2}$ (18)

It can use the baseline ELM solution with Eq.8 as initial β_o for finding the initial diagonal matrix D as in Eq.18 that can be used iteratively with Eq.17 to find the converged β solution (optimum β).

The initial hidden output weight β_o is found with Eq.19 as:

$$\beta = \beta_o = H^\dagger T = (H^T H)^{-1} H^T.T \quad (19)$$

Our proposed algorithm needs few iterations (t) (less than predefined maximum iteration number t_{max}) to converge. It can be explained as follows:

Algorithm: Embedded Graph with L21-norm Regularized ELM

Input: A training data set containing N-samples is given as $\{(x_i, t_i) \mid x_i \in R^n, t_i \in R^m, i = 1,\cdots,N\}$,i.e, $X \in R^{N\times n}$ and $T \in R^{N\times m}$ where $T_{ij} = \begin{cases} 1 & for\ vectors\ of\ class_i = j \\ -1 & for\ vectors\ of\ class_i \neq j \end{cases}$

with any suitable activation function $G(x)$. Depend on Dimensional Reduction methods, the input X is reduced to $X \in R^{N\times d}$, i.e, $d < n$

Assign input weights w_i, bias b_i, hidden nodes number L and the regularization parameters $\lambda_1$and λ_2.

Output: Output weight matrix β

Initial

- Calculate the hidden layer output matrix H and graph Laplacian matrix Ł using Eq.6 and Eqs.13
- Set t=0, calculate the initial β_o using Eq.19

Repeat

- Update the diagonal matrix D and β with

$$D^{t+1} = \begin{bmatrix} \frac{1}{2\|\beta_1^t\|_2} & & \\ & \ddots & \\ & & \frac{1}{2\|\beta_L^t\|_2} \end{bmatrix}$$

$$\beta = (H^T H + \lambda_1 H^T Ł H + \lambda_2 D)^{-1}.H^T T, \qquad t = t + 1$$

until β converges or $t > tmax$

3. Experimental Study

In this section, we present experiments on a prevalent benchmark face dataset ORL and AR in order to evaluate the proposed approach in image classification with face recognition. These datasets are briefly detailed below:

1) The ORL dataset [10]: consists of 400 facial images depicting 40 persons, the images were captured at different times, varying the lighting, facial expressions.

2) The AR dataset [11]: consists of over 4000 facial images depicting 50 male and 50 female faces. These images having a frontal facial pose, performing several expressions, in different illumination conditions and with some occlusions.

Example of images of these two face recognition dataset is illustrated in Figure 1.

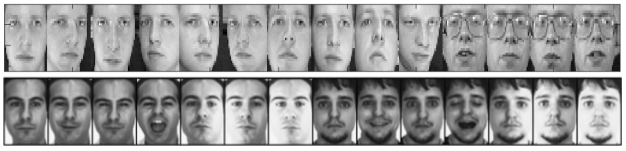

Fig. 1. The two rows show the fourteen image samples from the ORL and AR databases, respectively.

Table 1 presents briefly the information concerning the data sets used in our experiments where the training and testing samples (4 for training and 20 remaining for testing) are chosen randomly.

Table 1. Details of the datasets

Data Set	Size	# of Features	# of Classes
ORL	400	1024	40
AR	300	700	100

The experimental results are based on the average of 20 independent trials for regular size datasets. All simulations were implemented using MATLAB 8.6 (R2015b) environment and performed on an (Intel Core i5, 2.4 GHz CPU, 4GB RAM) computer. For all datasets, we chose randomly [2, 3, 4, 5, 10, 20 30] samples per subject for training and the rest for testing. Our proposed algorithm GELML21 was tested and compared with three well-known approaches: standard ELM (ELM) [1], L2-Regularized ELM (RELM) [7], Graph ELM (GELM) [8].
The only activation function used for all experiments was the "sigmoid" function.
Table 2 shows the mean of the classification accuracy with standard deviation for different trains per subject on the ORL using ELM, RELM, GELM, and GELML21.

Table 2. Performance (mean ± standard deviation) % for ELM methods on ORL dataset

Algorithms	2 Train	3 Train	4 Train	5 Train
ELM	78.7±2.6	84.7±2.2	89.8±1.8	93.5±1.4
RELM	82.1±2.3	87.9±1.8	93.2±1.6	95.0±1.5
GELM	86.7±2.3	90.4±1.8	95.0±1.4	96.0±1.1
GELML21	**87.8±2.2**	**91.3±1.7**	**96.3±1.4**	**97.2±1.1**

Figure 2 presents the histogram comparison of the accuracies of the corresponding Table 2 between ELM, L2-RELM, GELM, and GELML21 on the ORL.

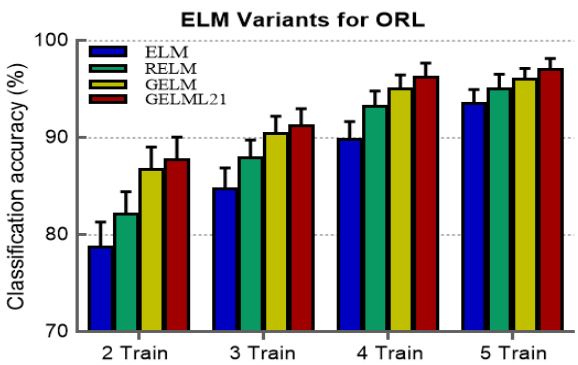

Fig 2. ELM methods comparison testing accuracies for face recognition on dataset ORL

From the results, it is clear that the GELML21 can achieve superior performance over other ELM variants. This means that the hidden layer generated with the GELML21 method is more effective due to the including of the most discriminative features. Figure 3 represents the execution time comparison for different training samples of ELM variants on the ORL dataset. In general, the GELML21 algorithm is based on an iterative solution, thereby it has a higher execution time than other ELM methods.

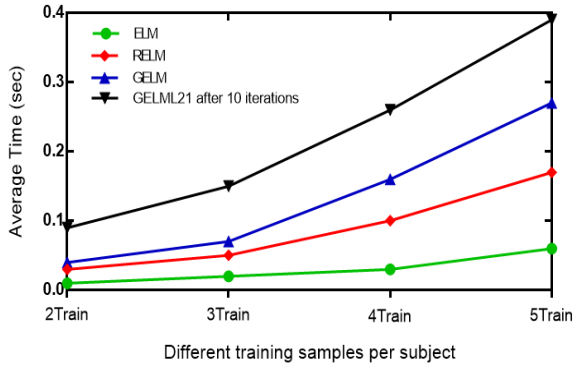

Fig. 3. Comparison of time for different training samples of ELM variants on the ORL dataset

GELM and GELML21 are working on both R^n input and R^L feature spaces, as almost (n>>L). They need to enlarge hidden nodes number for providing global solution. Incorporating subspace learning can reduce the input data from R^n to R^d (d<<n) with effective features for improving the accuracy and minimizing the operation time. In this work, GELML21 exploits different feature spaces determined by applying many non-supervised and supervised dimensional reduction approaches with linear and non-linear feature mapping [12] as principal component analysis (PCA), linear discriminant analysis (LDA), kernel principal component analysis (KPCA), kernel discriminant analysis (KDA).

Table 3 shows a comparison that reveals GELML21 classification accuracies for subspace learning methods (PCA, LDA, KPCA and KDA) with $L = 500$ on ORL dataset. Since all methods are of comparable standard deviations, the mean of accuracies are written as in Table 3.

Table 3. Accuracy of different training sampling on ORL dataset for $L = 500$ hidden nodes

	GELML21 (PCA)	GELML21 (LDA)	GELML21 (KPCA)	GELML21 (KDA)
2Train	77.50	**83.6**	42.6	52.5
3Train	84.1	**87.8**	48.4	61.1
4Train	87.6	**93.2**	43.8	70.3
5Train	90.7	**94.1**	45.2	76.4

It is clear from Table 3 that representing the data in different feature spaces in training data can impact the ELM performance. In addition, the best reduced feature space used with GELML21 is LDA, this is because of LDA is supervised method and the distribution of the data in the face images is normally distributed that meet the LDA properties.

Finally, as in the Table 4, we compare the performance of GELML21 approach on the AR dataset with the state of the art algorithms, which are widely used in the face recognition: nearest

neighbor classifier (NN), linear regression classifier (LRC), support vector machine (SVM) [13], sparse representation-based classification (SRC) [14], collaborative representation-based regularized least square (CRC RLS) [15]. As can be seen, GELML21 is more accurate and outperforms over other classification methods in all dimension cases and under different conditions.

Table 4. Different state of the art classifier accuracies (%) for AR face recognition dataset for different dimension

	Testing accuracy (%)		
Algorithms	**Dim d=50**	**Dim d=150**	**Dim d=n=300**
1NN	67.8	70.2	71.3
LRC	71.1	74.6	76.0
SVM	68.3	74.7	75.4
SRC	81.9	90.1	**93.3**
CRC_RLS	80.7	**91.0**	**93.7**
GELML21	**85.0**	**92.1**	**93.9**

4. Conclusion

In this study, an efficient algorithm with proved convergence called GELML21 is proposed. It can utilize the graph Laplacian and L_{21}-norm with different subspace learning to solve the ELM resultant optimization function with a few iterations, resulting in a more compact and distinct hidden layer, hence improve the classification performance.

GELML21 is examined in widely adopted face recognition data sets with many cases and sizes, also it is compared with relevant ELM works including ordinary, regularized and graph ELMs. Clearly, the proposed GELML21 is more accurate than various ELM variants and can achieve much performance gain over other state-of-the-art classification methods.

For future work, enhancement through generalization of the proposed algorithm can be developed with different topics and datasets.

Acknowledgment

This work was partially supported by the Al-Mustansiriyah University, Bagdad, Iraq.

5. References

[1] Huang GB, Zhu QY, Siew CK, "Extreme learning machine: a new learning scheme of feedforward neural networks", Neural Networks, Proceedings IEEE International Joint Conference on. 2004.

[2] Wang Y, Cao F, Yuan Y, "A Study on Effectiveness of Extreme Learning Machine", National Natural Science Foundation of China (No. 60873206), 2014.

[3] Luo M, Zhang K, "A hybrid approach combining extreme learning machine and sparse representation for image classification", Engineering Applications of Artificial Intelligence; 27: pp:228-235, 2014.

[4] Huang GB, Song S, You K, "Trends in extreme learning machines: A review", Neural Networks 2015.

[5] Liu X, Wang L, Huang GB, Zhang J, Yin J "Multiple Kernel Extreme Learning Machine", Article in Neurocomputing, 2013, DOI: 10.1016/j.neucom.2013.09.072.

[6] Miche Y, Sorjamaa A, Bas P, Simula O, Jutten C, Lendasse A, "OP-ELM: Optimally Pruned Extreme Learning Machine", IEEE Transactions on Neural Networks, pp: 158-162, 2010.

[7] Huang GB, Zhou H, Ding X & Zhang R, "Extreme learning machine for regression and multiclass classification", IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, 513–529, 2012.

[8] Y. Peng, S. Wang, X. Long, and B. L. Lu, "Discriminative graph regularized Extreme Learning Machine for face recognition", Neurocomputing, vol. 149, pp. 340–353, 2015.

[9] A. Iosifidis, A. Tefas and I. Pitas, "Graph Embedded Extreme Learning Machine", IEEE Transactions on Cybernetics,

[10]http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html

[11] A.M. Martinez, "The AR Face Database", CVC Technical Report 24.

[12] D. Cai, X. He, and J. Han., "Spectral regression for efficient regularized subspace learning", In Proc. Int. Conf. Computer Vision (ICCV'07), 2007.

[13] Suykens JA, Vandewalle J, "Least squares support vector machine classifiers", Neural Processing letters, 9(3) pp: 293–300, 1999.

[14] 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), 210–227, 2009.

[15] Zhang L, Yang M, Feng X, "Sparse representation or collaborative representation: which helps face recognition? ", Proceeding of IEEE International Conference on Computer Vision, 471–478, 2011.