Class-wise Deep Dictionaries for EEG Classification

Prerna Khurana
IIIT Delhi
New Delhi, India
prerna@iiitd.ac.in

Angshul Majumdar
IIIT Delhi
New Delhi, India
angshul@iiitd.ac.in

Rabab Ward
University of British Columbia
Vancouver, Canada
rababw@ece.ubc.ca

***Abstract*— In this work we propose a classification framework called class-wise deep dictionary learning (CWDDL). For each class, multiple levels of dictionaries are learnt using features from the previous level as inputs (for first level the input is the raw training sample). It is assumed that the cascaded dictionaries form a basis for expressing test samples for that class. Based on this assumption sparse representation based classification is employed. Benchmarking experiments have been carried out on some deep learning datasets (MNIST and its variations, CIFAR and SVHN); our proposed method has been compared with Deep Belief Network (DBN), Stacked Autoencoder, Convolutional Neural Net (CNN) and Label Consistent KSVD (dictionary learning). We find that our proposed method yields better results than these techniques and requires much smaller run-times. The technique is applied for Brain Computer Interface (BCI) classification problems using EEG signals. For this problem our method performs significantly better than Convolutional Deep Belief Network(CDBN).**

***Index Terms*— dictionary learning, deep learning, EEG**

I. Introduction

In the last decade dictionary learning techniques have become popular in the signal processing and computer vision communities; the seminal work that initiated research in this area is the KSVD [1]. The main objective of dictionary learning is to learn a basis that can represent a class of signals; mostly in a sparse fashion. In signal processing these techniques are mostly used for solving inverse problems dealing with restoration, denoising, reconstruction, super-resolution etc.

For solving inverse problem the learning task is unsupervised; the only constraint is the sparsity of the learned coefficients. Researchers in computer vision introduced the notion of supervised dictionary learning. The idea is straightforward – to learn the dictionary / coefficients in a supervised fashion one just needs to add the corresponding penalty terms. Initial techniques proposed simple approaches which learnt specific dictionaries for each class [2]. Later approaches incorporated discriminative penalties into the dictionary learning framework such as softmax discriminative cost function, Fisher discrimination criterion, linear predictive classification error penalty and hinge loss function.

In recent years machine learning witnessed success and popularity of deep learning techniques. Convolutional Neural Network (CNN), Deep Boltzman Machine (DBM) and Stacked Denoising Autoencoder (SDAE) have been successful in various supervised and unsupervised learning scenarios. Motivated by the success of deep learning, we propose a classification framework based called deep dictionary learning. Our framework is based on the Sparse Representation based Classification (SRC) approach [3]; instead of using features from training classes as the basis for new test samples, we use learned dictionaries as the basis. This is a simple technique but yet yields competent results with more sophisticated deep learning architectures.

We have carried out two types of experiments. Since we are proposing a new classification framework, we test it on several benchmark deep learning datasets. The results show that our method yields better results than state-of-the-art deep learning tools; in fact our method features among the top-10 results on these datasets. In the next part of the experiment, we compare our algorithm for some BCI competition problems. Here we perform significantly better than state-of-the-art existing approaches.

The rest of the paper will be organized into several sections. A brief literature review on dictionary learning ensues in the following section. The proposed framework for deep dictionary learning is described in section 3. Experimental results are shown in section 4. The conclusions of this work are discussed in section 5.

II. Literature Review

A. Dictionary Learning

Early studies in dictionary learning wanted to learn a basis for representation. There were no constraints on the dictionary atoms or on the loading coefficients. The method of optimal directions (MOD) [4] was employed to solve the learning problem:

$$\min_{D,Z} \|X - DZ\|_F^2 \quad (1)$$

Here X is the training data, D is the dictionary to be learnt and Z consists of the loading coefficients. Today, we know this problem in the name of matrix factorization.

For problems in sparse representation, the objective is to learn a basis that can represent the samples in a sparse fashion (2), i.e. Z needs to be sparse. KSVD [1] is perhaps the most well known work in this respect, but the problem of learning sparse representations from overcomplete basis

dates back to the late 90's [5]. Fundamentally it solves a problem of the form:

$$\min_{D,Z} \|X - DZ\|_F^2 \text{ such that } \|Z\|_0 \le \tau \quad (2)$$

Dictionary learning is a bilinear (non-convex) problem; it is usually solved in an alternating fashion. In the first stage it learns the dictionary and in the next stage it uses the learned dictionary to sparsely represent the data.

Researchers in machine learning became interested in dictionary learning owing to its flexibility. Dictionary learning provides the opportunity to design dictionaries to yield not only sparse representation (e.g., curvelet, wavelet, and DCT) but also discriminative information. Initial techniques in discriminative dictionary learning propose naïve approaches which learn specific dictionaries for each class [6-8]. Later, discriminative penalties are introduced in dictionary learning framework to improve classification performance. One such technique is to include softmax discriminative cost function [9-11]; other discriminative penalties include Fisher discrimination criterion [12], linear predictive classification error [13, 14] and hinge loss function [15, 16]. In [17, 18] discrimination is introduced by forcing the learned features to map to corresponding class labels.

B. Sparse Representation based Classification

The SRC assumes that the training samples of a particular class approximately form a linear basis for a new test sample belonging to the same class. One can write the aforesaid assumption formally. If x_{test} is the test sample belonging to the k[th] class then,

$$x_{test} = \alpha_{c,1}x_{c,1} + \alpha_{c,2}x_{c,2} + ... + \alpha_{c,n_k}x_{c,n_k} + \eta \quad (3)$$

where $x_{c,i}$ are the training samples and η is the approximation error.

In a classification problem, the training samples and their class labels are provided. The task is to assign the given test sample with the correct class label. This requires finding the coefficients $\alpha_{c,i}$ in equation (3). Equation (3) expresses the assumption in terms of the training samples of a single class. Alternately, it can be expressed in terms of all the training samples so that

$$x_{test} = X\alpha + \eta \quad (4)$$

where $X = [x_{1,1} | ... | x_{n,1} | ... | x_{c,1} | ... | x_{c,n_c} | ...x_{C,1} | ... | x_{C,n_C}]$

and $\alpha = [\alpha_{1,1}...\alpha_{1,n_1}...\alpha_{c,1}...\alpha_{c,n_c}...\alpha_{C,1}...\alpha_{C,n_C}]^T$.

According to the SRC assumption, only those α's corresponding to the correct class will be non-zeroes. The rest are all zeroes. In other words, α will be sparse. Therefore, one needs to solve the inverse problem (4) with sparsity constraints on the solution. This is formulated as:

$$\min_{\alpha} \|x_{test} - X\alpha\|_2^2 + \lambda\|x\|_1 \quad (5)$$

Once (10) is solved, the representative sample for every class is computed: $x_{rep}(c) = \sum_{j=1}^{n_c} \alpha_{c,j} v_{c,j}$. It is assumed that the test sample will look very similar to the representative sample of the correct class and will look very similar, hence the residual $\varepsilon(c) = \|x_{test} - x_{rep}(c)\|_2^2$, will be the least for the correct class. Therefore once the residual for every class is obtained, the test sample is assigned to the class having the minimum residual.

III. Proposed Approach

A. Deep Dictionary Learning

In this work we propose the concept of deep dictionary learning. Instead of learning a single level of dictionary, we learn the dictionaries in layers. The representation is expressed as,

$$X = D_1D_2...D_NZ \quad (6)$$

Here X is the training data, $D_1 \ldots D_N$ are different layers of dictionaries and Z is the representation at the final layer. For a sparse Z, the optimization problem that needs to be solved is,

$$\min_{D_1,...,D_N,Z} \|X - D_1D_2...D_NZ\|_F^2 + \lambda\|Z\|_{1/0} \quad (7)$$

Here we abuse the notations a bit, '1/0' mean that it can be either l_1-norm or l_0-norm.

The problem is difficult to solve. Following studies in deep learning, we propose solving it in a greedy fashion [19] – one layer at a time. For the first layer, we substitute, $Z_1 = D_2...D_NZ$, in (6), leading to,

$$X = D_1Z_1 \quad (8)$$

Z_1 is not sparse, hence this problem (9) can be solved by matrix factorization, i.e.,

$$\min_{D_1,Z_1} \|X - D_1Z_1\|_F^2 \quad (9)$$

There are a plethora of techniques to solve (9). In this work, we solve it using simple alternating minimization.

$$Z_1 \leftarrow \min_{Z_1} \|X - D_1Z_1\|_F^2 \quad (10a)$$

$$D_1 \leftarrow \min_{D_1} \|X - D_1Z_1\|_F^2 \quad (10b)$$

Both have a closed form solution. It starts with initializing D_1. In every iteration, the features are updated assuming a fixed D_1 (10a); and the dictionary is updated assuming a fixed Z_1. Iterations continue till the solution reaches some local minima.

Once the first level is solved, we substitute $X = D_1Z_1 = D_1D_2Z_2, Z_2 = D_3...D_NZ$. Hence we need to solve,

$$\min_{D_2,Z_2} \|Z_1 - D_2Z_2\|_F^2 \quad (11)$$

As before, this too can be solved using alternating minimization.

Such substitution is carried out till the final layer, where we get,

$$Z_{N-1} = D_NZ \quad (12)$$

The feature is sparse at the final level. The problem is therefore posed as,

$$\min_{D_N,Z} \|Z_{N-1} - D_N Z\|_F^2 + \lambda \|Z\|_{1/0} \quad (13)$$

Alternating minimization of (13) leads to,

$$Z \leftarrow \min_{Z} \|Z_{N-1} - D_N Z\|_F^2 + \lambda \|Z\|_{1/0} \quad (14a)$$

$$D_N \leftarrow \min_{D_N} \|X - D_N Z\|_F^2 \quad (14b)$$

The second problem has a closed form solution. The first problem for the case of l_1-norm is easily solved via iterative soft-thresholding (IST) [20]. For l_0-norm it can be solved using iterative hard thresholding.

IST

$$B = Z_{k-1} + \sigma D_N^T \left(Z_{N-1} - D_N Z\right)$$
$$Z_k = signum(B) \cdot \max(0, |B| - \lambda\sigma)$$

IHT

$$B = Z_{k-1} + \sigma D_N^T \left(Z_{N-1} - D_N Z\right)$$
$$Z_k = Hard(B, \lambda\sigma)$$
$$\text{where } Hard(s,\tau) = \begin{cases} s^{(i)} \text{ when } s^{(i)} > \lambda\sigma \\ 0 \text{ when } s^{(i)} \le \lambda\sigma \end{cases}$$

In deep learning, the layers are learnt in a greedy fashion – one by one. This is called the pre-training phase. After layer-wise learning, the entire network is learnt in one go – this is the fine-tuning phase. The fine-tuning allows feedback into previous layers. However, such a fine tuning is not mandatory as has been shown by FaceNet [22]; they learn the layers greedily and use them for feature extraction. In this work we follow a similar approach. We just learn the deep dictionaries in a greedy fashion. This is equivalent to a feed-forward neural network; there is no back-propagation.

Our other reason to not go for fine-tuning is because single layer dictionary learning enjoys certain theoretical advantages [23-26]. There are local convergence guarantees. These cannot be easily extended to deep dictionary learning when all the dictionaries are learnt in one go. But when we learn the dictionaries are learnt in a greedy fashion, one layer at a time, each layer is bound to converge.

The schematic diagram of deep dictionary learning is given in the following figure.

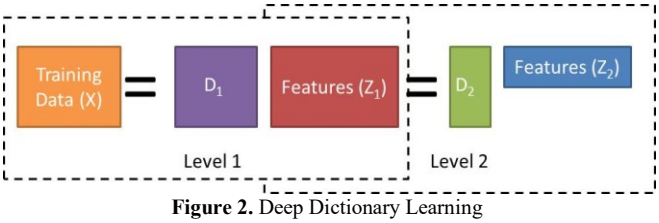

Figure 2. Deep Dictionary Learning

B. Class-wise Learning

The Sparse Representation based Classification (SRC) approach assumes that the training samples from a class form a linear basis for representing the test samples of the same class. Assuming that the test sample belongs to class k, this is represented as:

$$x_{test} = X^{(c)} \alpha^{(c)} \quad (15)$$

Here $X^{(c)} = [x_{c,1} | ... | x_{c,n_c}]$, from (3).

In this work, we learn a basis / dictionary for representing each class. It is assumed that the test sample should be sparsely represented by the dictionary for the correct class only; for all other classes the coefficients should be zero or negligibly small (we have not enforced this criterion during dictionary learning, but we hope that this is being satisfied). According to our model, the test sample is represented as:

$$x_{test} = \left[D^{(1)} | ... | D^{(c)} | ... | D^{(C)}\right] \alpha \quad (16)$$

where α is sparse.

Such a migration from using raw samples to learned dictionaries for representing test data has been envisaged by the text retrieval community. Salton introduced the Vector Space Model for text mining in the 70s [27]; it used the term-document matrix for representing a new document. In late 90's, Lee and Seung [28] proposed learning a basis for the term-document matrix using non-negative matrix factorization for document representation.

However, in our case, the basis is not a single level of dictionary, instead it is a multi-level deep dictionary, i.e.

$$D^{(c)} = D_1^{(1)} D_2^{(c)} ... D_N^{(C)} \quad (17)$$

The classification proceeds in a fashion similar to SRC. The sparse coefficient vector α is found via l_1-norm minimization.

$$\min_{\alpha} \left\| x_{test} - \left[D^{(1)} | ... | D^{(c)} | ... | D^{(C)}\right] \alpha \right\|_2^2 + \mu \|\alpha\|_{1/0} \quad (18)$$

Once α is obtained, the residual error for each class is obtained as:

$$error(c) = \left\| v_{test} - D^{(c)} \alpha^{(c)} \right\|_2, \forall c = 1...C \quad (19)$$

It must be noted that the proposed technique is not the same as [29] – they propose a feature extraction scheme where a separate classifier is required; our method encompasses both feature extraction and classification; it is a complete solution. In [29] the authors propose a classical approach for multi-stage vector quantization – well known in digital voice processing literature. They create multiple stages of visual codebooks by image patches. These codebooks are not class specific. The error in each encoding stage, is encoded by the codebook from the subsequent stage. This technique is fundamentally different from ours. We learn multiple stages of class specific dictionaries – this is not the same as vector quantization. Besides our dictionaries are class specific, theirs are not. We encode the coefficient / representation by dictionaries from the subsequent stage – not the quantization error.

IV. Experimental Evaluation

A. Benchmarking Experiments

Figure 2. Samples from MNIST dataset

We have carried the experiments on benchmark datasets. The first one is the MNIST (Figure 2). The MNIST digit classification task is composed of 28x28 images of the 10 handwritten digits. There are 60,000 training images with 10,000 test images in this benchmark. No preprocessing has been done on this dataset.

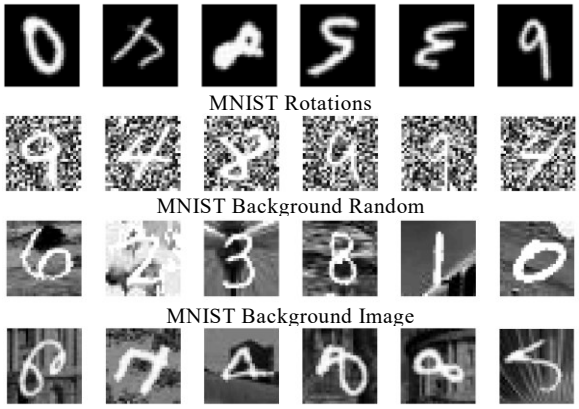

Figure 3. Samples from MNIST Variations (www.iro.umontreal.ca/~lisa/twiki/bin/view.cgi/Public/MnistVariations)

Experiments were carried out on the MNIST variations datasets (Figure 3); these have been routinely used for benchmarking deep-learning algorithms. The datasets are MNIST basic, MNIST background random (bg-rand), MNIST background image (bg-image), MNIST rotated (rot) and MNIST background image with rotations (bg-image-rot). All these datasets are widely used for benchmarking deep learning tools. Both the rotations and random noise background dataset has 12000 training samples and 50000 test samples. Usually the training is done using 10,000 samples and the remaining 2,000 are used for validation.

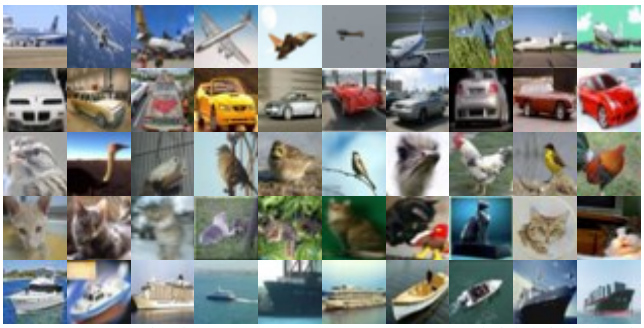

Figure 4. Samples from CIFAR-10 (www.cs.toronto.edu/~kriz/cifar.html)

The CIFAR-10 dataset is composed of 10 classes of natural images with 50,000 training examples in total, 5,000 per class. Each image is an RGB image of size 32x32 taken from the tiny images dataset and labeled by hand. These images need to be preprocessed. We follow the standard preprocessing technique – the RGB is converted to YUV and the Y channel is used. Before putting it for training, mean subtraction and global contrast normalization is done.

Figure 5. Samples from SVHN (ufldl.stanford.edu/housenumbers/)

The Street View House Numbers (SVHN) dataset is composed of 604,388 images (using both the difficult training set and simpler extra set) and 26,032 test images. The goal of this task is to classify the digit in the center of each cropped 32x32 color image. This is a difficult real world problem since multiple digits may be visible within each image. We preprocessed these samples in the same way as CIFAR.

Our proposed method is non-convex and hence the results are dependent on the initialization of the dictionary. In order to make the results repeatable we create the dictionary by choosing columns after Gram-Schmidt orthogonalization of the input feature. This may not yield the best classification results but makes the results repeatable.

In the experiments, the final level learnt dictionary for sparse features, the rest of the levels generated dictionaries

for dense features. We use two variants of the proposed deep dictionary learning. In the first one, we use l_1-minimization in the final stage. In the second case, we use l_0-minimization All the datasets use 3 levels with 100-80-10 atoms.

The benchmarking was carried out with state-of-the-art deep learning tools – Deep Belief Network (DBN) [30], Stacked Denoising Autoencoder (SDAE) [31], Label Consistend KSVD (LC-KSVD) [18] and Convolutional Neural Network (CNN). The CNN, SDAE and DBN were of 3 layers; they were optimized for the said datasets. For LC-KSVD, we report the best results.

TABLE I. CLASSIFICATION ACCURACY

Dataset	Proposed - l_1	Proposed - l_0	DBN	SDAE	LC-KSVD	CNN
MNIST	**99.48**	**99.12**	98.78	98.72	87.05	99.06
MNIST basic	97.15	**97.27**	96.89	96.54	83.59	**98.56**
MNIST rot	89.38	**90.6**	**89.70**	**89.70**	80.42	89.45
MNIST bg-image	79.15	**84.40**	83.69	77.00	70.92	**88.89**
MNIST bg-rand	90.83	**93.33**	**93.27**	88.72	80.04	93.23
MNIST bg-image-rot	50.17	**52.83**	52.61	48.07	42.31	**57.97**
CIFAR-10	**85.55**	**83.60**	78.90	74.30	60.32	83.40
SVHN	**94.42**	93.11	92.60	89.70	80.64	**94.97**

In all cases, one of our proposed method is among the top-2 results. In fact when training data is limited, our method yields significantly better results than generic deep learning techniques like DBN and SDAE and supervised shallow dictionary learning – LC-KSVD. Our method gives result at par with CNN. In general, between the two of our methods, the l_0-norm yields better results when the number of training samples are considerably large, but the l_1-norm excels for limited number of samples.

One may wonder, if collapsing all the dictionaries into one will have the same effect as having the set of cascaded dictionaries. The answer would be in the negative. This is because the dictionary learning process is bi-linear and non-convex; hence it is not possible to collapse all the dictionaries into one. We have done a simple experiment to verify this empirically. On all the databases we fixed the dictionary atoms to 10 – the same as the final level of our three level dictionary learning; and to 80 – the same as the first level of our three level dictionary learning. The results are shown in Table 2. The results are markedly different.

TABLE II. SINGLE LEVEL VS MULTI-LEVEL DICTIONARY LEARNING

Dataset	Multi-level	Single level	Single level
MNIST	99.48	10.76	11.43
MNIST basic	97.15	9.94	12.23
MNIST rot	89.38	13.40	9.80
MNIST bg-image	79.15	8.13	8.39
MNIST bg-rand	90.83	11.49	9.16
MNIST bg-image-rot	50.17	8.18	11.08

The runtimes for the proposed method is compared with two deep learning techniques – deep belief network and stacked autoencoder. Our method is a complete solution for classification. Therefore we compare it with CNN, SDAE and DBN with soft-max classifier. We only show the results on the MNIST and MNIST basic. The size of other MNIST variations take almost the same time as the basic dataset. The CIFAR-10 dataset is of the same size as MNIST; so the run-times are almost the same. The machine used is Intel (R) Core(TM) i5 running at 3 GHz; 8 GB RAM, Windows 10 (64 bit) running Matlab 2014a.

TABLE III. RUN-TIME IN MINUTES

Dataset	SDAE	DBN	CNN	Proposed - l_1	Proposed – l_0
MNIST	2010	505	1106	315	330
basic	1422	100	400	80	90

B. BCI Experiments

Some recent studies showed that deep belief network and its variants yield very good results for EEG classification problems [32-35]. In this work we focus on the Brain Computer Interface (BCI) classification problem addressed in [35]. We follow the experimental protocol outlined therein.

We have used two open source EEG datasets to conduct the experiments. Dataset 1 comes from dataset III in 2003 BCIC II; it contains a total of 280 groups of left and right hand Motor Imagery (MI) EEG data. Dataset 2 comes from dataset Iva in 2005 BCIC III; it includes the 'aa', 'al' and 'aw' three subsets, and each subset contains a total of 280 groups of right hand and foot MI EEG data. Dataset 3 is from dataset III in 2005 BCIC III; it contains 360 groups of 4 classes (left hand, righthand, foot, tongue). These are the same datasets used in [35].

The experimental protocol followed in this work stems from [35]. Dataset 1 and dataset 2 each have 280 trails. We set different numbers of training samples, which are 80, 120, 160, 200, 240; the remaining samples are test samples. Dataset 3 has 297 trails. We set different numbers of training

samples for dataset 3, which are 140, 160, 180; the remaining samples are test samples. For a fixed number of training samples, 20 independent runs were conducted; for each run the training samples were selected uniformly at random. The mean and standard deviation of the classification accuracy is reported: Table 4 (Dataset 1); Table 5 (Dataset 2, 'aa'), Table 6 (Dataset 2, 'al'), Table 7 (Dataset 2 'aw') and Table 8 (Dataset 3).

The prior work [35] showed that convolutional deep belief network (CDBN) yields superior results compared to traditional feature extraction techniques used in BCI. They compared it with Bandpower [36], MVAAR (multivariate adaptive autoregressive) [37] and CSP (common spatial pattern) [38]. Therefore in this work we only compare our proposed class-wise deep dictionary learning algorithm with CDBN [35]; the architecture used for CDBN is detailed in their paper.

As a pre-processing step, the time domain signal is first converted to frequency domain. This is because the data we handle is from an asynchronous BCI. We choose the signal in 8-30Hz frequency band (mentioned in [35]). As the number of channels of EEG data is large (e.g., 118 channels of dataset 2), PCA is used to whiten the data and reduce dimensionality before feeding it to the learning algorithm.

TABLE IV. DATASET 1: CLASSIFICATION ACCURACY

#training samples	CDBN [35] (%)	Proposed-l_1 (%)	Proposed-l_0 (%)
80	83.6, ±1.8	86.1, ±2.8	86.4, ±2.8
120	85.9, ±1.8	88.2, ±3.1	88.7, ±3.0
160	86.0, ±2.1	89.0, ±3.4	90.0, ±3.1
200	86.1, ±3.3	90.9, ±3.8	91.6, ±3.8
240	88.3, ±5.7	92.1, ±4.2	92.6, ±4.0

TABLE V. DATASET 2 'AA': CLASSIFICATION ACCURACY

#training samples	CDBN [35] (%)	Proposed-l_1 (%)	Proposed-l_0 (%)
80	88.6, ±2.8	90.1, ±2.9	90.5, ±2.8
120	85.9, ±2.6	91.2, ±3.0	91.7, ±3.0
160	85.4, ±2.6	91.3, ±3.0	91.9, ±3.3
200	85.9, ±2.8	91.9, ±3.2	92.6, ±3.5
240	88.3, ±3.7	92.7, ±3.6	93.0, ±4.0

TABLE VI. DATASET 2 'AL': CLASSIFICATION ACCURACY

#training samples	CDBN [35] (%)	Proposed-l_1 (%)	Proposed-l_0 (%)
80	95.1, ±0.8	98.2, ±0.9	99.5, ±0.8
120	96.2, ±0.6	100, ±0.0	100, ±0.0
160	100, ±0.0	100, ±0.0	100, ±0.0
200	100, ±0.0	100, ±0.0	100, ±0.0
240	100, ±0.0	100, ±0.0	100, ±0.0

TABLE VII. DATASET 2 'AW': CLASSIFICATION ACCURACY

#training samples	CDBN [35] (%)	Proposed-l_1 (%)	Proposed-l_0 (%)
80	94.6, ±1.2	97.2, ±1.9	98.9, ±0.5
120	96.2, ±0.5	98.8, ±0.8	99.5, ±0.2
160	95.3, ±1.0	99.6, ±0.5	100, ±0.0
200	100, ±0.0	100, ±0.0	100, ±0.0
240	100, ±0.0	100, ±0.0	100, ±0.0

TABLE VIII. DATASET 3: CLASSIFICATION ACCURACY

#training samples	CDBN [35] (%)	Proposed-l_1 (%)	Proposed-l_0 (%)
140	82.0, ±1.9	86.2, ±2.1	86.7, ±2.0
160	82.4, ±1.5	87.6, ±1.4	88.0, ±1.3
180	87.3, ±1.7	93.9, ±1.8	93.6, ±18

Datasets 1 and 2, correspond to two class problems; dataset 3 is a slightly more challenging 4 class problem. For ALL the problems our method performs better than the convolutional deep belief network. The margin between CDBN and our proposed method is considerably large. Between the two of our techniques (l_1-norm and l_0-norm), there is not much of a difference; the l_0-norm yields marginally better results compared to the l_1-norm.

V. CONCLUSION

In this work we propose a new feature extraction and classification framework – class-wise deep dictionary learning. The idea is to learn multiple levels of dictionaries from the features learnt from previous levels; the first level of dictionary learning uses the raw data as input. This is the first work that proposes deep dictionary learning.

The classification is based on the sparse representation approach based on the assumption that the class-wise dictionaries will represent the test data in the sparse fashion. This is a relatively simple technique; we compared it with all major deep learning tools like DBN, SDAE, CNN and a shallow supervised dictionary learning (LC-KSVD) technique. On the benchmark problems our method yields the best results on an aggregate; especially in the challenging scenario when the training data is limited. For a real problem on BCI, our method has been compared with the state-of-the-art convolutional DBN; we always perform considerably better. We have shown that our technique is much faster than other deep learning techniques.

The dictionary learning technique we employ at each level is naïve – it produces either a dense or a sparse set of features. There has been significant amount of work on discriminative dictionary learning techniques. We believe that the results can be improved even further if such techniques can be incorporated in our framework.

The proposed technique is fraught with the same limitations as SRC. It would be good for problems on multi-class classification such as identification in biometrics. However it would not succeed very well for problems such as verification and detection.

ACKNOWLEDGEMENT

This work was made possible by NPRP grant 7 - 684 - 1 - 127 from the Qatar National Research Fund (a member of Qatar Foundation). The statements made herein are solely the responsibility of the authors.

REFERENCES

[1] M. Aharon, M. Elad and A. Bruckstein, "K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation", IEEE Transactions on Signal Processing, Vol. 54 (11), pp. 4311-4322, 2006.

[2] M. Yang, L. Zhang, J. Yang, and D. Zhang. metaface learning for sparse representation based face recognition. ICIP, 2010

[3] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry and Y. Ma, "Robust Face Recognition via Sparse Representation". IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 31, pp. 210-227, 2009.

[4] K. Engan, S. Aase, and J. Hakon-Husoy, "Method of optimal directions for frame design," IEEE International Conference on Acoustics, Speech, and Signal Processing, 1999.

[5] B. Olshausen and D. Field, "Sparse coding with an overcomplete basis set: a strategy employed by V1?" Vision Research, Vol. 37 (23), pp. 3311-3325, 1997.

[6] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. "Discriminative learned dictionaries for local image analysis". IEEE Conference of Computer Vision and Pattern Recognition, 2008.

[7] L. Yang, R. Jin, R. Sukthankar, and F. Jurie. "Unifying discriminative visual codebook generation with classifier training for object category recognition". IEEE Conference of Computer Vision and Pattern Recognition, 2008.

[8] W. Jin, L. Wang, X. Zeng, Z. Liu and R. Fu, "Classification of clouds in satellite imagery using over-complete dictionary via sparse representation", Pattern Recognition Letters, Vol. 49 (1), pp. 193-200, 2014.

[9] Y. Boureau, F. Bach, Y. LeCun, and J. Ponce. "Learning mid-level features for recognition". IEEE Conference of Computer Vision and Pattern Recognition, 2010.

[10] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. "Supervised dictionary learning". Advances in Neural Information Processing Systems, 2009.

[11] J. Mairal, M. Leordeanu, F. Bach, M. Hebert, and J. Ponce. "Discriminative sparse image models for class-specific edge detection and image interpretation". European Conference on Computer Vision, 2008.

[12] K. Huang and S. Aviyente. "Sparse representation for signal classification". Advances in Neural Information Processing Systems, 2007.

[13] D. Pham and S. Venkatesh. "Joint learning and dictionary construction for pattern recognition". IEEE Conference of Computer Vision and Pattern Recognition, 2008.

[14] Q. Zhang and B. Li. "Discriminative k-svd for dictionary learning in face recognition". IEEE Conference of Computer Vision and Pattern Recognition, 2010.

[15] J. Yang, K. Yu, and T. Huang. "Supervised translation-invariant sparse coding". IEEE Conference of Computer Vision and Pattern Recognition, 2010.

[16] J. Mairal, M. Leordeanu, F. Bach, M. Hebert, and J. Ponce. "Discriminative sparse image models for class-specific edge detection and image interpretation". European Conference on Computer Vision, 2008.

[17] Q. Zhang and B. Li, "Discriminative K-SVD for dictionary learning in face recognition". IEEE Conference of Computer Vision and Pattern Recognition, 2010.

[18] Z. Jiang, Z. Lin and L. S. Davis, "Learning A Discriminative Dictionary for Sparse Coding via Label Consistent K-SVD", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 35, pp. 2651-2664, 2013

[19] Y. Bengio, "Learning deep architectures for AI", Foundations and Trends in Machine Learning, 2 (1),1-127. 2009

[20] I. Daubechies and M. Defrise, and C. De Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint", Communications on Pure and Applied Mathematics Vol. 4 (57), 1413–1457, 2004.

[21] Thomas Blumensath and Mike E. Davies, "Iterative Thresholding for Sparse Approximations", Journal of Fourier Analysis Applications, Vol. 14 (5), 629-654, 2008.

[22] Florian Schroff, Dmitry Kalenichenko, James Philbin, "FaceNet: A Unified Embedding for Face Recognition and Clustering", arXiv:1503.03832

[23] P. Jain, P. Netrapalli and S. Sanghavi, "Low-rank Matrix Completion using Alternating Minimization", Symposium on Theory Of Computing, 2013.

[24] A. Agarwal, A. Anandkumar, P. Jain and P. Netrapalli, "Learning Sparsely Used Overcomplete Dictionaries via Alternating Minimization", International Conference On Learning Theory, 2014.

[25] D. A. Spielman, H. Wang and J. Wright, "Exact Recovery of Sparsely-Used Dictionaries", International Conference On Learning Theory, 2012

[26] S. Arora, A. Bhaskara, R. Ge and T. Ma, "More Algorithms for Provable Dictionary Learning", arXiv:1401.0579v1

[27] G. Salton, A. Wong, and C. S. Yang, "A Vector Space Model for Automatic Indexing," Communications of the ACM, Vol. 18, pp. 613-620, 1975.

[28] D. D. Lee and H. S. Seung,. "Learning the parts of objects by non-negative matrix factorization". Nature 401 (6755), pp. 788-791, 1999.

[29] S. Bai, X. Wang, C. Yao and X. Bai, "Multiple stage residual model for accurate image classification", ACCV, 2014.

[30] G. Hinton, "Deep belief networks". Scholarpedia 4 (5): 5947. doi:10.4249/scholarpedia.5947, 2009.

[31] 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", Journal of Machine Learning Research, Vol. 11, pp. 3371-3408, 2011.

[32] J. T. Turner, A. Page, T. Mohsenin and T. Oates, "Deep Belief Networks used on High Resolution Multichannel Electroencephalography Data for Seizure Detection", AAAI, 2014.

[33] A. M. Al-kaysi1, A. Al-Ani1 and T. W. Boonstra, "A Multichannel Deep Belief Network for the Classification of EEG Data", ICONIP, 2015.

[34] Z. Wang, S. Lyu, G. Schalk and Q. Ji "Deep Feature Learning Using Target Priors with Applications in ECoG Signal Decoding for BC", IJCAI, 2013.

[35] Y. Ren and Y. Wu, "Convolutional Deep Belief Networks for Feature Extraction of EEG Signal", IJCNN, 2014.

[36] N. Brodu, F. Lotte, A. Lecuyer, "Comparative study of band-power extraction techniques for motor imagery classification," 2011 IEEE Symposium on Computational Intelligence, Cognitive Algorithms, Mind and Brain (CCMB), pp.1-6.

[37] C.W. Anderson, E.A. Stolz, S. Shamsunder, “Multivariate autoregressive models for classification of spontaneous electroencephalographic signals during mental tasks”, IEEE Trans. Biomed.Eng., vol. 45, no. 3, pp.277-286 ,1998.

[38] J. Müller-Gerking, G. Pfurtcheller, H. Flyvberg, “Designing optimal spatial filters for single-trial EEG classification in a movement task”, Clinical Neurophysiology, vol.110, no.5, pp.787-789,1991.