

University of Louisville

## ThinkIR: The University of Louisville's Institutional Repository

---

Electronic Theses and Dissertations

---

5-2016

### Sparse feature learning for image analysis in segmentation, classification, and disease diagnosis.

Ehsan Hosseini-Asl

Follow this and additional works at: <https://ir.library.louisville.edu/etd>



Part of the [Theory and Algorithms Commons](#)

---

#### Recommended Citation

Hosseini-Asl, Ehsan, "Sparse feature learning for image analysis in segmentation, classification, and disease diagnosis." (2016). *Electronic Theses and Dissertations*. Paper 2456.  
<https://doi.org/10.18297/etd/2456>

This Doctoral Dissertation is brought to you for free and open access by ThinkIR: The University of Louisville's Institutional Repository. It has been accepted for inclusion in Electronic Theses and Dissertations by an authorized administrator of ThinkIR: The University of Louisville's Institutional Repository. This title appears here courtesy of the author, who has retained all other copyrights. For more information, please contact [thinkir@louisville.edu](mailto:thinkir@louisville.edu).

SPARSE FEATURE LEARNING FOR IMAGE ANALYSIS IN  
SEGMENTATION, CLASSIFICATION, AND DISEASE DIAGNOSIS

By

Ehsan Hosseini-Asl  
M.Sc., Petroleum University of Technology, 2008

A Dissertation  
Submitted to the Faculty of the  
J. B. Speed School of Engineering of the University of Louisville  
in Partial Fulfillment of the Requirements  
for the Degree of

Doctor of Philosophy in Electrical Engineering

Department of Electrical and Computer Engineering  
University of Louisville  
Louisville, Kentucky

May 2016



SPARSE FEATURE LEARNING FOR IMAGE ANALYSIS IN  
SEGMENTATION, CLASSIFICATION, AND DISEASE DIAGNOSIS

By

Ehsan Hosseini-Asl  
M.Sc., Petroleum University of Technology, 2008

A Dissertation Approved On

January 28, 2016

by the following Dissertation Committee:

---

Jacek M. Zurada, Dissertation Director

---

Ayman El-Baz, Co-Advisor

---

Tamer Inanc

---

Amir Amini

---

Olfa Nasraoui

## ACKNOWLEDGEMENTS

I would first like to thank my advisor, Dr. Jacek M. Zurada, Director of the Computational Intelligence Laboratory at the University of Louisville, for his guidance, encouragement, and patience. I am also deeply grateful to my co-advisor, Dr. Ayman El-Baz for his broad and unlimited support during my research. I would also like to thank Dr. Olfa Nasraoui, Dr. Tamer Inanc, and Dr. Amir Amini for agreeing to serve on the dissertation committee. I am grateful for their encouragement and advice.

# ABSTRACT

## SPARSE FEATURE LEARNING FOR IMAGE ANALYSIS IN SEGMENTATION, CLASSIFICATION, AND DISEASE DIAGNOSIS

Ehsan Hosseini-Asl

January 28, 2016

The success of machine learning algorithms generally depends on intermediate data representation, called features that disentangle the hidden factors of variation in data. Moreover, machine learning models are required to be generalized, in order to reduce the specificity or bias toward the training dataset. Unsupervised feature learning is useful in taking advantage of large amount of unlabeled data, which is available to capture these variations. However, learned features are required to capture variational patterns in data space. In this dissertation, unsupervised feature learning with sparsity is investigated for sparse and local feature extraction with application to lung segmentation, interpretable deep models, and Alzheimer's disease classification. Nonnegative Matrix Factorization, Autoencoder and 3D Convolutional Autoencoder are used as architectures or models for unsupervised feature learning. They are investigated along with nonnegativity, sparsity and part-based representation constraints for generalized and transferable feature extraction.

# TABLE OF CONTENTS

	Page
ACKNOWLEDGEMENTS	iii
ABSTRACT	iv
LIST OF TABLES	viii
LIST OF FIGURES	ix
CHAPTER	
I INTRODUCTION	1
II UNSUPERVISED FEATURE LEARNING METHODS	2
A Matrix decomposition . . . . .	2
1 Sparse coding . . . . .	3
2 Nonnegative Matrix Factorization (NMF) . . . . .	4
B Deep learning . . . . .	9
1 Autoencoders . . . . .	10
2 Deep networks based on stacked autoencoder . . . . .	12
3 Convolutional Autoencoder and Networks (CAE/CNN) . . . . .	13
C Unsupervised and Transfer Learning . . . . .	15
III MOTIVATIONS AND RESEARCH GOALS	17
A Image segmentation . . . . .	17
B Image classification . . . . .	18
C Disease diagnosis . . . . .	19
IV LUNG SEGMENTATION BASED ON UNSUPERVISED FEATURE LEARNING	21
A NMF-based segmentation . . . . .	22
1 Preprocessing . . . . .	23

2	NMF-based visual appearance modeling . . . . .	23
3	Segmentation . . . . .	25
B	INMF-based automatic segmentation of pathological lungs . . . . .	26
1	INMF-based visual appearance modeling . . . . .	27
2	Segmentation . . . . .	27
C	ICNMF-based 3D lung segmentation . . . . .	28
1	Incremental Constrained NMF (ICNMF) . . . . .	30
2	ICNMF-based visual appearance modeling . . . . .	32
3	Segmentation . . . . .	34
D	Experiments with synthetic and real data . . . . .	36
E	Conclusions . . . . .	42
V	PART-BASED REPRESENTATION FOR DEEP LEARNING . . . . .	44
A	Nonnegativity Constrained Autoencoder (NCAE) . . . . .	46
B	Deep learning using NCAE . . . . .	47
C	Feature extraction by NCAE . . . . .	49
1	Unsupervised feature learning . . . . .	49
2	Learning part-based representation of images . . . . .	49
3	Semantic feature discovery from text data . . . . .	60
4	Supervised learning . . . . .	64
D	Structured Sparse Convolutional Autoencoder (SSCAE) . . . . .	68
E	Feature extraction by SSCAE . . . . .	72
1	Minimizing dead filters . . . . .	73
2	Improving learning of reconstruction . . . . .	75
F	Conclusion . . . . .	76
VI	DEEP NETWORK FOR ALZHEIMER’S DISEASE DETECTION . . . . .	79
A	Alzheimer’s Disease (AD) . . . . .	79
B	AD diagnosis . . . . .	81
C	MRI imaging for AD diagnosis . . . . .	82
D	Proposed model . . . . .	84

1	3D Convolutional autoencoder . . . . .	84
2	Deeply Supervised Adaptive 3D-CNN (DSA-3D-CNN) . . . . .	86
E	Experiment . . . . .	87
1	Generic and task-specific feature evaluation: . . . . .	89
2	Classification performance evaluation . . . . .	96
F	Conclusion . . . . .	97
VII CONCLUSION AND FUTURE WORK		99
A	Lung Segmentation . . . . .	99
B	Image Classification . . . . .	100
C	Disease Diagnosis . . . . .	101
D	Future work . . . . .	102
REFERENCES		103
APPENDICES		137
A	LIST OF ACRONYMS	137
CURRICULUM VITAE		140

## LIST OF TABLES

TABLE		Page
1	Segmentation accuracy for different weights in Eq.(40). . . . .	36
2	Comparative segmentation accuracy (DSC) of Algorithm 5 using the $(3 \times 3 \times 3)$ context w.r.t. five other algorithms on the <i>in vivo</i> data sets. . . . .	39
3	Accuracy of the ICNMF-based Algorithm 5 with the $(3 \times 3 \times 3)$ contexts w.r.t. the INMF-based [1] and NMF-based [2] algorithms on synthetic and real ( <i>in vivo</i> ) data sets: $\text{mean}_{\pm\text{std}} [p\text{-value}]$ . . . . .	39
4	Overlaps with the true left (LL) and right lungs (RL) for Algorithm 5 (A5) w.r.t. a human expert and the most (rank 1), median (rank 8), and least accurate (rank 15) LOLA11 segmentation of the 55 data sets : the mean, standard deviation (std), minimum (min), 25%-quartile (Q1), median (med), 75%-quartile (Q3), and maximum (max) overlaps. . . . .	40
5	Performance of supervised learning methods on MNIST dataset. . . . .	67
6	Performance of supervised learning methods on NORB dataset. . . . .	67
7	Performance of supervised learning methods on Reuters-21578 Dataset. . . . .	67
8	Demographic information of 210 studied subjects from the ADNI dataset. . . . .	89
9	Task-Specific Classification Models Performance for a selected fold of <i>crossvalidation</i> for the proposed model. . . . .	94
10	Classification performance evaluation of the proposed model [mean(std)%] . . . . .	94
11	Performance comparison (ACC%) of the competing methods. . . . .	95

## LIST OF FIGURES

FIGURE	Page
1	Unsupervised feature learning based on matrix factorization. . . . . 3
2	Schematic diagram of a three-layer autoencoder. . . . . 11
3	Architecture of (a) autoencoder and (b) stacked autoencoder deep network [3]. . . . . 13
4	Construction of convolutional neural network by (a) locally connected neuron, and (b) convolution over feature maps [4, 5]. . . . . 14
5	Architecture of LeNet-5 model for digit recognition [4]. . . . . 15
6	Original 3D image slices (a) before and (b) after removing their background by 3D region growing from a seed at the 3D image corner. . . . . 23
7	The developed framework for 3D lung segmentation from CT images using NMF. . . . . 24
8	(a) The developed framework for automatic lung segmentation from CT images, (b) 2-step clustering for pathological lung case, where three clusters are detected by INMF method, and two clusters (lung and chest) are extracted [1]. . . . . 29
9	The developed framework for 3D lung segmentation based on ICNMF [6]. 33
10	Signal distributions for segmented voxels in the original 27-dimensional space (a) and the $r$ -dimensional spaces reduced with the NMF (b) and ICNMF (c) visualized using the t-SNE projection [7]. The better ICNMF performance is exemplified by signal distributions and segmentation results for pathologies on the lung-chest border [6]. . . . . 35
11	Original 2D CT slices (a) and 3D lungs segmented using $3 \times 3 \times 3$ (b), $7 \times 7 \times 3$ (c), $11 \times 11 \times 3$ (d), and $15 \times 15 \times 3$ (e) neighborhood size, and projected onto the axial (A), sagittal (S), and coronal (C) 2D planes for visualization. . . . . 37

12	Lungs segmented with Algorithm 5 using the $3 \times 3 \times 3$ neighborhood on pulmonary CT in the cases of airspace consolidation (a); tree-in-bud and micro-nodules (b); usual nodules (c); cancer (d); ground-glass opacity and juxtapleural nodules (e); honeycomb (f); diffuse consolidation (g), and cavity (h) [6]. . . . .	38
13	3D visualization of segmented lungs from selected LOLA11 subjects (“s” indicates the scan). . . . .	41
14	3D lungs segmented with Algorithm 5 using the $(3 \times 3 \times 3)$ context: the CT slice (“s” and “f” indicate the scan and slice numbers, respectively) vs. the corresponding lung region map (white) in the LOLA11 dataset [6].	43
15	Schematic diagram of a deep network . . . . .	48
16	196 Receptive fields learned from MNIST digit data set using (a) SAE, (b) NNSAE, (c) NCAE*, and (d) NMF. Black pixels indicate negative, and white pixels indicate positive weights. Black nodes in (b) indicate neurons with zero weights. . . . .	50
17	Encoding weight $\mathbf{W}^{(1)}$ distribution histogram of (a) SAE, (b) NNSAE, (c) NCAE*, and (d) NMF. . . . .	51
18	196 decoding filters ( $\mathbf{W}^{(2)}$ ) with weight histogram learned from MNIST digit data set using (a) SAE and (b) NCAE*. Black pixels indicate negative, and white pixels indicate positive weights. Black nodes in (b) indicate neurons with zero weights. . . . .	52
19	Decoding weight $\mathbf{W}^{(2)}$ distribution histogram of (a) SAE, (b) NNSAE, (c) NCAE*. . . . .	52
20	Histogram of the sparseness criterion (Eq.(11)) measured on 196 receptive fields. . . . .	53
21	Histogram of the sparseness criterion (Eq.(11)) measured on 196 decoding filters. . . . .	54
22	Reconstruction comparison of the MNIST digits data set by 196 receptive fields, using SAE , NNSAE [8], NCAE*, and NMF. . . . .	55
23	Comparison of reconstruction error computed by Eq.(25). . . . .	56

24	Sparsity of hidden units measured by the KL divergence in Eq.(27) for the MNIST dataset for $p= 0.05$ . . . . .	57
25	Visualization of MNIST handwritten digits. 196 higher representation of digits computed using (a) SAE, (b) NNSAE, (c) NCAE*, and (d) NMF are visualized using t-SNE projection [7]. . . . .	58
26	100 Receptive fields learned from the ORL faces data set using (a) SAE, (b) NCAE*, and (c) NMF. Black pixels indicate negative weights, and white pixels indicate positive weights. . . . .	59
27	Reconstruction of the ORL Faces test data using 300 receptive fields, using SAE (error=8.6447), NNSAE (error=15.7433), NCAE* (error=5.4944), and NMF (error=7.5653). . . . .	59
28	100 Receptive fields learned from ORL Faces data set using NCAE for varying nonnegativity penalty coefficients ( $\alpha$ ). Brighter pixels indicate larger weights. . . . .	60
29	100 Receptive fields learned from small NORB data set using (a) SAE, (b) NNSAE, (c) NCAE*, and (d) NMF. Black pixels indicate negative, and white pixels indicate positive weights. . . . .	61
30	An example of 7 most distinguishable categories, i.e., ship, crude, earn, acq, money-fx, grain and trade associated with top 20 words (ranked by their weights) discovered from the Reuters-21578 document corpus. The charts at the bottom row illustrate the weight impact of words on the category. . . . .	62
31	Visualization of the Reuters documents data based on the 15-dimensional higher representation of documents computed using (a) SAE and (b) NCAE*. Visualization used t-SNE projection [7]. . . . .	63
32	200 Receptive fields of the first layer of the deep network after fine-tuning using (a) all weights constrained, and (b) only Softmax weights constrained. Black pixels indicate negative weights, and white pixels indicate positive weights. . . . .	65

33	Weight distribution of the first layer of deep network after finetuning for (a) all weights constrained, and (b) Softmax-layer only constrained. According to histogram, 5.76% of weights become negative. . . . .	65
34	200 Receptive fields of the first layer of the deep network after fine-tuning using (a) SAE, (b) NCAE*, (c) DAE, (d) DpAE on the MNIST data. Black pixels indicate negative, and white pixels indicate positive weights.	66
35	16 example filters ( $\mathbf{W}^{k \in [1, \dots, 16]} = [w_{ij}]_{5 \times 5}$ ) and featuremaps ( $\mathbf{h}^{k \in [1, \dots, 16]} = [h_{ij}^k]_{24 \times 24}$ ), with feature vectors ( $\mathbf{h}_{ij} = [h_{ij}^k]_{1 \times 16}$ ), extracted from non-whitened MNIST with sigmoid nonlinearity and no pooling using (a) CAE, (b) SSCAE. Effect of sparse feature extraction using SSCAE is shown w/o pooling layer. Digits are input pixelmaps $28 \times 28$ , $n = 16$ for this example.	69
36	Structured Sparsity illustrated with on (a) two-dimensional and (b) three-dimensional space for featuremaps ( $\mathbf{h}^1, \mathbf{h}^2, \mathbf{h}^3$ ) of MNIST dataset. Each example is first projected onto the unit $\ell_2$ -ball and then optimized for $\ell_1$ sparsity. The unit $\ell_2$ -ball is shown together with level sets of the $\ell_1$ -norm. Notice that the sparseness of the features (in the $\ell_1$ sense) is maximized when the examples are on the axes [9]. . . . .	71
37	Model architecture of Structured Sparse Convolutional AutoEncoder (SSCAE) . . . . .	72
38	Comparison of 8 filters learned from MNIST by CAE and SSCAE w/o pooling (a,b) and w/ non-overlapping max-pooling (c,d) using ReLu ( $\max(0, x)$ ) nonlinearity. Select single filter and respective featuremaps shown on the digit. . . . .	73
39	SVHN data-flow visualization in (a) CAE and (b) SPCAE with 8 filters. The effect of structured sparsity is shown in encoding and decoding filters and the reconstruction. No ZCA whitening [3] is applied. . . . .	74
40	Selected featuremap of SVHN dataset extracted by (a) CAE, and (b) SSCAE with 8 filters of $11 \times 11 \times 3$ size. No ZCA whitening is applied. . . . .	75
41	16 Learnt encoding and decoding filters of (a)(c) CAE and (b)(d) SSCAE on small NORB dataset. . . . .	76

42	Learning rate convergence of CAE and SSCAE on (a) MNIST, (b) SVHN, (c) small NORB, and (d) CIFAR-10 dataset using 16 filters of $11 \times 11 \times 3$ size. . . . .	77
43	In the Alzheimer’s brain: (a) The cortex shrivels up, damaging areas involved in thinking, planning and remembering. (b) Shrinkage is especially severe in the hippocampus, an area of the cortex that plays a key role in formation of new memories. Ventricles (fluid-filled spaces within the brain) grow larger. Copyright©2011 Alzheimer’s Association®. . . . .	80
44	T1-weighted MRI images. (a) Sagittal section of dilated ventricular system of AD patient, compared to ventricular system of normal subject. (b) Coronal section through the hippocampus. AD patients have shrunken hippocampus and enlarged ventricles relative to healthy age-matched controls [10, 11]. . . . .	82
45	Schematic diagram of 3D-CAE for feature extraction of a 3D Image based on reconstructing the input. Note that the image dimension increases in the encoding layer due to full convolution, and decreases to original dimension by valid convolution. . . . .	85
46	Proposed 3D-CNN network for AD. . . . .	88
47	Selected slices of hierarchical 3D feature maps in (i,j,k) axial and (l) sagittal view extracted at 3 layers of stacked 3D-CAE indicating (i) cortex thickness and volume, (j) brain size, (k) ventricle size, and (l) hippocampus model, extracted from the brain structural MRI. The feature maps are downsampled at each layer using max-pooling to reduce the size and detect higher level features. . . . .	89
48	Manifold visualization of training data in (a,b,c) pretrained generic layers and (d,e,f,g,h) fine-tuned task-specific layers, using t-SNE projection [7].	90
49	Manifold visualization of train data in (a,b,c) pretrained generic layers and (d,e,f,g,h) fine-tuned task-specific layers, using t-SNE projection [7].	92

50	Receiver Operating Curve (ROC) and Area Under Curve (AUC) performance of (a) AD/MCI/NC, (b)AD+MCI/NC, (c) AD/NC, (d) AD/MCI and (e) MCI/NC task-specific classification models. . . . .	93
51	Averaged classification results achieved by different methods. . . . .	98

# CHAPTER I

## INTRODUCTION

The performance of machine learning methods is heavily dependent on the choice of data representation (or extracted features) on which they are applied. Therefore, much of the actual effort in training machine learning algorithms spent on the design of preprocessing pipelines and feature extraction that result in a representation of the data that can support effective machine learning. Such feature engineering is important but labor-intensive and highlights their inability to extract and organize the discriminative features from the data. Moreover, feature engineering is a way to use the domain expert and prior knowledge to compensate for that weakness [12].

Therefore, designing a fully trainable machine learning algorithm that learns the appropriate data representation, by extracting discriminative features is a key success toward Artificial Intelligence (AI). On the other hand, there is plenty of data without a label available, and employing an algorithm which can extract a set of comprehensive features is the crucial part of AI. Unsupervised feature learning includes a family of algorithms to automatically extract features without using information of domain experts. In this work, selected unsupervised feature learning algorithms, i.e. Nonnegative Matrix Factorization (NMF) [13], autoencoders (AE) and convolutional networks [4, 14, 15] are investigated. NMF is employed for feature extraction from lung Computed Tomographic (CT) images for lung segmentation [1, 2, 6]. Then image classification is investigated for understandable feature extraction in deep networks based on AE [16] and Convolutional Autoencoder (CAE). Finally, a 3D Convolutional Neural Network (CNN) is proposed for features extraction of brain Magnetic Resonance Imaging (MRI) for Alzheimer's Disease (AD) classification [17].

## CHAPTER II

### UNSUPERVISED FEATURE LEARNING METHODS

Feature learning comprises a set of algorithms to transform labeled or unlabeled data to a new space, where it can capture the parameters and patterns of variation by disentangling the hidden features. Features are learned through supervised and unsupervised learning scheme. Numerous unlabeled data is available in each domain, e.g. images, text data, speech, which contain several patterns of variation that can easily be collected for feature extraction, e.g. from Internet. The task of feature extraction from unlabeled data is known as unsupervised feature learning. Several algorithms have been developed for this task, e.g. NMF, Independent Component Analysis (ICA), and deep learning. In this section, the details of NMF and deep learning methods are addressed.

#### A Matrix decomposition

Data-adaptive representations are dependent on the statistics of data. Such representations are learned directly from the observed data by optimizing some measure that quantifies the desired properties of the representation [18]. This class of methods include Principal Component Analysis (PCA), ICA, Sparse Coding (SC), and NMF.

Assume that the observed data are in the form of a large number of i.i.d. random vectors  $\mathbf{a}_n$ , where  $n$  is the sample index. Arranging these into the columns of a matrix  $\mathbf{A}$ , then linear decompositions describe this data as,

$$\mathbf{A} \approx \mathbf{W}\mathbf{H} \tag{1}$$

where the matrix  $\mathbf{W}$  is called the mixing matrix, and contains as its columns the basis vectors (features) of the decomposition. The rows of  $\mathbf{H}$  contain the corresponding

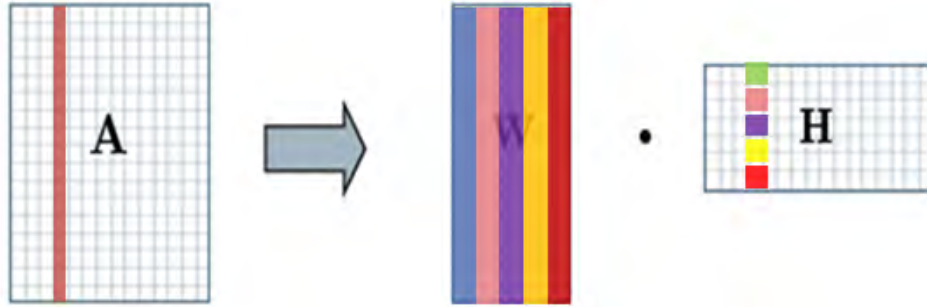


Figure 1. Unsupervised feature learning based on matrix factorization.

hidden components that apportion the contribution of each basis vector in the input vectors. A practical problem with linear decomposition methods is that both the learning  $\mathbf{W}$  and  $\mathbf{H}$  are computationally expensive, especially when a new data sample is introduced. Unsupervised feature learning is an efficient alternative to manual feature engineering, especially in case of high-dimensional images. However, generalization of feature learning is imposed by different constraints. Sections 1 and 1 cover the developed constraint to learn generalize features from training data.

## 1 Sparse coding

In linear sparse coding [19, 20], the goal is to find a decomposition in which the hidden components are sparse, meaning that they have probability densities which are highly peaked at zero and have heavy tails. This basically means that any given input vector can be well represented using only a few significantly non-zero hidden coefficients. Combining the goal of small reconstruction error with that of sparseness, one can arrive at the following objective function to be minimized,

$$J(\mathbf{W}, \mathbf{H}) = \frac{1}{2} \|\mathbf{A} - \mathbf{WH}\|^2 + \lambda \sum_{ij} f(h_{ij}) \quad (2)$$

where  $\lambda$  is a penalty term which controls the trade-off between accurate reconstruction and sparseness.

## 2 Nonnegative Matrix Factorization (NMF)

In standard sparse coding, the data is described as a combination of elementary features involving both additive and subtractive interactions. The fact that features can cancel each other out using subtraction is contrary to the intuitive notion of combining parts to form a whole. Thus, Lee et al. [13, 21] suggested the non-negative representations. The motivation for non-negative representations come from biological modeling [13], where such constraints are related to the non-negativity of neural firing rates. These non-negative representations assume that the input data  $\mathbf{A}$ , the basis  $\mathbf{W}$ , and the hidden components  $\mathbf{H}$  are all non-negative.

To find an approximate factorization, the cost function  $D(\mathbf{A}|\mathbf{WH})$  that quantifies the quality of the approximation should be defined [13]. Given a data matrix  $\mathbf{A} \in \mathbb{R}^{m \times n}$  and a positive integer  $k < m$ , nonnegative factorization of  $\mathbf{A}$  into matrices  $\mathbf{W} \in \mathbb{R}^{m \times k}$  and  $\mathbf{H} \in \mathbb{R}^{k \times n}$  is computed as,

$$\min_{\mathbf{W}, \mathbf{H}} D(\mathbf{A}|\mathbf{WH}) \text{ subject to } \mathbf{W} \geq 0, \mathbf{H} \geq 0 \quad (3)$$

where the notation  $\mathbf{A} \geq 0$  expresses nonnegativity of the entries of  $\mathbf{A}$  and  $D(\mathbf{A}|\mathbf{WH})$  indicates the factorization loss, such that,

$$D(\mathbf{A}|\mathbf{WH}) = \sum_{i=1}^m \sum_{j=1}^n d(a_{ij} | (\mathbf{WH})_{ij}) \quad (4)$$

where  $d(x|y)$  is a scalar cost function as known in the literature [22]. Several cost functions are defined in the literature. Most of them belong to the Bregman divergence family [23]. Generally, a divergence function is defined as,

$$D_\alpha(a, b) = \begin{cases} \alpha \frac{a^\alpha - b^\alpha}{\alpha} + b^\alpha(b - a) & : \alpha \in (0, 1] \\ \alpha(\log a - \log b) + (b - a) & : \alpha = 0 \end{cases} \quad (5)$$

where  $\alpha$  is chosen to define the type of the divergence function. Accordingly,  $D_1(a, b) = (a - b)^2$  measures the Euclidean distance, and  $D_0(a, b)$  defines the KullbackLeibler (KL) divergence [24], as below,

$$D_{Euclidean}(\mathbf{A}|\mathbf{WH}) = \sum_{i=1}^m \sum_{j=1}^n \frac{1}{2} (a_{ij} - (\mathbf{WH})_{ij})^2 \quad (6)$$

$$D_{KL-divergence}(\mathbf{A}|\mathbf{WH}) = \sum_{i=1}^m \sum_{j=1}^n (a_{ij} \log \frac{a_{i,j}}{(\mathbf{WH})_{ij}} - a_{ij} + (\mathbf{WH})_{ij}) \quad (7)$$

A key issue of NMF factorization is to minimize the cost function while constraining the elements of  $\mathbf{W}$  and  $\mathbf{H}$  matrices to be nonnegative. Another challenge is the existence of local minima due to non-convexity of  $D(\mathbf{A}|\mathbf{WH})$  in both  $\mathbf{W}$  and  $\mathbf{H}$ . Moreover, a unique solution to NMF problem does not exist, since for any invertible matrix  $\mathbf{B}$  whose inverse is  $\mathbf{B}^{-1}$ , a term  $\mathbf{WBB}^{-1}\mathbf{H}$  could also be nonnegative and also a solution. This is probably the main reason for non-convexity of  $D(\mathbf{A}|\mathbf{WH})$  function [24]. In [13], NMF is applied to face images, yielding features that can decompose face into interpretable parts, e.g. lips, nose, eyes, etc. This was contrasted with the holistic representations learned by PCA and vector quantization. Hosseini-Asl et al. [25] studied different optimization algorithms of NMF, described in the following sections, for document clustering. The performance of the algorithms were tested on Reuters Document Corpus for document clustering. The most efficient algorithms in terms of accuracy, entropy, purity, computational time and RMSR were identified.

Multiplicative algorithm

The multiplicative gradient descent approach is equivalent to updating each parameter by multiplying its value from previous iteration by the ratio of the negative and positive parts of the gradient of the cost function with regard to the updating parameter [21, 26]. The typical multiplicative algorithm originated by Lee et al. [13] for Euclidean and KL-divergence cost functions in Eq.(8) and Eq.(9), respectively,

$$h_{ij} \leftarrow h_{ij} \frac{(\mathbf{W}^T \mathbf{A})_{ij}}{(\mathbf{W}^T \mathbf{WH})_{ij}}, w_{ij} \leftarrow w_{ij} \frac{(\mathbf{AH}^T)_{ij}}{(\mathbf{WHH}^T)_{ij}} \quad (8)$$

$$h_{ij} \leftarrow h_{ij} \frac{\sum_i w_{ia} a_{il} / (\mathbf{WH})_{il}}{\sum_k \mathbf{W}_{ka}}, w_{ia} \leftarrow w_{ia} \frac{\sum_l h_{al} a_{il} / (\mathbf{WH})_{il}}{\sum_v h_{av}} \quad (9)$$

### Sparse NMF

SC [18] and Sparseness Constraint (SpC) [27] are developed to impose sparsity on  $\mathbf{H}$  matrix. Using the SC method, Euclidean cost function is penalized by the elements of  $\mathbf{H}$  matrix,

$$D_{SP}(\mathbf{A}|\mathbf{WH}) = \sum_{i=1}^m \sum_{j=1}^n \frac{1}{2} (a_{ij} - (\mathbf{WH})_{ij})^2 + \lambda \sum_{ij} h_{ij} \quad (10)$$

where  $\lambda \geq 0$  is the sparseness constant. In the SpC method, a *Sparseness* measure is computed based on  $\ell_1$  and  $\ell_2$ -norm for a vector  $x$ ,

$$Sparseness(x) = \frac{\sqrt{n} - (\sum |x_i|) / \sqrt{\sum x_i^2}}{\sqrt{n} - 1} \quad (11)$$

where  $n$  is the dimensionality of  $x$ . Eq.(11) to unity *iff*  $\mathbf{x}$  contains only a single non-zero component, and takes a value of zero *iff* all components are equal (up to sign), interpolating smoothly between the two extremes. Using this definition, Eq.(6) is minimized under additional constraints,

$$sparseness(w_i) = S_w, \forall i \quad (12)$$

$$sparseness(h_i) = S_h, \forall i \quad (13)$$

where  $w_i$  is the  $i$ -th column of  $\mathbf{W}$  and  $\mathbf{h}_i$  is the  $i$ -th row of  $\mathbf{H}$ . Here,  $S_w$  and  $S_h$  are the desired sparsenesses of  $\mathbf{W}$  and  $\mathbf{H}$ , respectively. These two parameters are set by the user.

### Hybrid algorithm

In this approach, the multiplicative method is used at each iterative step to approximate only the basis vector matrix  $\mathbf{W}$ . Then,  $\mathbf{H}$  is calculated using a constrained least squares (CLS) method to penalize the non-smoothness and non-sparsity of  $\mathbf{H}$ . The hybrid algorithm is denoted as Gradient Descent with Constrained Least Squares (GD-CLS) [28].

### Alternating Least Square (ALS) algorithms

In this family of algorithms, a least squares step is followed by another least squares step in an alternating fashion, thus giving rise to the ALS name, as shown in Eq.(14) and Eq.(15),

$$\min_{\mathbf{W}} D(\mathbf{A}|\mathbf{WH}) \text{ subject to } \mathbf{W} \geq 0 \quad (14)$$

$$\min_{\mathbf{H}} D(\mathbf{A}|\mathbf{W}\mathbf{H}) \text{ subject to } \mathbf{H} \geq 0 \quad (15)$$

ALS algorithms exploit the fact that, while the optimization problem of Eqs.(6) and (7) is not convex in both  $\mathbf{W}$  and  $\mathbf{H}$ , it is convex in either  $\mathbf{W}$  or  $\mathbf{H}$ , corresponding to Eqs.(14) and (15), respectively. Thus, given one matrix, the other matrix can be found with a simple least squares computation. However, the least square problem should result in nonnegative  $\mathbf{W}$  and  $\mathbf{H}$ , which means least square algorithm should be of class of nonnegative least square. Several algorithms have been proposed to keep nonnegativity constraint in ALS algorithm. The basic ALS algorithm uses nonnegativity threshold on elements of  $\mathbf{W}$  and  $\mathbf{H}$  matrices, to remove the nonnegative elements [29]. ALS method based on Projected Gradient Method (ALS-PGD) was proposed in [30], which contains nonnegativity constraint in the gradient based update algorithms.

An Alternating Nonnegative Least Square method (ANLS) based on Active Set (ANLS-AS) and Block Pivoting method (ANLS-BP) are proposed to solve nonnegative constrained least squares problem in a fast way [31,32]. Using ANLS-AS, the following ANLS problem with multiple right hand side,

$$\min_{\mathbf{W}>0} \|\mathbf{H}^T \mathbf{W}^T - \mathbf{A}^T\|^2 \quad (16)$$

$$\min_{\mathbf{H}>0} \|\mathbf{W}\mathbf{H} - \mathbf{A}\|^2 \quad (17)$$

are converted to the form of Eq.(18), alternately,

$$\min_{\mathbf{G}>0} \|\mathbf{B}\mathbf{G} - \mathbf{Y}\|^2 \quad (18)$$

where  $\mathbf{B} \in \mathbb{R}^{p \times q}$  and  $\mathbf{Y} \in \mathbb{R}^{p \times l}$ . Then Eq.(18) is decoupled into  $l$  independent Non-Negative Least Square (NNLS) problem each with single right-hand side as,

$$\min_{\mathbf{G}>0} \|\mathbf{B}\mathbf{G} - \mathbf{Y}\|^2 = \min_{g_1>0} \|\mathbf{B}g_1 - Y\|^2, \dots, \min_{g_l>0} \|\mathbf{B}g_l - Y\|^2 \quad (19)$$

where  $\mathbf{G} = [g_1, \dots, g_l] \in \mathbb{R}^{q \times l}$  and  $\mathbf{Y} = [y_1, \dots, y_l] \in \mathbb{R}^{p \times l}$ . Then each independent NNLS problems is solves using Active Set algorithm proposed in [32].

Using ANLS-BP method, a single right-hand side problem is solved using Block Principle Pivoting algorithm proposed, and it was generalized for multiple right-hand side problem [31]. These methods have also been developed to include sparsity and regularity inside the NNLS problem.

NMF based on  $\beta$ -divergence

The  $\beta$ -divergence is a family of cost functions parameterized by a single shape parameter  $\beta$ . This cost function could takes the form of Euclidean distance, KL divergence, and Itakura-Saito divergence as special cases ( $\beta = 2, 1, 0$ , respectively).

$$d_{\beta}(x|y) = \begin{cases} \frac{1}{\beta(\beta-1)}(x^{\beta} + (\beta - 1)y^{\beta} - \beta xy^{\beta-1}) & : \beta \in \mathbb{R}(0, 1) \\ x \log \frac{x}{y} - x + y & : \beta = 1 \\ \frac{x}{y} - \log \frac{x}{y} - 1 & : \beta = 0 \end{cases} \quad (20)$$

Fevotte et al. [22, 33] proposed algorithms, which are based on a surrogate auxiliary function (a local majorization of the criterion function). They developed a majorization minimization algorithm that leads to multiplicative updates, and a Majorization Equalization (ME) algorithm. The ME algorithm is used for NMF based on  $\beta$ -divergence (Beta-ME) in the experiment section.

NMF based on Correntropy

The correntropy cost function is defined as,

$$D_{Correntropy}(\mathbf{A}|\mathbf{WH}) = - \sum_{i=1}^m \sum_{j=1}^n \exp\left(\frac{-(a_{ij} - (\mathbf{WH})_{ij})^2}{2\sigma^2}\right) \quad (21)$$

where  $\sigma$  is a parameter of correntropy measure [34, 35]. The optimization algorithms try to minimize the correntropy, since it is a measure of similarity instead of distance between two elements. Ensari et al. [36, 37] used the general algorithm of Constrained Gradient Descent (CGD) method [38] for minimizing the correntropy function, and compared the results with the projected gradient descent method of Euclidean cost function. The major disadvantage of CGD is its high sensitivity to  $\sigma$  value of the cost

function. Du et al. [39] proposed a half-quadratic optimization algorithm to solve NMF based on correntropy cost function, and developed a multiplicative algorithm to solve NMF. Hosseini-Asl et al. [40] developed a multiplicative algorithm of NMF based on correntropy loss for document clustering

## B Deep learning

Recent studies have shown that deep architectures are capable of learning complex data distributions while achieving good generalization performance and efficient representation of patterns in challenging recognition tasks [14, 15, 41–44]. Deep architecture networks have many levels of nonlinearities, giving them an ability to compactly represent highly nonlinear complex mappings. However, they are difficult to train, since there are many hidden layers with many connections, which causes gradient-based optimization with random initialization to get stuck in poor solutions [45]. To improve on this bottleneck, a greedy layer-wise training algorithm was proposed in [46], where each layer is separately initialized by unsupervised pre-training, then the stacked layers are fine-tuned using a supervised learning algorithm [14, 45]. It was shown that an unsupervised pre-training phase of each layer helps in capturing the patterns in high-dimensional data, which results in a better representation in a low-dimensional encoding space [14], and could result in more sparse feature learning [47]. This pre-training also improves the supervised fine-tuning algorithm for classification by guiding the learning algorithm towards local minima of the error function, that support better generalization on training data [48, 49].

There are two popular algorithms for unsupervised learning which have been shown to work well to produce a good representation for initializing deep structures [50]: Restricted Boltzmann Machines (RBMs) trained with contrastive divergence [51], and different types of autoencoders [15].

## 1 Autoencoders

An autoencoder network is an unsupervised learning algorithm that tries to reconstruct its input vector in the output [50, 52]. As shown in Figure 2(a), it tries to learn an encoding function,

$$\hat{\mathbf{x}} = f_{\mathbf{W}, \mathbf{b}}(\mathbf{x}) \approx \mathbf{x} \quad (22)$$

where  $\mathbf{x}$  is the input vector, while  $\mathbf{W} = \{\mathbf{W}_1, \mathbf{W}_2\}$  and  $\mathbf{b} = \{\mathbf{b}_1, \mathbf{b}_2\}$  represent weights and biases of both layers, respectively. It takes an input vector  $\mathbf{x} \in [0, 1]^n$ , and first maps it to a hidden representation through a deterministic mapping, parametrized by  $\theta_1 = \{\mathbf{W}_1, \mathbf{b}_1\}$ , and given by

$$\mathbf{h} = g_{\theta_1}(\mathbf{x}) = \sigma(\mathbf{W}_1 \mathbf{x} + \mathbf{b}_1) \quad (23)$$

where  $\mathbf{h} \in [0, 1]^{n'}$ ,  $\mathbf{W}_1 \in R^{n' \times n}$ ,  $\mathbf{b}_1 \in R^{n' \times 1}$ , and  $\sigma(x)$  denotes an element-wise application of the logistic sigmoid,  $\sigma(x) = 1/(1 + \exp(-x))$ . The resulting hidden representation,  $\mathbf{h}$ , is then mapped back to a reconstructed vector,  $\hat{\mathbf{x}} \in [0, 1]^n$ , by a decoding function, parametrized by  $\theta_2 = \{\mathbf{W}_2, \mathbf{b}_2\}$ ,

$$\hat{\mathbf{x}} = g_{\theta_2}(\mathbf{h}) = \sigma(\mathbf{W}_2 \mathbf{h} + \mathbf{b}_2) \quad (24)$$

where  $\mathbf{W}_2 \in R^{n \times n'}$  and  $\mathbf{b}_2 \in R^{n \times 1}$ . To optimize the parameters of the model in Eq.(22), i.e.  $\theta = \{\theta_1, \theta_2\}$ , the average reconstruction error is used as the cost function,

$$J_E(\mathbf{W}, \mathbf{b}) = \frac{1}{m} \sum_{r=1}^m \left( \frac{1}{2} \|\hat{\mathbf{x}}^{(r)} - \mathbf{x}^{(r)}\|^2 \right) \quad (25)$$

where  $m$  is the number of training samples.

By imposing meaningful limitations on parameters  $\theta$ , e.g. limiting the dimension  $n'$  of the hidden representation  $\mathbf{h}$ , the autoencoder learns a compressed representation of the input, which helps discover the latent structure of data in a high-dimensional space.

Sparse representation can provide a simple interpretation of the input data in terms of a reduced number of parts and by extracting the structure hidden in the data. Several algorithms were proposed to learn a sparse representation using

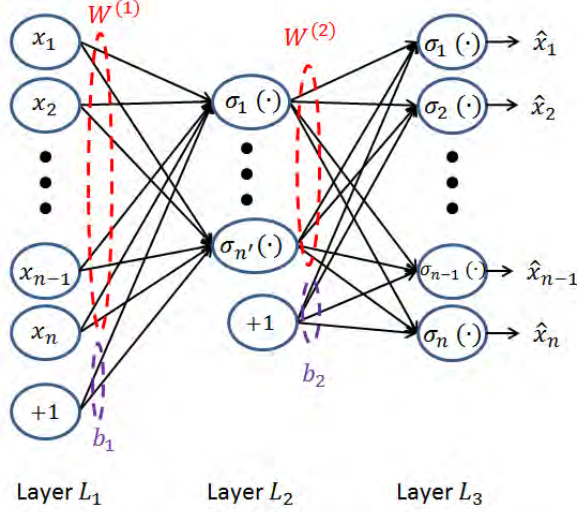


Figure 2. Schematic diagram of a three-layer autoencoder.

autoencoders [46, 53]. One common method for imposing sparsity is to limit the activation of hidden units  $\mathbf{h}$  using the KL divergence function [54, 55]. Let  $h_j(\mathbf{x}^{(r)})$  denote the activation of hidden unit  $j$  with respect to the input  $\mathbf{x}^{(r)}$ . Then the average activation of this hidden unit is:

$$\hat{p}_j = \frac{1}{m} \sum_{r=1}^m [h_j(\mathbf{x}^{(r)})] \quad (26)$$

To enforce sparsity, the average activation  $\hat{p}_j = p$  is constrained, where  $p$  is the sparsity parameter chosen to be a small positive number near 0. This also relates to the normalization of the input to the neurons of the next layer which results in faster convergence of training using the backpropagation algorithm [56]. To use this constraint in Eq.(68), The KL divergence similarity between  $\hat{p}_j$  and  $p$  is minimized by,

$$J_{\text{KL}}(p \parallel \hat{\mathbf{p}}) = \sum_{j=1}^{n'} p \log \frac{p}{\hat{p}_j} + (1 - p) \log \frac{1 - p}{1 - \hat{p}_j} \quad (27)$$

where  $\hat{\mathbf{p}}$  is the vector of average hidden activities. To prevent overfitting, a weight decay term is also added to the cost function of Eq.(25) [57]. The final cost function for learning a Sparse Autoencoder (SAE) becomes as follows:

$$\begin{aligned}
J_{\text{SAE}}(\mathbf{W}, \mathbf{b}) &= J_{\text{E}}(\mathbf{W}, \mathbf{b}) + \beta J_{\text{KL}}(p \parallel \hat{\mathbf{p}}) \\
&+ \frac{\lambda}{2} \sum_{l=1}^2 \sum_{i=1}^{s_l} \sum_{j=1}^{s_{l+1}} \left( w_{ij}^{(l)} \right)^2
\end{aligned} \tag{28}$$

where  $\beta$  controls the sparsity penalty term,  $\lambda$  controls the penalty term facilitating weight decay, and  $s_l$  and  $s_{l+1}$  are the index terms for nodes in adjacent layers.

## 2 Deep networks based on stacked autoencoder

Many hidden layers with many connections makes it difficult to train deep networks. The gradient-based optimization with random initialization tends to get stuck in poor solutions [45]. To overcome this problem, a greedy layer-wise training algorithm was proposed in [46], where each layer is separately initialized by unsupervised pre-training, then the stacked layers are fine-tuned using a supervised learning algorithm [14, 45]. It was shown that an unsupervised pre-training phase of each layer helps in step-wise capturing of patterns in high-dimensional data, which results in better representation in low-dimensional encoding space [14], and more sparse feature learning [47]. It was also shown that it improves the supervised fine-tuning algorithm for classification by guiding the learning algorithm towards basins of attraction of minima, which supports better generalization from the training data set [48, 49].

The greedy layer-wise approach for pre-training a deep network works by training each layer in step-wise manner. A stacked autoencoder is a neural network consisting of multiple encoding layers of autoencoders stacked on top of each other. Let  $\{\mathbf{W}_1^k, \mathbf{b}_1^k\}$  and  $\{\mathbf{W}_2^k, \mathbf{b}_2^k\}$  denote the set of encoding and decoding parameters of  $k$ -th autoencoder, respectively. Then the encoding step for the  $k$ -th stacked autoencoder is computed by forward propagation of data through encoder layer,

$$\mathbf{h}^k = \sigma(\mathbf{z}^k) \tag{29}$$

$$\mathbf{z}^{k+1} = \mathbf{W}_1^k \mathbf{h}^k + \mathbf{b}_1^k \tag{30}$$

The hierarchical features extracted in sequence through layers  $k = [1, \dots, L]$  are capturing the patterns of variations in the input  $\mathbf{x}$ . As shown in Figure 3, the features from the stacked autoencoder can be used for classification problems by forwarding  $\mathbf{h}^L$  to a softmax classifier.

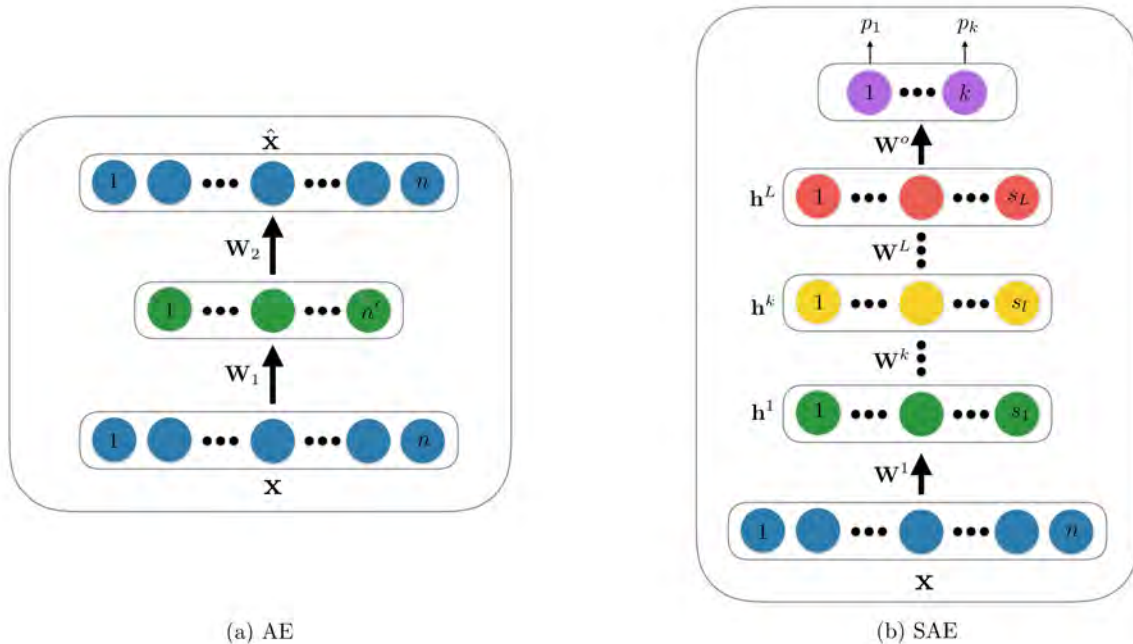


Figure 3. Architecture of (a) autoencoder and (b) stacked autoencoder deep network [3].

### 3 Convolutional Autoencoder and Networks (CAE/CNN)

The conventional method for unsupervised feature extraction from a dataset is based on encoding-decoding scheme, i.e. autoencoder. In this model, the data is transformed into a low-dimensional space (in the hidden layer) and reconstructed back (in the output layer) to the original space. To train the autoencoder to extract features capturing patterns of variation, the reconstruction error is reduced using back-propagation algorithm, while some properties of the low-dimensional space are constrained. In case of high-dimensional images, it is computationally expensive to extract global features (input weights) from image, since the number of weights in the input and output layers of network grows very large, and also needs a large training data [4]. Moreover, a fully connected autoencoder tries to learn global features,

whereas local features are more suitable for extracting patterns from high-dimensional images. To overcome this problem, an autoencoder based on nodes with locally connected and shared weights (CAE) is used to extract unsupervised local features from high-dimensional 2D images [58–60]. Using this method a 2D image is reduced using hierarchical layers of CAE, where the hidden activities (feature map) of each CAE is used as a training for the lower-layer CAE.

### Locally connected layers

Inspired by the localized receptive field of neurons in the visual cortex [61], it was proposed to restrict the connection between the hidden units and the input units, allowing each hidden unit to span only to a local neighboring input units [4]. In other words, each hidden neuron only connects to a small number of neighboring nodes of the input vector, as shown in Figure 4(a).

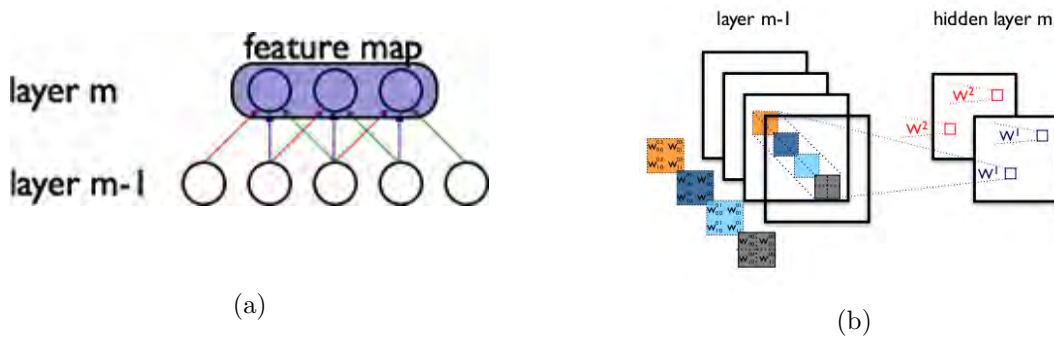


Figure 4. Construction of convolutional neural network by (a) locally connected neuron, and (b) convolution over feature maps [4, 5].

### Convolutional neural networks

Each receptive field in CNN, known as convolutional filter,  $\mathbf{W}^k$ , is shared across local neurons belonging to the same featuremap, as shown in Figure 4(b). As exemplified in Figure 4 (a), three hidden units belonging to the same featuremap are depicted. Weights of the same color are shared, i.e. constrained to be identical.

Conceptually, a feature map is obtained by convolving the input image with a linear filter, adding a bias term and then applying a non-linear function. Let  $\mathbf{h}^k$

denotes the  $k$ -th featuremap at a hidden layer, whose parameters, i.e. filter and bias, are determined by the weights  $\mathbf{W}^k$  and bias  $b^k$ , respectively. Then the feature map  $\mathbf{h}^k$  is computed as,

$$h_{ij}^k = \sigma \left( (\mathbf{W}^k * x)_{ij} + b_k \right) \quad (31)$$

where  $(*)$  indicates convolution operation.

To capture different patterns of variation in high-dimensional images, the hidden layers in CNN are composed of multiple featuremaps,  $\{\mathbf{h}^{(k)}, k = 0, \dots, K\}$ . An exemplified two-layered CNN is shown in Figure 4(b).

LeCun et al. [4] suggested a CNN model for digit recognition, as depicted in Figure 5. The lower layers are comprised of stacked convolutional and subsampling layers, e.g. max-pooling, mean-pooling, etc. Following several layers of convolution and subsampling, the input data dimension is reduced. Subsequently, several fully connected layers are stacked on top of lower layers with softmax regression layers as an output, to perform classification.

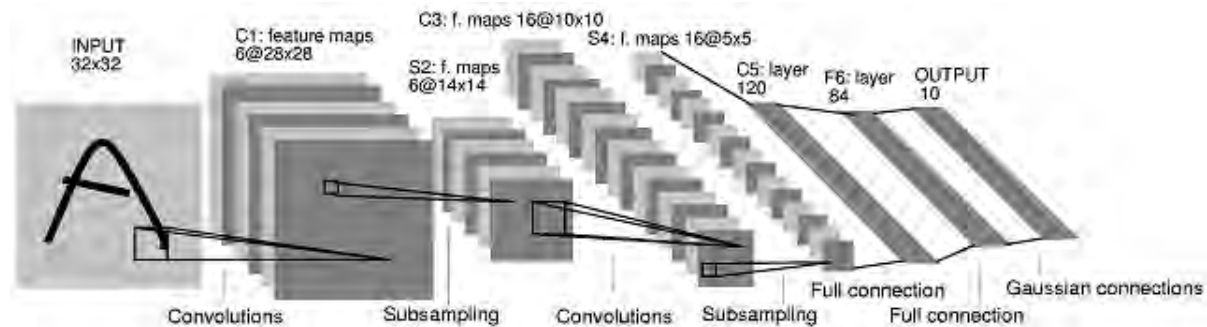


Figure 5. Architecture of LeNet-5 model for digit recognition [4].

## C Unsupervised and Transfer Learning

Supervised learning requires enough labeled data to achieve good performance on the given task. However, with limited labeled data, the classifier could benefit from additional knowledge, i.e. from similar learning task, rather than the provided dataset. To achieve such a generalized classifier, transfer learning is used that employs the

already absorbed knowledge, e.g. deep network weights, from a similar learning task, and uses them for initialization of the goal classification task [62–65].

Domain adaptation [66–68] refers to the knowledge transfer, by learning a classifier from a source data, and using the trained model on the target data. In supervised learning, where data  $X$  is drawn i.i.d from a distribution  $D_S$ , the goal is to learn a hypothesis  $h : X \rightarrow Y$  with minimization of a loss function  $\mathcal{L}$ . In domain adaptation, however, with given source domain  $\mathcal{D}_s$  and the target domain  $\mathcal{D}_t$ , the goal is to learn hypothesis  $h : X \rightarrow Y$  to minimize the loss  $\mathcal{L}$  on target domain, by transferring knowledge from source domain. To boost the prediction performance of deep model, leveraging the unsupervised feature learning is considered by transferring the trained features (knowledge) to the target domain [69].

## CHAPTER III

### MOTIVATIONS AND RESEARCH GOALS

This work is investigating the use of unsupervised feature learning for developing image segmentation, classification, and disease diagnosis models. This overview chapter provides motivations and summary of research goals for the reader of remaining chapters.

#### A Image segmentation

For image segmentation, lung segmentation from CT scans is investigated. NMF is chosen as the unsupervised feature learning technique with the following goals:

NMF-based method for lung segmentation

Each voxel can be defined as a feature vector using its neighborhood pixels. NMF can be used as an unsupervised feature learning to extract low-dimensional features to discriminate between voxels.

High-dimensional 3D lung segmentation using NMF

3D images contain large number of voxels, and implementing NMF is unfeasible due to time-consuming large matrix operations, i.e. inverse, transpose. This part focuses on developing an NMF-based approach for high-dimensional 3D images using online learning that addresses the issue of high computing costs.

Automatic detection of pathology using NMF

Identifying the number of organs/pathologies in lung CT images is usually left to the human expert which provides the information to CAD systems. This aspect of

work investigates a model that automatically detects the number of distinguishable organs in the CT scans through NMF online learning. The investigation and results are covered in Chapter IV.

## B Image classification

The image classification part of this work focuses on improving the interpretability in unsupervised feature learning method. The study is divided in two parts: (i) Autoencoder and (ii) Convolutional Autoencoder.

### Interpretability in autoencoder

In Chapter V-A and Chapter V-B, the interpretability in deep network for autoencoders is investigated in context of deep models classification performance with the following goals:

- An autoencoder is used as unsupervised feature learning to pretrain deep models for classification and recognition. To improve interpretability in classification models to discriminate between different objects, it should be able to distinguish between parts across object. Training an autoencoder to extract part-based features is a crucial step toward introducing interpretability in deep models.
- Pretrained deep model of autoencoder is fine-tuned by the labeled data. To maintain the interpretability in stacked layers of autoencoders during fine-tuning, a modified learning algorithm based on back-propagation should be developed. This part investigate the possible solution in developing an interpretable and accurate deep model using an interpretable autoencoder.

### Interpretability in convolutional autoencoder

To expand the investigation of feature extraction in AE into CAE model, the part-based feature extraction is further studied with following objectives:

- Convolutional networks preserve the local and spatial correlation in featuremaps due to convolution. To extract discriminative and compressed feature from high-dimensional images, the features should be sparsified. This part focuses on the sparsification of feature learning in CAE.
- To introduce interpretability in CAE models, the featuremaps should be learned in a way to decompose parts of objects across themselves. This part investigates how sparsity can be applied to improve interpretability in CAE models.
- Convnets are complex models composing several convolutional filters to achieve good classification. CAE as the core models of Convnets are used for pretraining. This part investigates how to simplify CAE models using sparsity.

The investigation and results are covered in Chapter V-D.

## C Disease diagnosis

For disease diagnosis, the AD classification is selected for investigation. The goals are:

### A 3D deep network for MRI-based AD classification

Accurate AD classification is highly dependent on detecting the AD biomarkers across different regions of brain. Most developed CAD systems employ human knowledge to design a pipeline to extract AD biomarkers and train classifiers on top of the extracted features. This part focuses on developing a fully-trainable model for AD feature extraction and classification, and to reduce the use of domain expert's knowledge.

### Transfer learning for improved AD classification

Different CAD systems can be trained on different datasets for AD classification. To leverage the access to different datasets for better AD diagnosis, transfer learning has been employed to share the extracted and trained features from different datasets.

This will require investigation on how to share the extracted features between datasets to improve the classification. The investigation and results are covered in Chapter VI.

## CHAPTER IV

# LUNG SEGMENTATION BASED ON UNSUPERVISED FEATURE LEARNING

Accurate automated segmentation of lung tissues from CT images is of profound importance for developing non-invasive computer-assisted system for early diagnosis of lung cancer and other pulmonary diseases [70–75]. This problem is challenging due to differences in CT scanners and scanning protocols, inhomogeneities of lungs, and lack of a strong pixel/voxel-wise discriminatory signal between tissues surrounding the lungs and pulmonary structures, such as arteries, veins, bronchi, and bronchioles, etc. Most lung segmentation techniques employ adaptive signal thresholding [76], evolving parametric [77], geometric (level-set based) [78], and geodesic deformable boundaries (active contours, or snakes), including active contours with a shape prior defining a stopping criterion [79] or guiding forces [80–82], and low-order Markov-Gibbs random field models of CT images [83–85].

Most of these techniques have notable drawbacks. Different image acquisition protocols and scanner types, as well as signal inhomogeneities in pulmonary structures hinder signal thresholding. Deformable models are excessively sensitive to initialization. Their conventional external forces depending on, e.g., edges, gradients, and other local signal properties fail to capture natural lung inhomogeneities and therefore the model towards a true lung boundary. More flexible active contours based on shape priors depend on how accurately the prior is aligned to the input image. Markov random field models usually take no account of high-order spatial signal dependencies, which are necessary to accurately describe complex lung appearances. Compared to all these approaches, the recent segmentation by NMF [1, 2, 86–88] demonstrated its ability in feature extraction from the image and using them for

discriminating between different objects/modalities.

Based on the literature, the NMF is rarely used for segmentation, especially, in application to medical images. The advantages include extracting one or more characteristic basis feature vectors per each region of interest; projecting each initial pixel/voxel-wise vector onto the feature space, and associating every projection with the most relevant feature(s). Xie et al. [86] applied the NMF to extract from diffusion tensor images (DTI) of rat brains the basis images of spinal cord, corpus callosum, and hippocampus regions. Then the image sites were stratified by the  $K$ -means clustering [89] of their projections in the decomposition matrix  $\mathbf{H}$ . To segment a multispectral barley grain cross-section, Lazar et al. [87] decorrelated the image dataset with the Principal Component Analysis (PCA) and performed the NMF of the decorrelated data. The data samples were classified by their closeness to the features found, where the closeness being evaluated by the maximum coefficient in the corresponding decomposition vector of the matrix  $\mathbf{H}$ . Sandler et al. [88] divided a texture mosaic into non-overlapping rectangular blocks, described each block with a vector of outputs of linear Gabor, or wavelet filters. The NMF based on the Earth mover's distance was used to find the representative basis features and classify each pixel using the Bayesian maximum a posteriori (MAP) decision rule.

To extract unsupervised features for lung segmentation, three frameworks for segmentation of 3D lung images are proposed, where the spatial information of the image is defined based on novel features that are extracted using NMF, Incremental NMF (INMF), and Incremental Constrained NMF (ICNMF). To perform the segmentation, the new extracted NMF spatial features is proposed in order to discriminate between lung and chest voxels [2]. The following sections explain the details of each algorithm.

## A NMF-based segmentation

In this section, a novel frameworks are proposed for lung image segmentation, which consists of four steps as shown in Figure 7. In the first step, the CT image volume is preprocessed to remove its background. Then, an NMF-based visual

appearance modeling is applied to extract novel image features. In the third step, the lung data is segmented based on the extracted features of the image. Finally, the segmentation is refined using a 3D region growing approach to produce the final segmentation. These steps are discussed in detail in the following section.

## 1 Preprocessing

Due to the similarity between background and gray values of the lung voxels, the first step of the proposed framework is to remove the background from the CT image using a 3D region growing method. An illustration for removing the background using this method is shown in Figure 6.

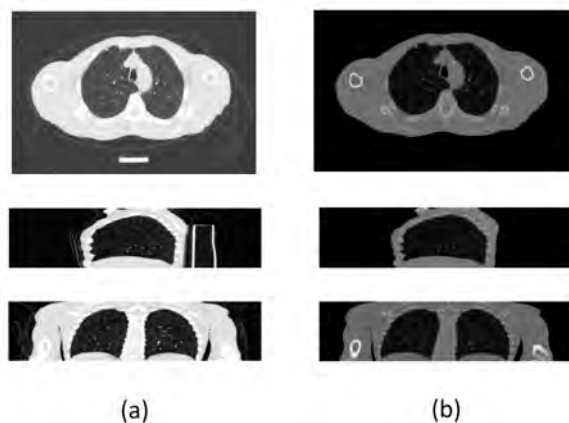


Figure 6. Original 3D image slices (a) before and (b) after removing their background by 3D region growing from a seed at the 3D image corner.

## 2 NMF-based visual appearance modeling

The traditional way to model the spatial interaction of the image voxels is to take into account their neighboring voxels. In this section, a new spatial interaction model is developed for the lung data by extracting new spatial features based on NMF. Let  $\mathbf{G}_{\mathbf{N}_{x,y,z}} \in \mathcal{Q}^{I_x \times I_y \times I_z}$  be the image signals of the neighborhood of the voxel  $(x, y, z)$ . By including the image signals of the neighborhood of all voxels, a 4D matrix  $\mathbf{G} \in \mathcal{Q}^{XYZ \times I_x \times I_y \times I_z}$  is composed. In the literature, different methods were proposed for decomposition of  $\mathbf{G}$ . For example, the Tucker1 Nonnegative Tensor Decomposition

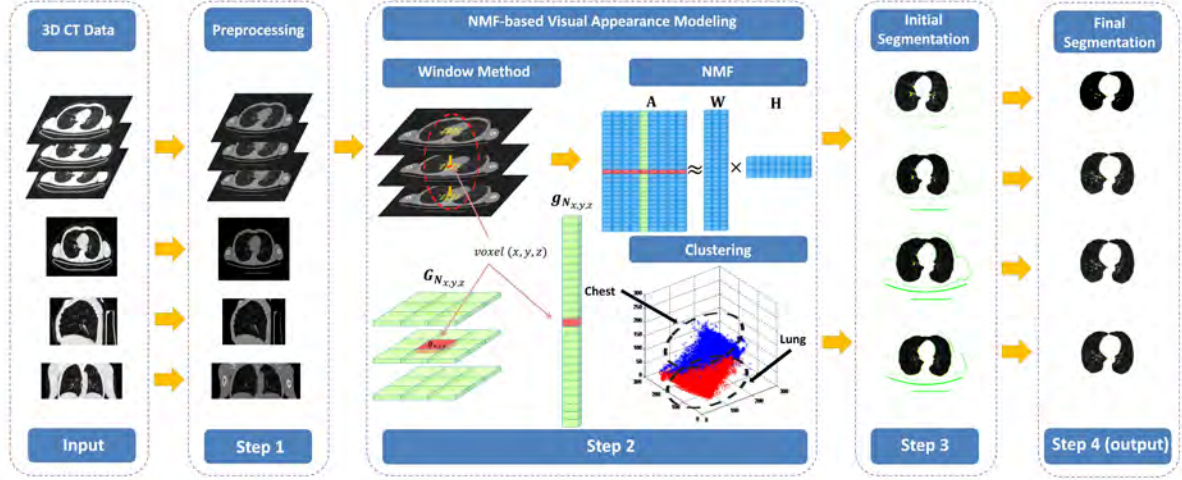


Figure 7. The developed framework for 3D lung segmentation from CT images using NMF.

(NTD) was applied to decompose a multi-dimensional matrix [90]. However, this method is computationally complex for large matrices (since it is computed using iterative steps of multi-dimensional matrix product and division), which is the case for the 3D lung data (i.e.,  $\mathbf{G} \in \mathcal{Q}^{XYZ \times I_x \times I_y \times I_z}$ ). To overcome this limitation, NMF is used instead of NTD. In this way, the decomposition computation becomes less complex due to replacing multi-dimensional matrix computation by 2D matrix computation. Using NMF, the input data matrix  $\mathbf{A} \in \mathcal{Q}^{I_x I_y I_z \times XYZ}$  can be factorized into two matrices:

$$\mathbf{A} \approx \mathbf{W}\mathbf{H} \quad (32)$$

where  $\mathbf{W} \in \mathbb{R}^{+I_x I_y I_z \times J}$  contains the basis vectors of the new feature space, and the vectors of  $\mathbf{H} \in \mathbb{R}^{+J \times XYZ}$  represent the new features of the voxels that model the visual appearance of the image [13]. To process the 3D lung data using NMF, the spatial feature  $\mathbf{G}_{\mathbf{N}_{x,y,z}}$ , for each voxel  $(x, y, z)$  is converted to the vector  $\mathbf{g}_{\mathbf{N}_{x,y,z}}$  in the input data matrix  $\mathbf{A}$  (see Figure 7). To estimate  $\mathbf{W}$  and  $\mathbf{H}$  from Eq.(32), the Euclidean cost function

$$D(\mathbf{A}|\mathbf{W}\mathbf{H}) = \frac{1}{2} \|\mathbf{A} - \mathbf{W}\mathbf{H}\|^2 \quad (33)$$

is minimized. Since the advent of NMF, several optimization algorithms have been developed for minimizing Eq.(33). Multiplicative, ALS, and Projected Gradient Descent (PGD) are examples of basic algorithms developed for minimization of

Eq.(33) [29]. In this section, the ANLS-AS method [31]. The reason for using this method is that the convergence of the ANLS-AS is faster than the other methods in terms of cost function minimization [25, 31]. Therefore, ANLS-AS results in factorization in lesser iteration. In this method, the optimal  $\mathbf{G} \in \mathbb{R}^{q \times l}$  in a general cost function of the form of Eq.(34) is solved,

$$\arg \min_{\mathbf{G} > 0} \|\mathbf{B}\mathbf{G} - \mathbf{Y}\|^2 \quad (34)$$

where  $\mathbf{B} \in \mathbb{R}^{p \times q}$  and  $\mathbf{Y} \in \mathbb{R}^{p \times l}$  are given. Using ANLS-AS method, Eq.(34) can be decoupled into  $l$  independent sub-problems:

$$\begin{aligned} \arg \min_{\mathbf{G} > 0} \|\mathbf{B}\mathbf{G} - \mathbf{Y}\|^2 = \\ \arg \min_{\mathbf{g}_1 > 0} \|\mathbf{B}\mathbf{g}_1 - \mathbf{Y}\|^2, \dots, \arg \min_{\mathbf{g}_l > 0} \|\mathbf{B}\mathbf{g}_l - \mathbf{Y}\|^2 \end{aligned} \quad (35)$$

where  $\mathbf{G} = [\mathbf{g}_1, \dots, \mathbf{g}_l] \in \mathbb{R}^{q \times l}$  and  $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_l] \in \mathbb{R}^{p \times l}$ , and each sub-problem is solved independently. To implement ANLS-AS for Eq.(33), this cost function is minimized by alternately solving:

$$\arg \min_{\mathbf{W} > 0} \|\mathbf{H}^T \mathbf{W}^T - \mathbf{A}^T\|^2 \quad (36)$$

$$\arg \min_{\mathbf{H} > 0} \|\mathbf{W}\mathbf{H} - \mathbf{A}\|^2 \quad (37)$$

where at each iteration, Eq.(36) and Eq.(37) are converted to the form of Eq.(34) alternately, and then solved by Eq.(35). When the convergence criterion is satisfied, each column in  $\mathbf{W}$  defines a basis of visual appearance, and each column of  $\mathbf{H}$  encodes each voxel in the new feature (J-dimensional) space. As a result,  $\mathbf{H}$  is expressing a new set of visual appearance. The basic steps of the proposed NMF-based visual appearance modeling is shown in Algorithm 1.

### 3 Segmentation

At this step, an initial segmentation for the lung is generated. First, the K-means clustering algorithm is used to cluster the voxels in the new feature space  $\mathbf{H}$  in two groups. Euclidean distance is used as a measure of distance between voxels.

---

**Algorithm 1** NMF-based Visual Appearance Modeling

---

1. Create  $\mathbf{G}_{\mathbf{N}_{x,y,z}}$  for each voxel, then generate matrix  $\mathbf{A}$  by including  $\mathbf{g}_{\mathbf{N}_{x,y,z}}$  of all voxels.
  2. Compute NMF,  $\mathbf{A} \approx \mathbf{WH}$ , based on ANLS-AS method as follows:
    - (a) Convert Eq.(36) and Eq.(37) alternately, to the corresponding form in Eq.(34)
    - (b) Solve the resulting cost functions based on Eq.(35) to compute the  $\mathbf{H}$  and  $\mathbf{W}$  matrices
    - (c) Iterate Steps 2-(a) and 2-(b) until convergence criterion satisfied, to obtain  $\mathbf{H}$  which represents the NMF-based visual appearance model for the 3D image.
- 

Then  $\ell_2$ -norm of centers of clusters is measured to determine the lung cluster.

Therefore, the cluster which its center is closer to the origin is assumed to be the lung cluster. The reason is that the gray value  $g_{x,y,z}$  of lung voxels is smaller than chest voxels, which results in smaller feature vector, in terms of  $\ell_2$ -norm, in  $\mathbf{H}$ . Finally, segmented image of the lung is generated from its cluster.

At the final step, the initial segmented lung is refined in two stage. In the first stage, the 3D region growing algorithm is used to remove the mis-clustered voxels, by choosing an initial seed point inside the segmented lung. At the second stage, the 2D region growing algorithm is employed to keep connectivity inside the lung region. This step is executed by choosing an initial seed point outside the segmented lung, where the whole voxels inside the lung is assumed as lung's voxels The proposed method of lung segmentation is outlined in Algorithm 2.

---

**Algorithm 2** Lung Segmentation Using NMF-based Visual Appearance Model

---

1. Preprocess image for removing background
  2. Extract NMF-based visual appearance model of the 3D image using Algorithm 1
  3. Cluster voxels into two groups of lung and chest using K-means clustering algorithm
  4. Refine segmented lung using 3D and 2D region growing to remove mis-clustered voxels, and keeping connectivity inside lung regions
- 

## B INMF-based automatic segmentation of pathological lungs

The main limitation of the conventional NMF model [13] is that it works only when the actual number of clusters,  $J$ , is known prior to the segmentation process.

Moreover, due to the slice-wise segmentation, the inter-slice signal dependencies were taken into account only implicitly, via the context. This limitation was overcome in [1] by decomposing the entire 3D context image in a voxel-by-voxel mode with a computationally more efficient Multiplicative Update Rule (MUR)-based INMF [91]. After removing an irrelevant image background by simple 3D region growing, as detailed in Section 1, the MUR-based INMF was applied to simultaneously estimate, in the space of decomposition vectors  $\mathbf{H}$ , an initially unknown number of clusters representing main objects, such as lungs, chest tissues, veins, arteries etc. Then all the decomposition vectors were reassigned to the darkest lung cluster and the brightest chest cluster, and the resulting 3D region map was refined by the 3D connected component analysis. Using this method, the number of image clusters in a pathological lung is estimated in an automatic iterative mode.

## 1 INMF-based visual appearance modeling

INMF is an online algorithm which is based on updating  $\mathbf{W}$  and  $\mathbf{H}$  matrices iteratively, when a new data sample (e.g. the neighborhood vector of the new voxel) is added to data matrix  $\mathbf{A}$ . By including all voxels' neighborhood vectors into matrix  $\mathbf{A}$ , the basis vector  $\mathbf{W}$  is trained and  $\mathbf{H}$  can be directly calculated from Eq.(32). The details of the INMF algorithm for visual appearance modeling is illustrated in Algorithm 3.

## 2 Segmentation

In this step, a two-step clustering approach is used to obtain the segmentation of the lung fields. First, the K-means clustering approach is applied on voxels in the  $\mathbf{H}$  space using  $J$  clusters. Then the  $\ell_2$ -norm of each cluster centroid is calculated. Since the signals of the lung voxels are darker (smaller values) than those of the chest, the centroid with the smallest  $\ell_2$ -norm is classified as the lung cluster centroid and the largest one as the chest cluster centroid. Second, the K-means approach is applied to all data points to classify them as lung or chest based on the nearest distance to lung and chest centroids. To refine the segmentation, 3D connected component analysis is

---

**Algorithm 3** INMF-based Visual Appearance Modeling
 

---

**1: Initialization**

- (i) Initialize  $\mathbf{A}$  by the neighborhood vector  $\mathbf{g}_{\mathbf{N}_{0,0,0}}$
- (ii) Set the number of clusters to 1 (i.e.  $J = 1$ )
- (iii) Initialize  $\mathbf{W}$  and  $\mathbf{H}$  randomly and set the first cluster centroid ( $\mathbf{c}_1 = \mathbf{h}_1$ )

**2: Incremental Iterations:** For each voxel  $(x, y, z) \in \mathbf{R}$ 

- (i) Add the neighborhood vector of the voxel  $\mathbf{g}_{\mathbf{N}_{x,y,z}}$  as  $\mathbf{a}_k = \{a_{i,n} : i = 1, \dots, I_x I_y I_z; n = k\}$  to the data matrix, which compose  $\mathbf{A}_k$ .
- (ii) Perform alternative update of  $\mathbf{W}_k$  and  $\mathbf{h}_k = \{h_{j,n} : j = 1, \dots, J; n = k\}$  iteratively for  $j = 1, \dots, J$  and  $i = 1, \dots, I_x I_y I_z$ :

$$(\mathbf{h}_k)_j \leftarrow (\mathbf{h}_k)_j \frac{(\mathbf{W}_k^T \mathbf{a}_k)_j}{(\mathbf{W}_k^T \mathbf{W}_k \mathbf{h}_k)_j} \quad (38)$$

$$(\mathbf{W}_k)_{ij} \leftarrow (\mathbf{W}_k)_{ij} \frac{(\mathbf{A}_k \mathbf{H}_k^T + \mathbf{a}_k \mathbf{h}_k^T)_{ij}}{\mathbf{W}_k (\mathbf{H}_{k-1} \mathbf{H}_{k-1}^T + \mathbf{h}_k \mathbf{h}_k^T)_{ij}} \quad (39)$$

- (iii) Compute  $dist = \min_{\mathbf{c}_j} \|\mathbf{h}_k - \mathbf{c}_j\| : j \in [1, \dots, J]$ 
  - If  $dist$  is less than a given threshold, update the closest cluster centroid  $\mathbf{c}_j \leftarrow \frac{1}{k}((k-1)\mathbf{c}_j + \mathbf{h}_k)$
  - Otherwise, increase the rank of  $\mathbf{W}_k$  by adding a randomly generated column ( $J = J + 1$ ) and update  $\mathbf{h}_k$  and  $\mathbf{W}_k$  iteratively using (38) and (39)
- (iv) Stop when all voxels are added.  $J$  represents the final number of clusters and  $\mathbf{W}$  represents the new computed basis of the data

**3: Output:** Solve (32) to output  $H$  matrix that represents the new visual appearance of the image
 

---

applied to select the largest component as the final segmentation of the lung field. The proposed method of lung segmentation is outlined in Algorithm 4.

### C ICNMF-based 3D lung segmentation

The INMF approach used voxels-wise incremental learning to determine the number of clusters, which is computationally expensive for large 3D images. Moreover, the number of clusters are highly sensitive to the distance parameter of algorithm, which is directly related to the sparseness and smoothness of the decomposition space  $\mathbf{H}$ . These drawbacks are overcome in [6] by using an introduced below ICNMF, which

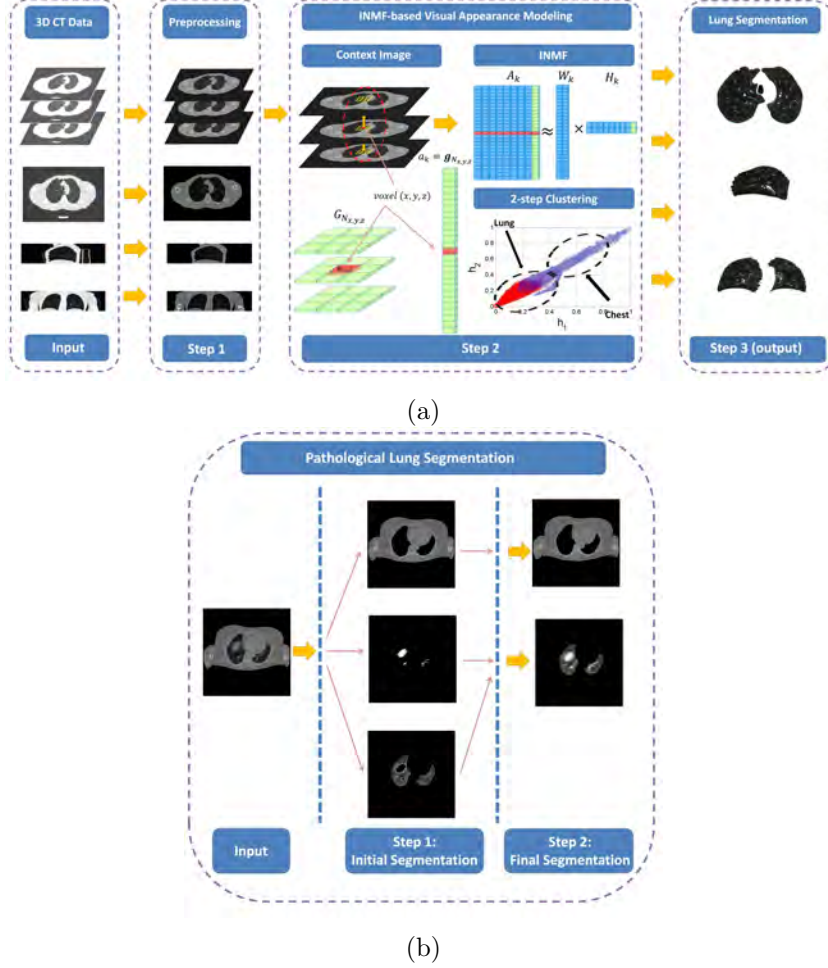


Figure 8. (a) The developed framework for automatic lung segmentation from CT images, (b) 2-step clustering for pathological lung case, where three clusters are detected by INMF method, and two clusters (lung and chest) are extracted [1].

combines basic ideas of the INMF [91] and Constrained NMF (CNMF) [93, 94]. The ICNMF decomposes every large data matrix  $\mathbf{A}$  in a slice-by-slice mode, such that factorization of each next axial CT slice in a 3D CT image is initialized with the basis and decomposition matrices, having been already obtained from all the preceding slices.

To minimize the reconstruction error in Eq.(33) for a large matrix  $\mathbf{A}$  in a computationally feasible way, the INMF forms the goal matrices  $\mathbf{W}$  and  $\mathbf{H}$  incrementally, using the iterative MUR, that at each computational step, converge to the closest local minimum [91]. After each next data vector is added to  $\mathbf{A}$ , the already computed matrices  $\mathbf{W}$  and  $\mathbf{H}$  initialize the next step, thus reducing the overall INMF complexity. However, the INMF by itself does not guarantee smooth and sparse data

---

**Algorithm 4** Automated Lung Segmentation Using INMF-based Visual Appearance Model
 

---

- 1: Preprocess image for removing background
  - 2: Extract INMF-based visual appearance model of the 3D image using Algorithm 3
  - 3: Cluster image voxels into  $J$  groups (calculated in step 2) using the K-means clustering algorithm [92], (as shown in step 1 in Figure 8.(b))
  - 4: Assign smallest cluster centroid in terms of  $\ell_2$ -norm as lung cluster and the maximum centroid as the chest cluster
  - 5: Use K-means to classify data points into lung or chest classes based on the nearest distance to lung and chest centroids (as shown in step 2 in Figure 8(b)).
  - 6: Refine the segmented lung fields using a 3D connected component analysis
- 

representation, obtained by the CNMF due to the constrained matrix factors. The ICNMF combines these main ideas of the INMF and CNMF.

### 1 Incremental Constrained NMF (ICNMF)

Let  $\mathbf{A}_k = [\mathbf{a}_1 \ \mathbf{a}_2 \ \dots \ \mathbf{a}_k]$ ;  $\mathbf{W}_k$ ;  $\mathbf{H}_k$ ;  $F_{\text{re}:k}$ , and  $F_k$ , denote the data matrix with the initial  $k$  data samples, the corresponding basis and decomposition matrices, the reconstruction error, and the constrained reconstruction error, respectively:

$$\begin{aligned}
 F_{\text{re}:k} &= \| \mathbf{A}_k - \mathbf{W}_k \mathbf{H}_k \|^2; \\
 F_k &= F_{\text{re}:k} + \lambda_{\mathbf{W}} \| \mathbf{W}_k \|^2 + \lambda_{\mathbf{H}} \| \mathbf{H}_k \|^2
 \end{aligned}
 \tag{40}$$

The INMF [91] assumes that every new sample,  $\mathbf{a}_{k+1}$ , does not significantly affect the current basis  $\mathbf{W}_k$ , optimized for the previous  $k$  samples, so that their decomposition vectors,  $\mathbf{H}_k$ , need not be updated. Then the first  $k$  columns of  $\mathbf{H}_{k+1}$  remain equal to  $\mathbf{H}_k$ , i.e.,  $\mathbf{H}_{k+1} = [\mathbf{H}_k \ \mathbf{h}_{k+1}]$ , and only the basis,  $\mathbf{W}_{k+1}$  and the last decomposition column vector,  $\mathbf{h}_{k+1}$  have to be updated [91]. To reach a local minimum of the constrained reconstruction error of Eq.(40) after adding the new sample,  $\mathbf{a}_{k+1}$ , these updates:

$$\begin{aligned}
 F_{\text{re}:k+1} &= \| \mathbf{A}_{k+1} - \mathbf{W}_{k+1} \mathbf{H}_{k+1} \|^2; \\
 F_{k+1} &= F_{\text{re}:k+1} + \lambda_{\mathbf{W}} \| \mathbf{W}_{k+1} \|^2 + \lambda_{\mathbf{H}} \| \mathbf{H}_{k+1} \|^2
 \end{aligned}
 \tag{41}$$

are converted into an incremental form separating the previous samples from the new one:

$$\begin{aligned}
 F_{k+1} &\cong \widehat{F}_{\text{re}:k} + \| \mathbf{a}_{k+1} - \mathbf{W}_{k+1} \mathbf{h}_{k+1} \|^2 \\
 &\quad + \lambda_{\mathbf{W}} \| \mathbf{W}_{k+1} \|^2 + \lambda_{\mathbf{H}} \| \mathbf{h}_{k+1} \|^2
 \end{aligned}
 \tag{42}$$

Here,  $\widehat{F}_{\text{re};k}$  is the reconstruction error for the first  $k$  samples, which has been updated for the current  $k + 1$  samples:

$$\begin{aligned}\widehat{F}_{\text{re};k} &= \sum_{i=1}^m \sum_{j=1}^k (a_{ij} - (\mathbf{W}_{k+1} \mathbf{H}_{k+1})_{ij})^2 \\ &\cong \sum_{i=1}^m \sum_{j=1}^k (a_{ij} - (\mathbf{W}_{k+1} \mathbf{H}_k)_{ij})^2\end{aligned}\quad (43)$$

Therefore, after adding the next sample  $\mathbf{a}_{k+1}$ , the conditional minimization of Eq.(42) is rewritten to

$$\begin{aligned}F_{k+1} &= \|\mathbf{A}_k - \mathbf{W}_{k+1} \mathbf{H}_k\|^2 + \\ &\quad \|\mathbf{a}_{k+1} - \mathbf{W}_{k+1} \mathbf{h}_{k+1}\|^2 + \\ &\quad \lambda_{\mathbf{W}} \|\mathbf{W}_{k+1}\|^2 + \\ &\quad \lambda_{\mathbf{H}} (\|\mathbf{H}_k\|^2 + \|\mathbf{h}_{k+1}\|^2)\end{aligned}\quad (44)$$

optimizes the basis,  $\mathbf{W}_{k+1}$ , and adds the new column,  $\mathbf{h}_{k+1}$ , to the decomposition matrix,  $\mathbf{H}_{k+1} = [\mathbf{H}_k \ \mathbf{h}_{k+1}]$ . A conditional local minimum of the error in Eq.(44) is found by a gradient-like iterative search [95]:

$$\begin{aligned}(\mathbf{h}_{k+1})_q &\leftarrow (\mathbf{h}_{k+1})_q - \alpha_q \frac{\partial F_{k+1}}{\partial (\mathbf{h}_{k+1})_q}; \\ (\mathbf{W}_{k+1})_{iq} &\leftarrow (\mathbf{W}_{k+1})_{iq} - \beta_{iq} \frac{\partial F_{k+1}}{\partial (\mathbf{W}_{k+1})_{iq}};\end{aligned}\quad (45)$$

$$q = 1, \dots, r; \quad i = 1, \dots, m$$

where  $\alpha_q$  and  $\beta_{iq}$  are specific steps for updating the elements  $(\mathbf{h}_{k+1})_q$  and  $(\mathbf{W}_{k+1})_{iq}$ , respectively, and the partial derivatives follow from Eq.(45):

$$\begin{aligned}\frac{\partial F_{k+1}}{\partial \mathbf{h}_{k+1}} &= -2\mathbf{W}_{k+1}^\top (\mathbf{a}_{k+1} - \mathbf{W}_{k+1} \mathbf{h}_{k+1}) \\ &\quad + 2\lambda_{\mathbf{H}} \mathbf{h}_{k+1}; \\ \frac{\partial F_{k+1}}{\partial \mathbf{W}_{k+1}} &= -2(\mathbf{A}_k - \mathbf{W}_{k+1} \mathbf{H}_k) \mathbf{H}_k^\top \\ &\quad - 2(\mathbf{a}_{k+1} - \mathbf{W}_{k+1} \mathbf{h}_{k+1}) \mathbf{h}_{k+1}^\top \\ &\quad + 2\lambda_{\mathbf{W}} \mathbf{W}_{k+1}\end{aligned}\quad (46)$$

As shown in [95], the required adaptive steps result in the multiplicative updates, ensuring the factors  $\mathbf{W}_k$  and  $\mathbf{H}_k$ , which initially (for  $k = 1$ ) were nonnegative, remain

nonnegative at every updating iteration, and hence at every step,  $k = 2, \dots, n$ : for  $q = 1, \dots, r$  and  $i = 1, \dots, m$

$$\begin{aligned}
\alpha_q &= \frac{(\mathbf{h}_{k+1})_q}{(\mathbf{W}_{k+1}^\top \mathbf{W}_{k+1} \mathbf{h}_{k+1})_q}; \\
(\mathbf{h}_{k+1})_q &\leftarrow (\mathbf{h}_{k+1})_q \frac{(\mathbf{W}_{k+1}^\top \mathbf{a}_{k+1})_q}{(\mathbf{W}_{k+1}^\top \mathbf{W}_{k+1} \mathbf{h}_{k+1} + \lambda \mathbf{H} \mathbf{h}_{k+1})_q}; \\
\beta_{iq} &= \frac{(\mathbf{W}_{k+1})_{iq}}{S_{iq}}; \\
(\mathbf{W}_{k+1})_{iq} &\leftarrow (\mathbf{W}_{k+1})_{iq} \frac{(\mathbf{A}_k \mathbf{H}_k^\top + \mathbf{a}_{k+1} \mathbf{h}_{k+1}^\top)_{iq}}{S_{iq}}
\end{aligned} \tag{47}$$

where  $S_{iq} = (\mathbf{W}_{k+1} \mathbf{H}_k \mathbf{H}_k^\top + \mathbf{W}_{k+1} \mathbf{h}_{k+1} \mathbf{h}_{k+1}^\top + \lambda \mathbf{W} \mathbf{W}_{k+1})_{iq}$  and  $\mathbf{W}_{k+1}$  is initialized with  $\mathbf{W}_k$ , when the new sample  $\mathbf{a}_{k+1}$  is added. The updates of Eq.(47) provably guarantee that iterations at every step  $k$  converge to a local minimum of the reconstruction error [21, 95].

Computational complexity of the ICNMF (like the INMF) is  $O(mr^2)$  per iteration, comparing with  $O(nmr)$  for the NMF with the running time depending linearly on the number  $n$  of samples. The independence of the running time from the number of samples, makes the ICNMF and INMF more suitable for learning the matrix factors  $\mathbf{W}$  and  $\mathbf{H}$  for representing a large collection of data samples. Moreover, as follows from Eq.(47), the computations for the learning process can be simplified by taking into account that both matrices  $\mathbf{A}_k$  and  $\mathbf{H}_k$  do not change after adding every new data sample  $\mathbf{a}_{k+1}$ . Thus, instead of keeping separately the growing matrices  $\mathbf{A}_k$  and  $\mathbf{H}_k$ , only their fixed-size products  $\mathbf{A}_k \mathbf{H}_k^\top$  and  $\mathbf{H}_k \mathbf{H}_k^\top$  have to be stored.

The above updating process holds (with mostly notational changes), if the data matrix  $\mathbf{A}_k$  is appended at each step  $k + 1$  with not a single data vector, but a small-size matrix  $\mathbf{a}_{k+1}$ , e.g., a context image for the next 2D slice in the lung segmentation case.

## 2 ICNMF-based visual appearance modeling

The proposed lung segmentation in a 3D CT chest image is outlined in Figure 9 and Algorithm 5: (i) removing an image background by conventional 3D region growing (Figure 6); (ii) modeling visual appearance of the remaining chest-lung image with the ICNMF, and (iii) extracting 3D lung voxels by data clustering and cleaning the region map. The last two stages are detailed below.

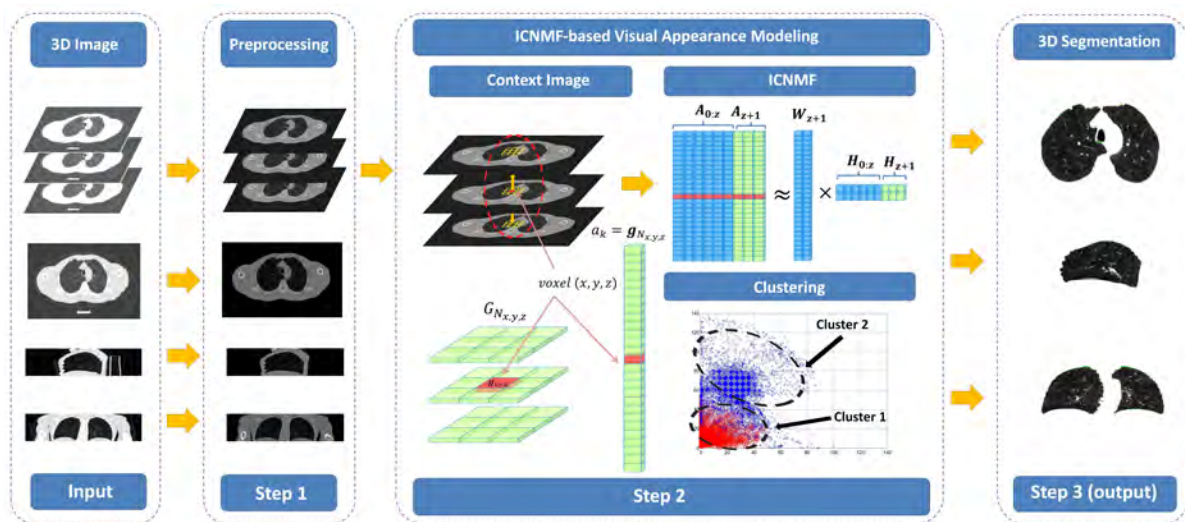


Figure 9. The developed framework for 3D lung segmentation based on ICNMF [6].

---

**Algorithm 5** 3D Lung Segmentation by ICNMF-based Visual Appearance Modeling.

---

- 1: Preprocessing: Remove a background of an input 3D CT image  $\mathbf{g}$ .
  - 2: Apply Algorithm 6 to the remaining 3D context image to describe its visual appearance by  $r$ -dimensional voxel-wise decomposition vectors in matrix  $\mathbf{H}$ .
  - 3: Assign the voxel descriptors to a prescribed number,  $K$ , of objects by the  $K$ -means clustering [89].
  - 4: Discriminate between the lung and chest clusters by characterizing their relative brightness with the Frobenius norms of their centroids in  $\mathbf{H}$ -space.
  - 5: Refine the segmented 3D lung regions by analyzing 3D connected components.
- 

To model the visual appearance of objects-of-interest with the ICNMF, the context image [96] is built from the original 3D image  $\mathbf{g} = \{g_{x,y,z} : (x, y, z) \in \mathbb{R}; g_{x,y,z} \in \mathbb{Q}\}$ . Here,  $\mathbb{R} = \{(x, y, z) : x = 0, \dots, X; y = 0, \dots, Y; z = 0, \dots, Z\}$  is a finite arithmetic lattice supporting 3D digital images and their region maps, and  $\mathbb{Q}$  is a finite set of integer voxel-wise intensities, or gray values. Each voxel  $(x, y, z)$  of the context image is described with the context vector  $\mathbf{a}$ , which contains the intensities for this voxel and its nearest 3D neighbors in the original image  $\mathbf{g}$ , e.g., the 27 intensities in total for the nearest  $3 \times 3 \times 3$  neighborhood.

To minimize the constrained reconstruction error of Eq.(42) after adding the context vector  $\mathbf{a}_{k+1}$  for every next voxel  $k + 1$ , the details in Section 1, ICNMF uses the

---

**Algorithm 6** The ICNMF-based Visual Appearance Modeling
 

---

**1: Initialization**

- (i) Given an image  $\mathbf{g}$ , form the data matrix  $\mathbf{A}_{z=0}$  from the context vectors of the first slice  $z = 0$ .
- (ii) Set the number  $r$  of data clusters characterizing visual appearance of the image.
- (iii) Initialize randomly the factors  $\mathbf{W}_0$  and  $\mathbf{H}_0$ .
- (iv) Update  $\mathbf{H}_0$  and  $\mathbf{W}_0$  until convergence:

$$\begin{aligned}
 (\mathbf{H}_0)_{qj} &\leftarrow (\mathbf{H}_0)_{qj} \frac{(\mathbf{W}_0^\top \mathbf{A}_0)_{qj}}{(\mathbf{W}_0^\top \mathbf{W}_0 \mathbf{H}_0 + \lambda_{\mathbf{H}} \mathbf{H}_0)_{qj}}; \\
 (\mathbf{W}_0)_{iq} &\leftarrow (\mathbf{W}_0)_{iq} \frac{(\mathbf{A}_0 \mathbf{H}_0^\top)_{iq}}{(\mathbf{W}_0 (\mathbf{H}_0 \mathbf{H}_0^\top) + \lambda_{\mathbf{W}} \mathbf{W}_0)_{iq}}
 \end{aligned} \tag{48}$$

- (v) Set  $\mathbf{A}_{0:0} = \mathbf{A}_0$  and  $\mathbf{H}_{0:0} = \mathbf{H}_0$ .

**2: Slice-wise increments:** For each slice  $z = [1, \dots, Z]$ ,

- (i) Form the matrix  $\mathbf{A}_z$  from the context vectors of the slice  $z$  and extend the joint data matrix  $\mathbf{A}_{0:z} = [\mathbf{A}_{0:z-1} \ \mathbf{A}_z]$ .
- (ii) Initialize the slice-wise decomposition matrix  $\mathbf{H}_z$  randomly and extend the joint decomposition matrix  $\mathbf{H}_{0:z} = [\mathbf{H}_{0:z-1} \ \mathbf{H}_z]$ .
- (iii) Initialize the basis matrix  $\mathbf{W}_z = \mathbf{W}_{z-1}$  and iteratively update  $\mathbf{H}_z$  and  $\mathbf{W}_z$  until convergence:

$$\begin{aligned}
 (\mathbf{H}_z)_{qj} &\leftarrow (\mathbf{H}_z)_{qj} \frac{(\mathbf{W}_z^\top \mathbf{A}_z)_{qj}}{(\mathbf{W}_z^\top \mathbf{W}_z \mathbf{H}_z + \lambda_{\mathbf{H}} \mathbf{H}_z)_{qj}}; \\
 (\mathbf{W}_z)_{iq} &\leftarrow (\mathbf{W}_z)_{iq} \frac{(\mathbf{A}_{0:z-1} \mathbf{H}_{0:z-1}^\top + \mathbf{A}_z \mathbf{H}_z^\top)_{iq}}{(\mathbf{W}_z (\mathbf{H}_{0:z-1} \mathbf{H}_{0:z-1}^\top + \mathbf{H}_z \mathbf{H}_z^\top) + \lambda_{\mathbf{W}} \mathbf{W}_z)_{iq}}
 \end{aligned} \tag{49}$$

**3: Output:** The joint decomposition matrix  $\mathbf{H}_{0:Z}$  describing visual appearance of the image  $\mathbf{g}$  as a mixture of the  $r$  clusters specified by the basis matrix  $\mathbf{W}_Z$ .
 

---

multiplicative algorithm of Eq.(47) that converges iteratively to the goal descriptor  $\mathbf{h}_{k+1}$  of this voxel and the updated basis matrix  $\mathbf{W}$ . However, a very large size of the 3D CT image makes repetitive computations for all the image voxels too expensive. To reduce the computational complexity, all the voxels of every CT slice are added to the matrix  $\mathbf{A}$  at the same time, while the slices are processed sequentially. The above ICNMF algorithm remains almost the same, apart of considering  $\mathbf{a}_{k+1}$  and  $\mathbf{h}_{k+1}$  as matrices, rather than vectors ( $\mathbf{A}_z$  and  $\mathbf{H}_z$ ). In this case, the already computed optimal basis  $\mathbf{W}_z$  of the previous  $z$  slices, initializes updating the basis  $\mathbf{W}_{z+1}$  for the  $z + 1$  slices. Algorithm 6 outlines the proposed version of the ICNMF.

### 3 Segmentation

After modeling visual appearance with the ICNMF, the  $K$ -means clustering is applied to the voxels in the  $\mathbf{H}_{0:Z}$ -space in order to form a prescribed number,  $K$ , of

data clusters (in this case  $K = 2$ : the lungs and the chest tissues). The Frobenius norm of each cluster’s centroid helps to identify the goal lungs, by their relative brightness in the image: the brighter the voxel and its neighborhood, the farther their descriptor from the  $\mathbf{H}$ -space origin, so that the darker (low-intensity) contexts have the smaller norms. Then the 3D connected component analysis refines the segmentation by removing isolated voxels from another cluster inside each large segmented region.

To highlight capabilities of the ICNMF in revealing characteristic inter-voxel dependencies, Figure 10 compares the voxel distribution in the original 27-dimensional space for the  $3 \times 3 \times 3$  voxel neighborhoods to the same distribution in the reduced  $r$ -dimensional decomposition space ( $\mathbf{H}$ -space) formed by the ICNMF. Since the lung voxels are much better separated from the chest voxels in the latter space, it yields more accurate data clustering and segmentation.

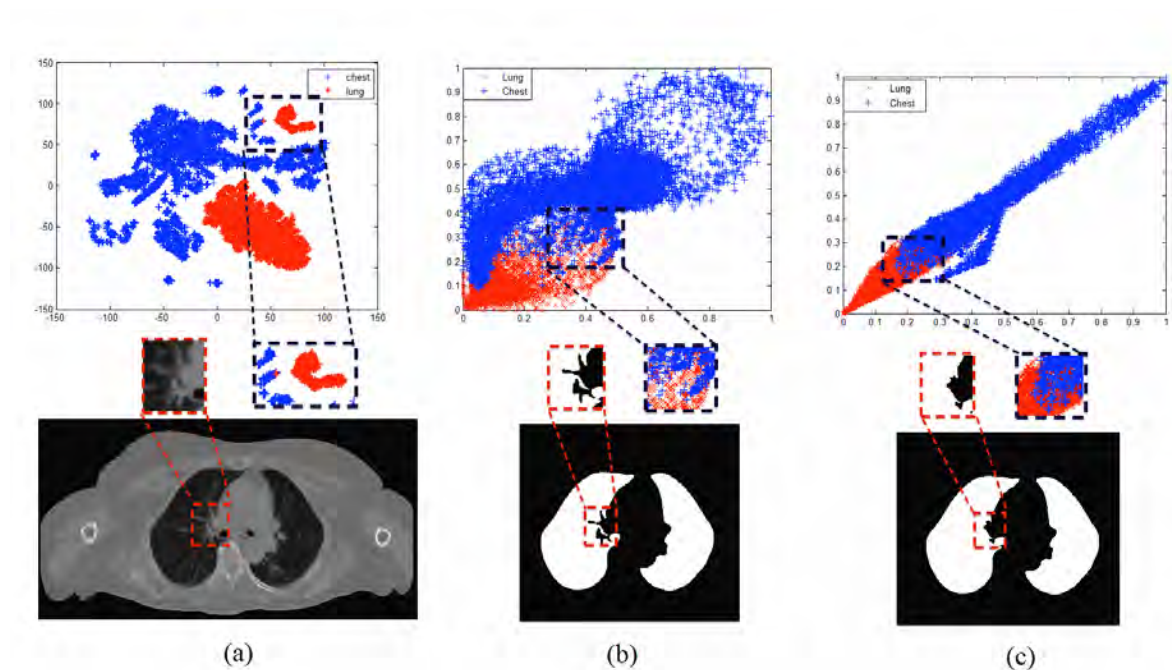


Figure 10. Signal distributions for segmented voxels in the original 27-dimensional space (a) and the  $r$ -dimensional spaces reduced with the NMF (b) and ICNMF (c) visualized using the t-SNE projection [7]. The better ICNMF performance is exemplified by signal distributions and segmentation results for pathologies on the lung-chest border [6].

## D Experiments with synthetic and real data

This section starts with evaluating ICNMF method and compare the performance of three developed methods at the end. The developed algorithms in the previous sections were evaluated on both real (*in-vivo*) and synthetic data using three common performance metrics: (*i*) the Dice Similarity Coefficient (DSC) [97], (*ii*) the Modified 95-percentile Hausdorff distance (MHD) [98], and (*iii*) the Absolute Lung Volume Difference (ALVD). Synthetic 3D phantoms [85] simplify initial performance tests, because accurate lung borders on real CT images are very difficult to obtain manually, due to the observers’ variability. The phantom images mimic visual appearance of the real 3D CT data by Gibbs sampling of a learned generalized 3D Gauss-Markov random field model [99].

The *in-vivo* CT image data sets for 17 patients have been acquired with a multi-detector GE Light Speed Plus scanner (General Electric, Milwaukee, USA) using the following scanning protocol: the 2.5mm-thick slices reconstructed every 5 mm; the scanning pitch of 1.5; 140 KV; 100 MA; and F.O.V of 36 cm and the size of  $512 \times 512 \times 390$  voxels of each 3D test data set. The CT image segmentation separated two objects: the darker lung tissues and the brighter chest tissues.

Table 1 explores impacts of the sparseness and smoothness constraints of Eq.(40) on the segmentation accuracy in the DSC terms for a real data set, showing the decreased accuracy for the INMF-based modeling, i.e., for zero constraint weights ( $\lambda_{\mathbf{W}} = \lambda_{\mathbf{H}} = 0$ ) in Eq.(40). The ICNMF-based modeling resulted in the more accurate

TABLE 1. Segmentation accuracy for different weights in Eq.(40).

$\lambda_{\mathbf{W}} \lambda_{\mathbf{H}}$	0	0.1	0.3	1	3	10	30	100	300
0	0.821	0.930	0.892	0.898	0.961	0.955	0.947	0.941	0.955
0.1	0.850	0.942	0.940	0.935	0.953	0.936	0.928	0.953	0.955
0.3	0.839	0.953	0.952	0.940	0.953	0.953	0.953	0.953	0.959
1	0.847	0.953	0.953	0.953	0.952	0.947	0.953	0.958	0.962
3	0.851	0.953	0.952	0.951	0.953	0.953	0.952	0.953	0.957
10	0.848	0.953	0.951	0.953	0.952	0.952	0.953	0.953	0.955
30	0.848	0.952	0.952	0.953	0.953	0.953	0.954	0.951	0.955
100	0.849	0.949	0.953	0.952	0.953	0.952	0.953	0.952	0.957

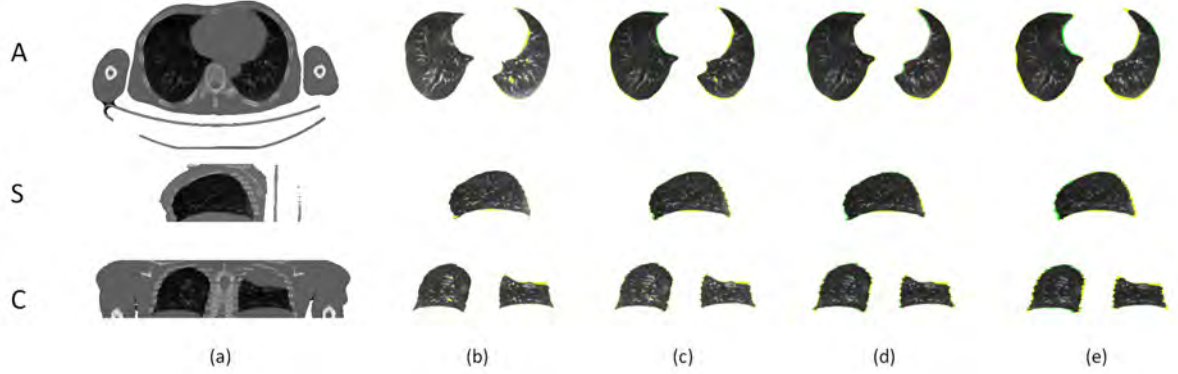


Figure 11. Original 2D CT slices (a) and 3D lungs segmented using  $3 \times 3 \times 3$  (b),  $7 \times 7 \times 3$  (c),  $11 \times 11 \times 3$  (d), and  $15 \times 15 \times 3$  (e) neighborhood size, and projected onto the axial (A), sagittal (S), and coronal (C) 2D planes for visualization.

segmentation, as the sparseness of  $\mathbf{H}$  affecting the accuracy more, than the smoothness of  $\mathbf{W}$ . At the same time, the accuracy varies insignificantly for many weight combinations, so that selecting the best pair requires a too long experimentation. Based on a few additional experiments, the weights have been set in the experiments to  $\lambda_{\mathbf{W}} = 1$  and  $\lambda_{\mathbf{H}} = 100$ .

The segmentation accuracy was also tested for different numbers,  $r \in \{2, 4, 6, 8\}$ , of the basis vectors, the best result having been achieved for  $r = 4$ . Obviously, the accuracy depends also on the neighborhood size and shape. Comparative experiments with the  $(3 \times 3 \times 3)$ ,  $(7 \times 7 \times 3)$ ,  $(11 \times 11 \times 3)$ , and  $(15 \times 15 \times 3)$  voxel neighborhoods in Figure 11 have shown that the more expanded the neighborhood, the lesser the segmentation accuracy. The increasing segmentation errors highlighted in green and yellow can be explained in part by higher similarity between the larger neighborhoods for the adjacent voxels along the lung-chest boundary. Also, a number of the CT scans of patients with different lung diseases [100] were segmented in order to evaluate the performance of Algorithm 5 in the case of severe lung pathologies. The eight CT scans selected in Figure 12 demonstrate diverse pulmonary patterns, such as, e.g., caused by airspace or diffuse consolidation; cancer; different nodules, including juxtapleural ones, etc. Algorithm 5 adapts successfully to such pathologies.

Table 2 confirms the higher accuracy of Algorithm 5 based on the ICNMF w.r.t.

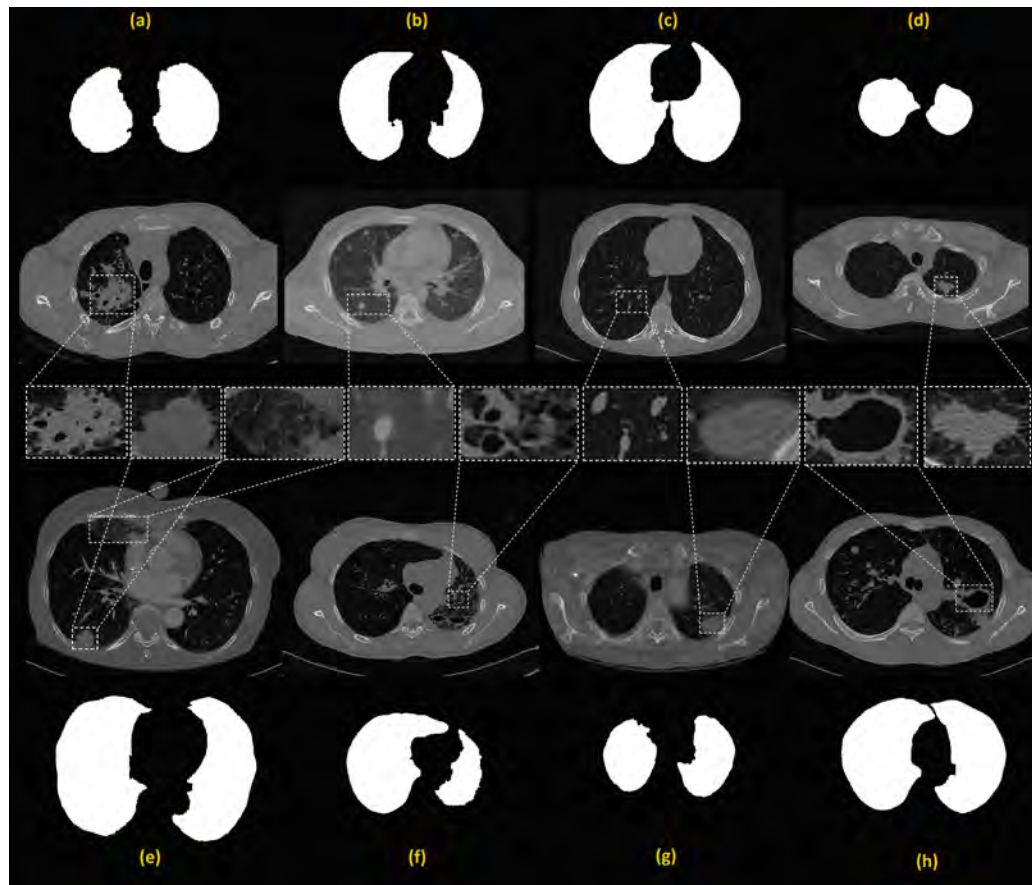


Figure 12. Lungs segmented with Algorithm 5 using the  $3 \times 3 \times 3$  neighborhood on pulmonary CT in the cases of airspace consolidation (a); tree-in-bud and micro-nodules (b); usual nodules (c); cancer (d); ground-glass opacity and juxtapleural nodules (e); honeycomb (f); diffuse consolidation (g), and cavity (h) [6].

TABLE 2. Comparative segmentation accuracy (DSC) of Algorithm 5 using the  $(3 \times 3 \times 3)$  context w.r.t. five other algorithms on the *in vivo* data sets.

Algorithm	Mean $\pm$ std	<i>p</i> -value
Algorithm 5	0.969 $\pm$ 0.010	
Segm-Int [101]	0.632 $\pm$ 0.091	0.0001
Segm-Int+S [101]	0.783 $\pm$ 0.078	0.0001
IT [76]	0.816 $\pm$ 0.091	0.0001
MRS [102]	0.613 $\pm$ 0.054	0.0001
GVF [103]	0.848 $\pm$ 0.087	0.0003

TABLE 3. Accuracy of the ICNMF-based Algorithm 5 with the  $(3 \times 3 \times 3)$  contexts w.r.t. the INMF-based [1] and NMF-based [2] algorithms on synthetic and real (*in vivo*) data sets: mean $\pm$ std [*p*-value].

	DSC		ALVD		MHD	
	Real data	Synthetic data	Real data	Synthetic data	Real data	Synthetic data
ICNMF	<b>0.96</b> $\pm$ 0.01	<b>0.97</b> $\pm$ 0.01	<b>0.87</b> $\pm$ 0.62	<b>0.51</b> $\pm$ 0.07	<b>9.0</b> $\pm$ 0.001	<b>4.8</b> $\pm$ 0.006
INMF [1]	0.95 $\pm$ 0.02 [0.037]	0.96 $\pm$ 0.02 [0.76]	2.2 $\pm$ 1.2 [0.0004]	2.2 $\pm$ 1.7 [0.040]	9.5 $\pm$ 0.003 [ $< 0.0001$ ]	5.7 $\pm$ 0.030 [ $< 0.0001$ ]
NMF [2]	0.95 $\pm$ 0.02 [0.027]	0.96 $\pm$ 0.01 [0.50]	2.4 $\pm$ 1.1 [ $< 0.0001$ ]	2.4 $\pm$ 1.1 [0.004]	9.7 $\pm$ 0.010 [ $< 0.0001$ ]	5.9 $\pm$ 0.005 [ $< 0.0001$ ]

five other segmentation algorithms, by comparing the means and standard deviations of their DSC values using the statistical paired *t*-tests. The latter algorithms account for only signal intensities (the abbreviation Segm-Int) or combined intensity and spatial information (Segm-Int+S) [101]; perform Iterative Thresholding (IT) followed by a sequence of morphological operations [76] or Multiple Resolution Segmentation (MRS) [102], or evolve a deformable boundary guided by the Gradient Vector Flow (GVF) [103].

Table 3 compares the proposed ICNMF-based Algorithm 5 with two developed NMF- and INMF-based segmentation algorithms [1, 2] on the 17 real and 7 synthetic data sets, using the three aforementioned performance metrics. By the DSC accuracy, all three algorithms differ insignificantly for the synthetic data, whereas for the real data, Algorithm 5 demonstrates a small, but statistically significant improvement (the DSC 0.96 $\pm$ 0.01 vs. 0.95 $\pm$ 0.02). At the same time, by the ALVD and MHD accuracy, the proposed algorithm outperforms the other two algorithms both on the real and synthetic data sets.

The sensitivity of the proposed segmentation algorithm against selecting its weights, the number of the basis vectors, and the neighborhood size, was evaluated

using Receiver Operating Characteristic (ROC) statistics, in particular, the area under the ROC curve (AUC). The AUC for  $\lambda_{\mathbf{H}}$ ,  $\lambda_{\mathbf{W}}$ ,  $r$  and the neighborhood size ( $\mathbf{N}_{x,y,z}$ ) was, respectively, 0.98, 0.98, 0.97, and 0.99, demonstrating good performance and low sensitivity of the algorithm. Also, as was shown experimentally, different arrangements of the 3D context voxels in the data vector  $\mathbf{a}$ , do not affect the segmentation results.

TABLE 4. Overlaps with the true left (LL) and right lungs (RL) for Algorithm 5 (A5) w.r.t. a human expert and the most (rank 1), median (rank 8), and least accurate (rank 15) LOLA11 segmentation of the 55 data sets : the mean, standard deviation (std), minimum (min), 25%-quartile (Q1), median (med), 75%-quartile (Q3), and maximum (max) overlaps.

Algorithm	Object	mean	std	min	Q1	med	Q3	max
A5[rank 5]	Average score:	0.965						
	LL	0.965	0.108	0.205	0.981	0.988	0.992	0.998
	RL	0.964	0.133	0.010	0.982	0.988	0.991	0.997
Human	Average score:	0.984						
	LL	0.984	0.031	0.782	0.987	0.992	0.996	0.998
	RL	0.984	0.047	0.662	0.988	0.995	0.997	0.999
Rank 1 [104]	Average score:	0.973						
	LL	0.974	0.097	0.277	0.987	0.992	0.995	0.999
	RL	0.972	0.135	0.000	0.991	0.994	0.996	0.999
Rank 7 [100]	Average score:	0.955						
	LL	0.957	0.137	0.034	0.979	0.987	0.995	0.999
	RL	0.952	0.151	0.000	0.984	0.990	0.997	0.999
Rank 13 [105]	Average score:	0.939						
	LL	0.929	0.154	0.083	0.945	0.974	0.983	0.995
	RL	0.950	0.121	0.150	0.960	0.978	0.988	0.994

To demonstrate its applicability to the data collected by various scanning protocols, Algorithm 5 was also tested on 55 real chest 3D CT scans provided by the Lobe and Lung Analysis 2011 (LOLA11) challenge ([www.lola11.com](http://www.lola11.com)) and acquired at different places with several scanners, scanning protocols, and reconstruction parameters. To evaluate the results using the LOLA11 dataset, the trachea and main bronchi are removed, and if needed, separated the lung by finding a maximum cost path in connected axial slices as in [106]. Table. 4 presents the results in comparison with the best, median, and worst results for the 13 lung segmentation algorithms participated in the LOLA11 challenge for 2011 - 2014. Selected examples of the lung

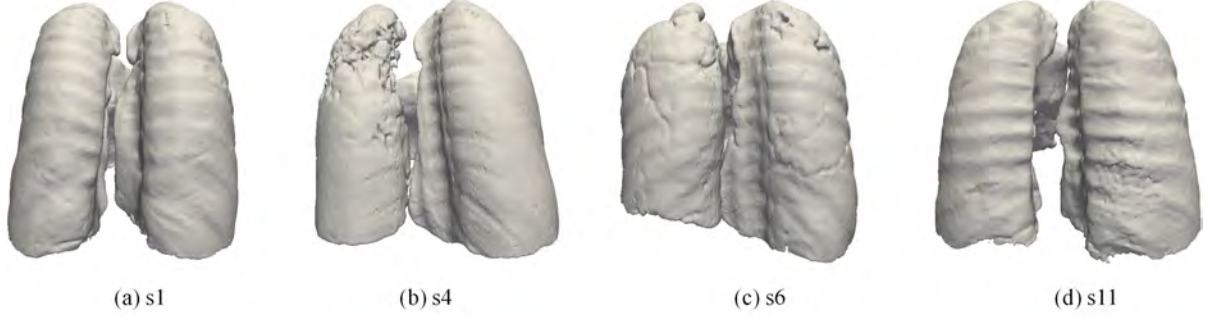


Figure 13. 3D visualization of segmented lungs from selected LOLA11 subjects (“s” indicates the scan).

region maps obtained by Algorithm 5 are shown in Figure 14. To further demonstrate the algorithm’s performance, Figure 13 visualizes the segmented 3D lungs for selected subjects.

It should be noted that several pathologies in the LOLA11 data set are far outside capabilities of the proposed simple visual appearance descriptors accounting for only the nearest-neighbor relations of the voxels. Accurate segmentation of such pathological lungs requires much more profound lung and chest models. Moreover, there is no consensus of the medical imaging community on whether the pleural fluid should be considered as a part of the lung field as it is done in the LOLA11 ground truth [100]. Because our ICNMF-based Algorithm 5 does not include the pleural fluid to the lungs, by the overall accuracy of 0.965 (the relative overlap with the ground truth) it has the 5<sup>th</sup> rank among all the LOLA11 contestants. However, without the relevant nine pathological subjects from the LOLA11 data set it achieves the top-rank accuracy of 0.986 for the remaining 46 subjects.

In terms of the algorithm complexity, the proposed ICNMF-based segmentation extracts voxel-wise features in a completely unsupervised mode, using only a few parameters, such as, e.g.,  $\lambda_{\mathbf{W}}$ ,  $\lambda_{\mathbf{H}}$ ,  $r$ , and  $N_{x,y,z}$ , whereas the conventional top-ranked techniques [100, 104, 107] comprise specific feature engineering steps, which require proper initialization and parameter tuning.

## E Conclusions

The experiments with both *in-vivo* and synthetic images confirmed that the proposed ICNMF-based 3D lung segmentation outperforms NMF- and INMF-based segmentation algorithms by its DSC-, ALVD-, and MHD-accuracy. Its better performance stems from a more accurate compressed description of characteristic spatial signal dependencies in every input image, by revealing smooth features (columns of the basis nonnegative matrix  $\mathbf{W}$ ) and encoding them with sparse descriptors (columns of the nonnegative decomposition matrix  $\mathbf{H}$ ) of a corresponding context image. To make the description computationally feasible for a typically very large 3D CT image, the ICNMF combines the conventional INMF and CNMF, i.e., the optimal basis and decomposition matrices are estimated incrementally, while their constrained Frobenius norms enforce their smoothness and sparseness, respectively. Testing on the 3D chest CT images provided by the LOLA11, collected by different scanners, scanning protocols, and reconstruction parameters, indicated that the proposed algorithm is scored sufficiently high among 13 other state-of-the-art methods.

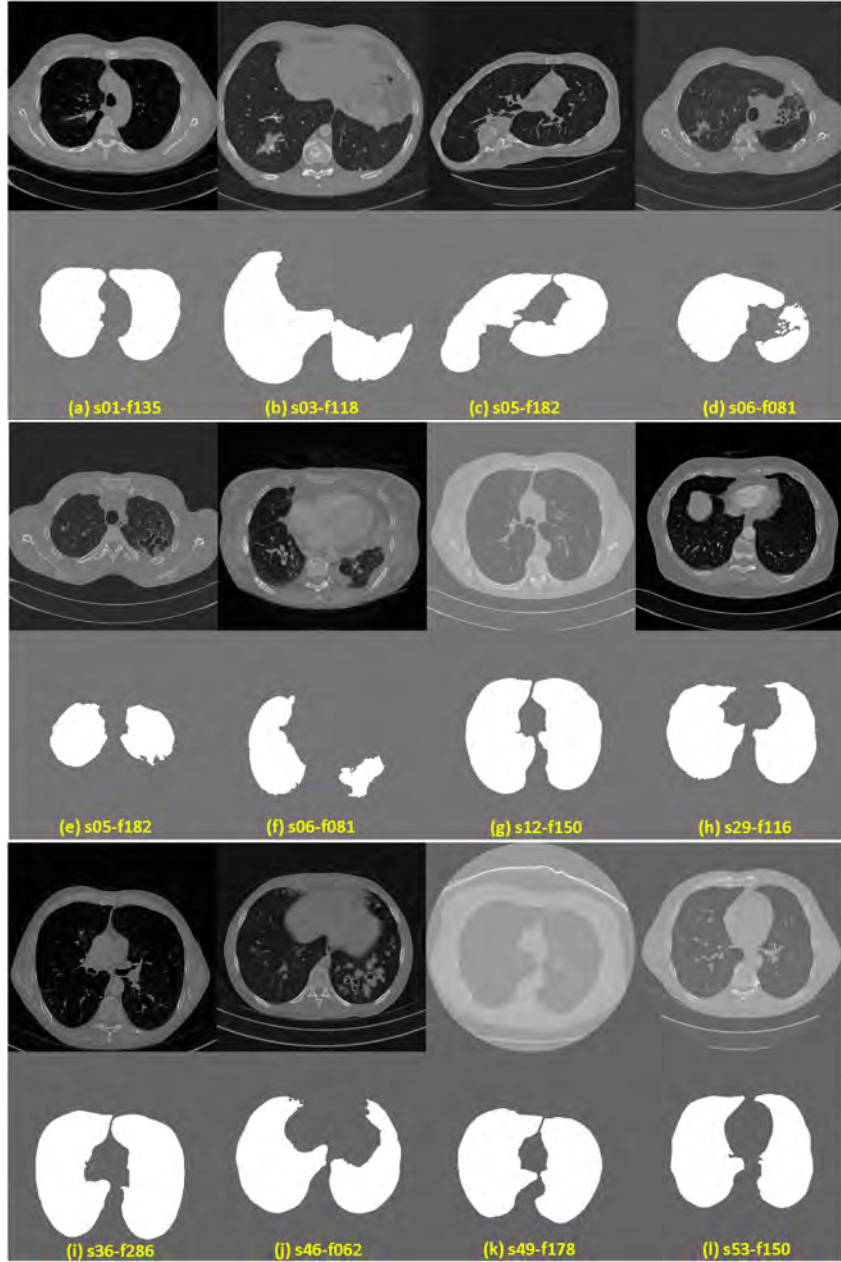


Figure 14. 3D lungs segmented with Algorithm 5 using the  $(3 \times 3 \times 3)$  context: the CT slice (“s” and “f” indicate the scan and slice numbers, respectively) vs. the corresponding lung region map (white) in the LOLA11 dataset [6].

## CHAPTER V

### PART-BASED REPRESENTATION FOR DEEP LEARNING

This chapter will demonstrate how to achieve a meaningful representation from data that discovers the hidden structure of high-dimensional data based on autoencoders [12, 16, 108] and convolutional autoencoder [58]. Inspired by the idea of sparse coding [20, 53, 109] and NMF [13, 110], learning features that exhibit sparse part-based representation of data is expected to disentangle the hidden structure of data. It will be shown that these features result in good generalization ability for the trained model, and improve the reconstruction error.

Using NMF, the features and the encoding of data are forced to be nonnegative, which results in part-based additive representation of data. However, while sparse coding within NMF needs an expensive optimization process to find the encoding of test data, this process is relatively fast in autoencoders [47]. Therefore, training an autoencoder which could exploit the benefits of part-based representation using nonnegativity is expected to improve the performance of a deep learning network.

In Section A, a new approach is developed to train an autoencoder by introducing a nonnegativity constraint into its learning, in order to learn a sparse, part-based representation of data. The training is then extended in Section B to train a deep network with stacked autoencoders and a softmax classification layer, while constraining the weights of the network to be nonnegative. The goal is two-fold: part-based representation in the autoencoder network to improve its ability to disentangle the hidden structure of the data, and producing a better reconstruction of the data. It is shown that these criteria improves the prediction performance of a deep learning network.

Convolutional net (Convnet) [4] have shown to be powerful models in extracting

rich features from high-dimensional images. They employ hierarchical layers of combined convolution and pooling to extract compressed features that capture the intra-class variations between images. The purpose of applying pooling over neighbor activations in featuremaps of Convnet is to break the spatial correlation of neighboring pixels, and to improve the scale and translation invariant features learned by Convnet. This also helps in learning filters for generic feature extraction of low-mid-high level of concepts, such as edge detectors, geometric shapes, and object class [111–114]. Several regularization techniques have been proposed to improve feature extraction in Convnet and to overcome overfitting in large deep networks with many parameters. A dropout technique in [115] is based on randomly dropping hidden units with its connection during training to avoid co-adaptation or redundant filter training. This method resemble averaging over ensemble of sub-models, where each sub-model is trained based on a subset of parameters. A maxout neuron is proposed in [116] while a maxout neuron, is taking the maximum activity across featuremaps of Convnet, similar to max-pooling. Maxout networks have shown to improve the classification performance by building a convex an unbounded activation function, which prevents learning dead filters (delta-shape filters). A winner-take-all method is employed in [59] to reduce or eliminate redundant and delta type filters in pretraining of Convnet using CAE, by taking the maximum activity inside featuremap in each training step.

Bach et al. [117] organize  $\ell_1$  sparsity in a structured form to capture interpretable features and improve prediction performance of the model. In this work, a novel Structured Model of sparse feature extraction in CAE that improves the performance of feature extraction by regularizing the distribution of activities inside and across featuremaps. In Section D, the idea of sparse filtering [9] is employed, to regularize the activity across featuremaps and to improve sparsity within and across featuremaps. The model uses  $\ell_2$  and  $\ell_1$  normalization, as in [9], on the featuremap activations to implement part-based feature extraction.

## A Nonnegativity Constrained Autoencoder (NCAE)

Ideally, part-based representation is implemented through decomposing data into parts, which produce the original data when combined. However, the combination of parts here is only allowed to be additive [13]. As shown in [118] that demonstrates part-based representation in neural networks, the input data can be decomposed in each layer into parts, while the weights in  $\mathbf{W}$  are constrained to be nonnegative [118].

Intuitively, to improve the performance of the autoencoder in terms of reconstruction of input data, it should be able to decompose data into parts which are sparse in the encoding layer, and then combine them in an additive manner in the decoding layer. To achieve this goal, a nonnegativity constraint is imposed on the connecting weights  $\mathbf{W}$ . This means that the column vectors of  $\mathbf{W}$  are coerced to be sparse, i.e. only a small portion of entries per column is non-zero.

To encourage nonnegativity in  $\mathbf{W}$ , the weight decay term in Eq.(28) is replaced and a quadratic function [119, 120] is used. This results in the following cost function for NCAE:

$$J_{\text{NCAE}}(\mathbf{W}, \mathbf{b}) = J_{\text{E}}(\mathbf{W}, \mathbf{b}) + \beta J_{\text{KL}}(p \parallel \hat{p}) + \frac{\alpha}{2} \sum_{l=1}^2 \sum_{i=1}^{s_l} \sum_{j=1}^{s_{l+1}} f(w_{ij}^{(l)}) \quad (50)$$

where

$$f(w_{ij}) = \begin{cases} w_{ij}^2 & w_{ij} < 0 \\ 0 & w_{ij} \geq 0 \end{cases} \quad (51)$$

and  $\alpha \geq 0$ . Minimization of Eq.(50) would result in reducing the average reconstruction error, increased sparsity of hidden layer activations, and reduced number of nonnegative weights of each layer. To update the weights and biases, the gradient of Eq.(50) used in the backpropagation algorithm is computed:

$$w_{ij}^{(l)} = w_{ij}^{(l)} - \eta \frac{\partial}{\partial w_{ij}^{(l)}} J_{\text{NCAE}}(\mathbf{W}, \mathbf{b}) \quad (52)$$

$$b_i^{(l)} = b_i^{(l)} - \eta \frac{\partial}{\partial b_i^{(l)}} J_{\text{NCAE}}(\mathbf{W}, \mathbf{b}) \quad (53)$$

where  $\eta > 0$  is the learning rate. The derivative of Eq.(50) with respect to the weights consists of three terms as shown below,

$$\begin{aligned} \frac{\partial}{\partial w_{ij}^{(l)}} J_{\text{NCAE}}(\mathbf{W}, \mathbf{b}) &= \frac{\partial}{\partial w_{ij}^{(l)}} J_{\text{E}}(\mathbf{W}, \mathbf{b}) \\ &+ \beta \frac{\partial}{\partial w_{ij}^{(l)}} J_{\text{KL}}(p \parallel \hat{\mathbf{p}}) \\ &+ \alpha g(w_{ij}^{(l)}) \end{aligned} \quad (54)$$

where

$$g(x) = \begin{cases} w_{ij} & w_{ij} < 0 \\ 0 & w_{ij} \geq 0 \end{cases} \quad (55)$$

The derivative term in Eq.(53) and the first two terms in Eq.(54) are computed using the backpropagation algorithm [3, 121].

## B Deep learning using NCAE

A greedy layer-wise training algorithm is used to build a deep network, with each layer pre-trained separately by unsupervised feature learning [46]. In this section, a deep network is pretrained using an NCAE network, i.e., several layers of the autoencoder are trained step by step, with the hidden activities of the previous autoencoder used as input for the next autoencoder. Finally, the hidden activities of the last autoencoder is used as an input to a softmax regression classifier to be trained in a supervised mode. In the proposed approach, the nonnegative weights of the softmax classifier during training is constrained, as described for training NCAE. The misclassification cost function of the softmax classifier is,

$$J_{\text{CL}}(\mathbf{W}) = -\frac{1}{m} \left[ \sum_{r=1}^m \sum_{p=1}^k 1(y^{(r)} = p) \log \frac{e^{\mathbf{w}_p^T x^{(r)}}}{\sum_{l=1}^k e^{\mathbf{w}_l^T x^{(r)}}} \right] \quad (56)$$

where  $k$  is the number of classes,  $\mathbf{W}$  is the matrix of input weights of all nodes in the softmax layer, and  $\mathbf{w}_p$  is the  $p$ -th column of  $\mathbf{W}$  referring to the input weights of the  $p$ -th softmax node. Therefore, the cost function of Nonnegativity-Constrained Softmax

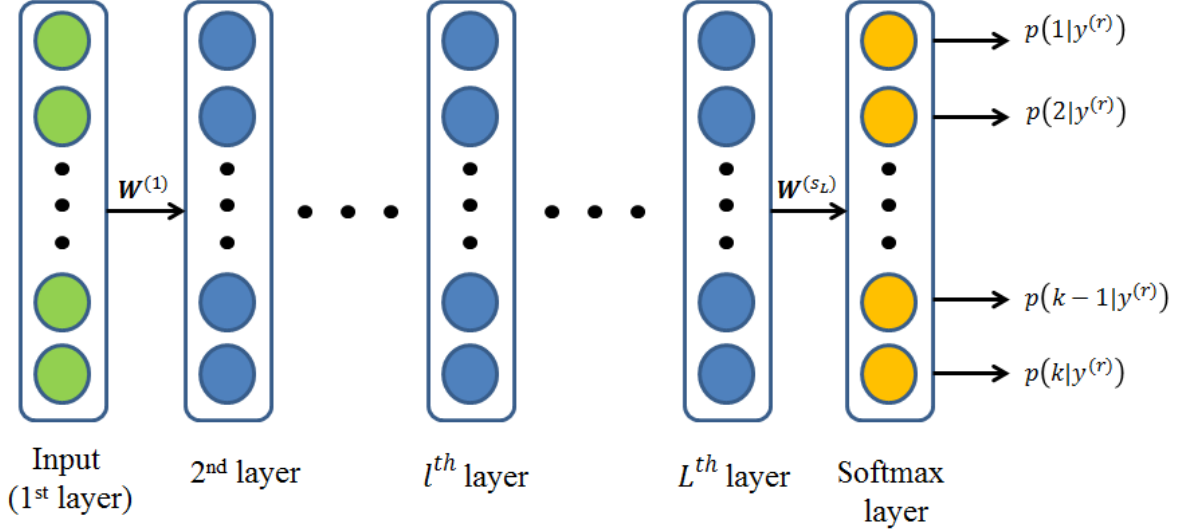


Figure 15. Schematic diagram of a deep network

is defined as,

$$J_{NC-Softmax}(\mathbf{W}) = J_{CL}(\mathbf{W}) + \frac{\alpha}{2} \sum_{i=1}^{s_L} \sum_{j=1}^k f(w_{ij}^{(L)}) \quad (57)$$

where  $s_L$  denotes the number of hidden nodes of the final autoencoder,  $f(\cdot)$  is as in Eq.(51) to penalize the negative weights of the softmax layer. The final step of greedy-wise training is to stack the trained NCAE and softmax layers, and fine-tune the network in supervised mode to improve the classification accuracy of the network [46]. Only the negative weights of the softmax layer are constrained during fine-tuning. The cost function for fine-tuning the Deep Network (DN) is given by

$$J_{DN}(\mathbf{W}, \mathbf{b}) = J_{CL}(\mathbf{W}_{DN}, \mathbf{b}_{DN}) + \frac{\alpha}{2} \sum_{i=1}^{s_L} \sum_{j=1}^k f(w_{ij}^{(L)}) \quad (58)$$

where  $\mathbf{W}_{DN}$  contains the input weights of the NCAE and softmax layers, and  $\mathbf{b}_{DN}$  is the bias input of NCAE layers, as shown in Figure 15.

A batch gradient descent algorithm is used, where the Limited-memory BFGS (L-BFGS) quasi-Newton method [122] is employed for optimization of Eq.(50), Eq.(57), and Eq.(58). The L-BFGS algorithm computes an approximation of the inverse of the Hessian matrix, which results in less memory to store the vectors which approximate the Hessian matrix. The details of the algorithm and the software implementation can

be found in [38].

## C Feature extraction by NCAE

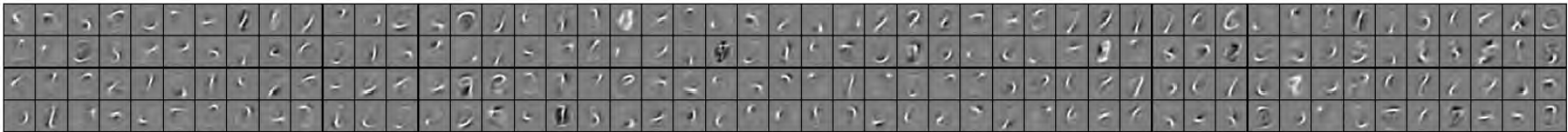
This section tests the performance of the proposed method in learning unsupervised features for three benchmark image data sets and one text dataset. A deep network using NCAE as a building block is trained, and its classification performance is evaluated. The MNIST digit data set is used for handwritten digits [4], the ORL face data set [123] for face images, and the small NORB object recognition dataset [124]. The Reuters 21578 document corpus is used from text to evaluate the ability of the proposed method in learning semantic features.

### 1 Unsupervised feature learning

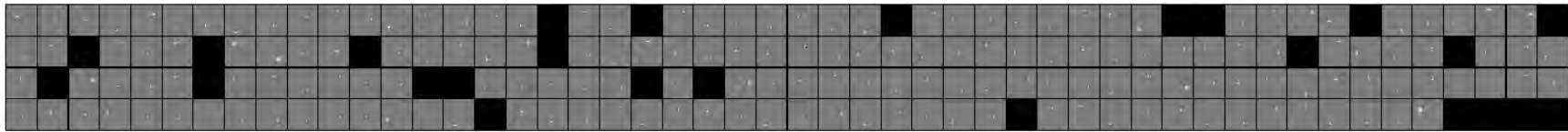
A three-layer Autoencoder with NCAE using Eq.(50) was trained. In the case of image data, the input weights of hidden nodes  $\mathbf{W}_1$  are rendered as images called receptive fields. The results of the NCAE method are compared to the receptive fields learned by a three-layer SAE of Eq.(28), NNSAE [8], and the basis images learned by NMF. The multiplicative algorithm has been used to compute the basis images  $\mathbf{W}$  of NMF [13]. In the case of text data,  $\mathbf{W}_1$  represents the group of words to evaluate the ability to extract meaningful features connected to the topics in the documents.

### 2 Learning part-based representation of images

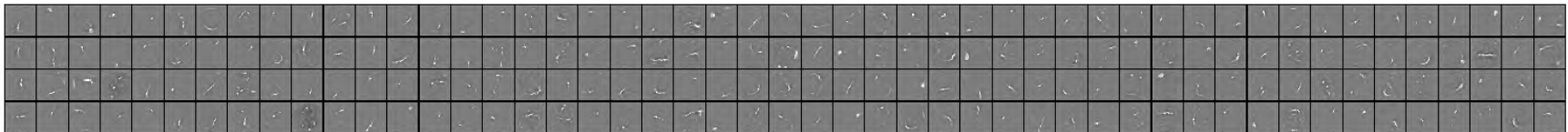
In the first experiment, an NCAE network was trained on the MNIST digit data set. This dataset contains 60,000 training and 10,000 testing grayscale images of handwritten digits, scaled and centered inside a  $28 \times 28$  pixel box. The NCAE network contains 196 nodes in the hidden layer. Its receptive fields have been compared with those of SAE, NNSAE, and NMF basis images in Figure 16, and decoding filters are compared with SAE in Figure 18, with the histogram of weight distribution in Figure 17 and Figure 19, respectively. The results show that receptive fields, learned by NCAE, are more sparse and localized than SAE, NNSAE, and NMF. The darker pixels



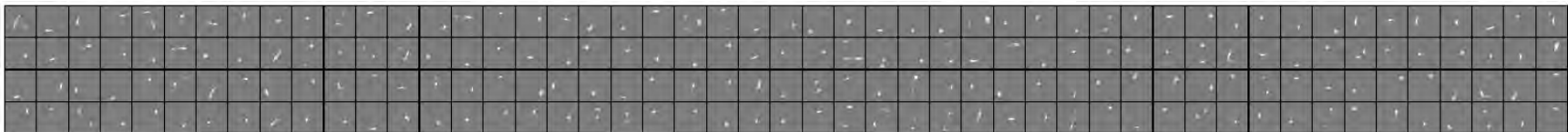
(a) SAE



(b) NNSAE



(c) NCAE\*



(d) NMF

Figure 16. 196 Receptive fields learned from MNIST digit data set using (a) SAE, (b) NNSAE, (c) NCAE\*, and (d) NMF. Black pixels indicate negative, and white pixels indicate positive weights. Black nodes in (b) indicate neurons with zero weights.

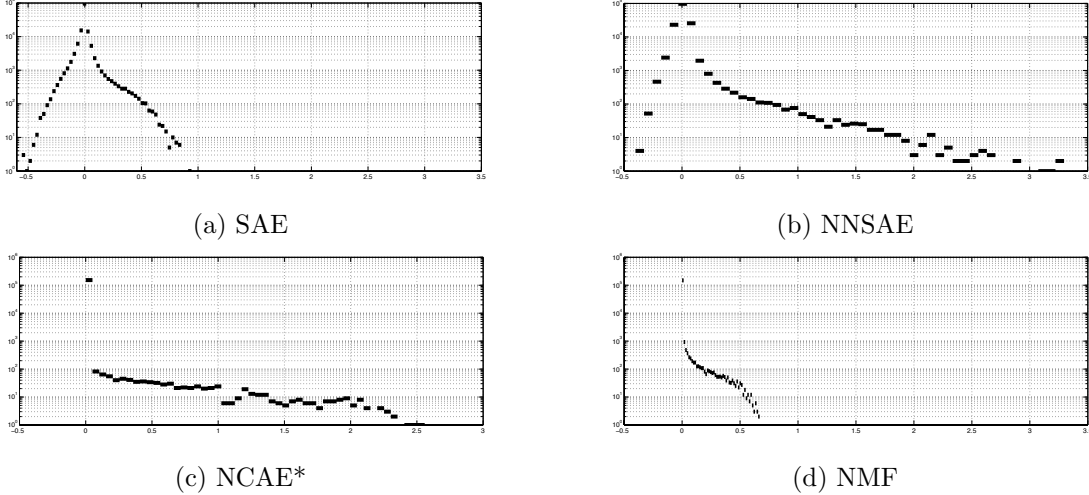
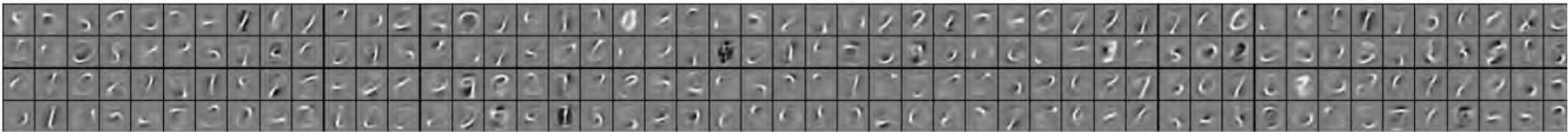


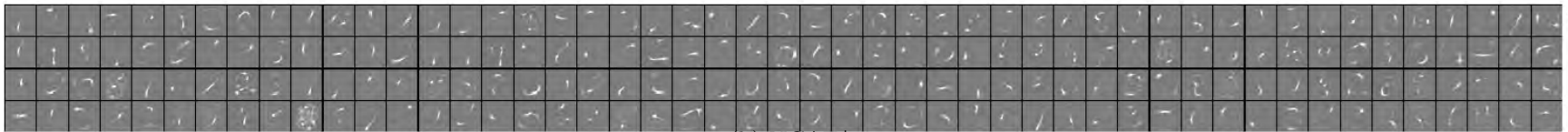
Figure 17. Encoding weight  $\mathbf{W}^{(1)}$  distribution histogram of (a) SAE, (b) NNSAE, (c) NCAE\*, and (d) NMF.

in SAE features indicate negative input weights. In contrast, those values are reduced in NCAE features due to the nonnegativity constraint. Features learned by NCAE in Figure 16 indicate that basic structures of handwritten digits such as strokes and dots are discovered, whereas these are much less visible in SAE, where some features are parts of digits or the whole digits in a blurred form. On the other hand, the features learned by NNSAE and NMF are more local than NCAE, since it is harder to judge them as strokes and dots or parts of digits. As a result, Figure 16 and Figure 18 indicate that the NCAE network learns a sparse and part-based representation of handwritten digits that is easier to interpret, by constraining the negative weights. The comparison of encoding weight distribution histogram to SAE, NNSAE, and NMF method is shown in Figure 17, demonstrating the nonnegativity constraint effect. as demonstrated by the weight histogram. To better investigate the sparsity of weights in the NCAE network, the sparseness is measured using the relationship between the  $\ell_1$  and  $\ell_2$  norms proposed in [27], and the sparseness histograms are compared with other methods in Figure 20 and Figure 21, for the receptive fields and decoding filters, respectively. The results indicate that the nonnegativity constraints improve the sparsity of weights in the encoding and decoding layer.

To evaluate the performance of NCAE in terms of digit reconstruction, the



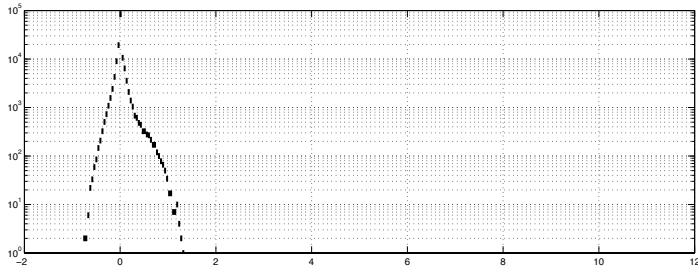
(a) SAE



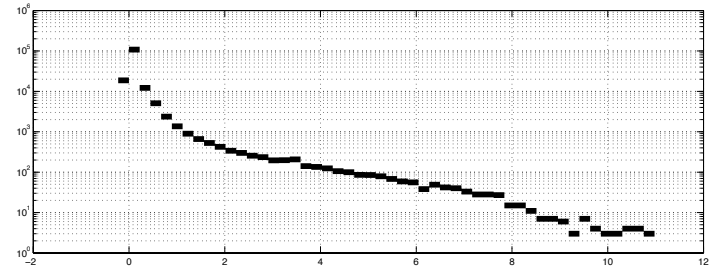
(b) NCAE\*

52

Figure 18. 196 decoding filters ( $\mathbf{W}^{(2)}$ ) with weight histogram learned from MNIST digit data set using (a) SAE and (b) NCAE\*. Black pixels indicate negative, and white pixels indicate positive weights. Black nodes in (b) indicate neurons with zero weights.



(a) SAE



(b) NCAE\*

Figure 19. Decoding weight  $\mathbf{W}^{(2)}$  distribution histogram of (a) SAE, (b) NNSAE, (c) NCAE\*.

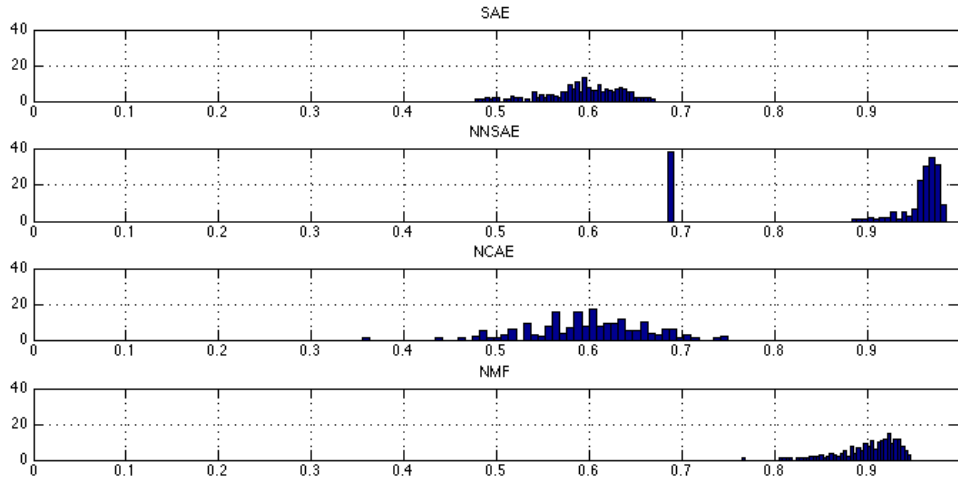


Figure 20. Histogram of the sparseness criterion (Eq.(11)) measured on 196 receptive fields.

selected reconstructed digits and the reconstruction error of NCAE for different numbers of hidden nodes are compared with those of SAE, NNSAE, and NMF in Figure 22. The reconstruction of ten selected digits from ten classes is shown in Figure 22. The top row depicts the original digits from the data set, where the reconstructed digits using SAE, NNSAE, NCAE, and NMF algorithms are shown below. It is clear that the digits reconstructed by NCAE are more similar to the original digits than those by the SAE and NNSAE methods, and also contain fewer errors. On the other hand, the results of NCAE and NMF are similar, while digits in NMF are more blurred than NCAE, which indicates reconstruction errors. In order to test the performance of the proposed method using different numbers of hidden neurons, the reconstruction error (Eq.(68)) of all digits of the MNIST data set is depicted in Figure 23. The results demonstrate that NCAE outperforms SAE and NNSAE for different numbers of hidden neurons. It can be seen that the reconstruction errors in NCAE and NMF methods are the lowest and similar, whereas NCAE shows better reconstruction over NMF in one case. The results in Figure 23 demonstrate that the nonnegativity constraint forces the autoencoder networks to learn part-based representation of digits, i.e. strokes and dots, and it results in more accurate reconstruction from their encodings than SAE and NNSAE.

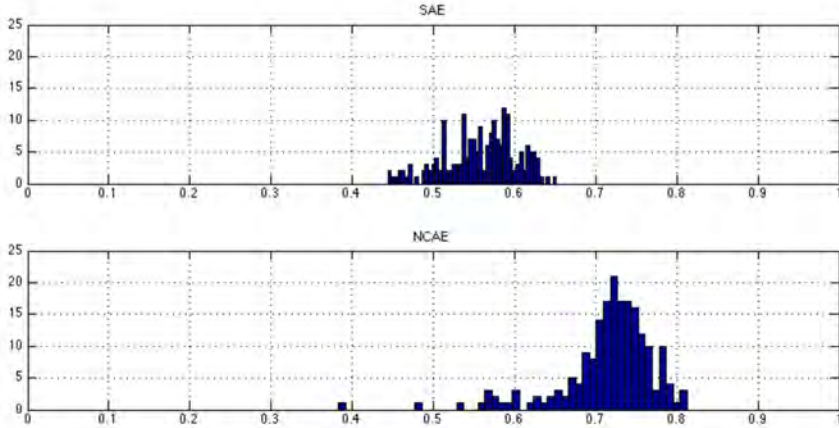


Figure 21. Histogram of the sparseness criterion (Eq.(11)) measured on 196 decoding filters.

To better evaluate the hidden activities, Figure 24 depicts the sparsity measured by the KL divergence of Eq.(27) for different numbers of hidden neurons in NCAE and SAE networks. The results indicate that the hidden activations in NCAE are more sparse than SAE, since  $J_{KL}(p||\hat{p})$  is reduced significantly. This means that the hidden neurons in NCAE are less activated than in SAE when averaged over the full training set. In order to evaluate the ability of the proposed method in discovering the hidden structure of data in the original high-dimensional space, the distributions of MNIST digits in the higher representation level, i.e. hidden activities in SAE, NNSAE and NCAE neural networks, and feature encoding of NMF ( $\mathbf{H}$ ), are visualized in Figure 25(a), 25(b), 25(c), and 25(d) for SAE, NNSAE, NCAE, and NMF, respectively. The figures show the reduced 196-dimensional higher representations of digits in 2D space using t-distributed Stochastic Neighbor Embedding (t-SNE) projection [7]. The comparison between these methods reveals that the distributions of digits for SAE, NCAE, and NMF are more similar to each other than NNSAE. It is clear that manifold of digits in NNSAE have more overlap and more twists than the other methods. On the other hand, the manifolds of digits 7, 9, 4 in NCAE are more linear than in SAE and NMF. The comparison between manifolds of other digits in terms of shape and

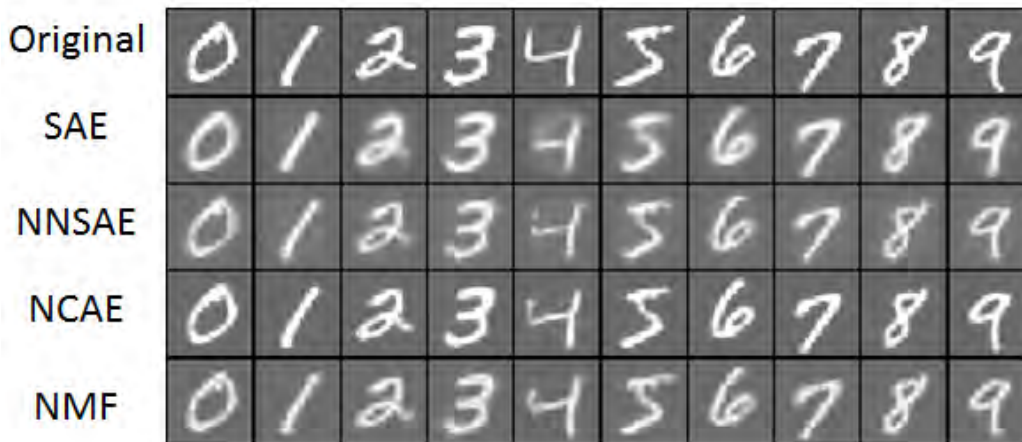


Figure 22. Reconstruction comparison of the MNIST digits data set by 196 receptive fields, using SAE , NNSAE [8], NCAE\*, and NMF.

distance indicates that NCAE, SAE, and NMF have similar characteristics.

The second experiment is to test the performance of the proposed method on the ORL database of faces (AT&T at Cambridge) [123]. This database contains 10 different images of 40 subjects. For some subjects, the images were taken at different times, with varying lighting, facial expressions, and facial details. The original size of each image is  $92 \times 112$  pixels, with 256 gray levels per pixels. To decrease the computational time, the input layer size of SAE and NCAE is reduced by resizing the images to  $46 \times 56$  pixels. The dataset is divided to 300 faces for training and 100 for testing.

The features learned from the ORL data are depicted in the images of receptive fields in Figure 26(a-d) using the SAE, NNSAE, NCAE, and NMF methods, respectively. The receptive fields of SAE indicate holistic features from different faces, i.e. each feature is a combination of different faces of the database. On the other hand, the receptive fields in NCAE indicate sparse features of faces, where several parts of faces can be recognized. Most of the features learned by NCAE contain some parts of the faces, e.g. eye, nose, mouth, etc. together. The nonnegativity constrains negative weights in the NCAE network to become zero, as indicated by fewer darker pixels in the receptive fields. The features learned by NNSAE and NMF indicate that most features are holistic, whereas most face parts are visible in the basis images. In NMF

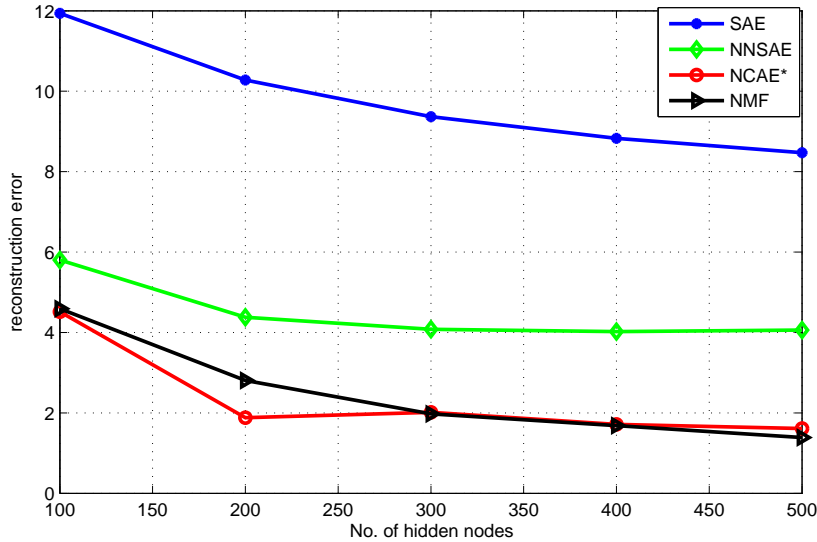


Figure 23. Comparison of reconstruction error computed by Eq.(25).

and NNSAE, the extracted features are only nonnegative values, but it does not help in creating sparse features, because hard constraints on negative weights force the algorithm to learn complex receptive field of the basis image in NNSAE and NMF, respectively. It can be concluded that NCAE was able to learn hidden features showing part-based representation of the faces using soft constraints on the negativity weights, whereas this is not achieved by SAE, NNSAE, and NMF. To assess the performance of the proposed method in recovering the images, the reconstructed faces of several subjects are shown in Figure 27. The faces reconstructed by NCAE appear more similar to the original images than those by SAE, NNSAE, and NMF. The reason is that NCAE could extract the hidden features, which show parts of the faces in the encoding layer, and these features help the autoencoder network in composing the faces from these features, e.g. eye, nose, mouth, etc., in the decoding layer. However, it is hard to compose the original face from the holistic features created by SAE, NNSAE, and NMF.

To investigate the effect of the nonnegativity constraint penalty coefficient ( $\alpha$ ) in NCAE for learning part-based representation, different values of  $\alpha$  is tested to train NCAE. The hidden features are depicted in Figure 28. For this experiment,  $\alpha$  is increased logarithmically for three values in the range  $[0.003, \dots, 0.3]$ . The results indicate that by increasing  $\alpha$ , the resulting features are more sparse, and decompose

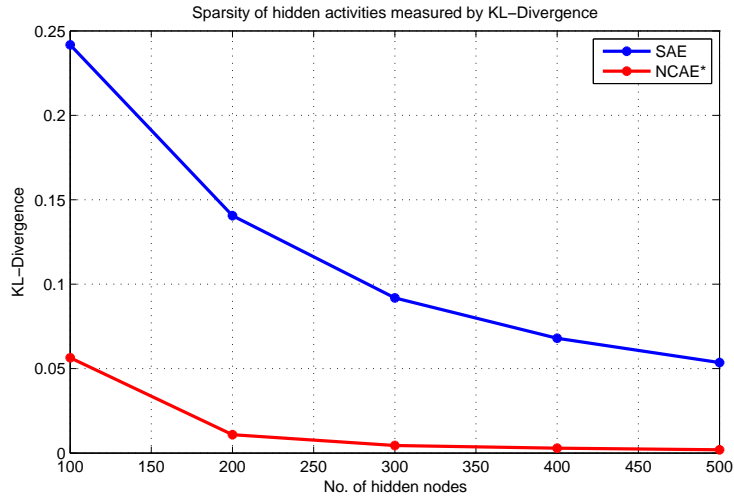
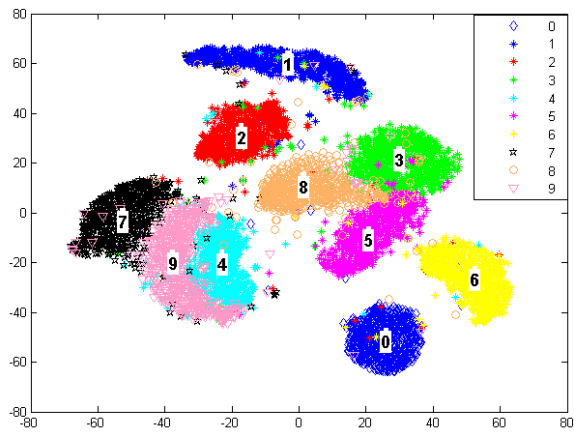


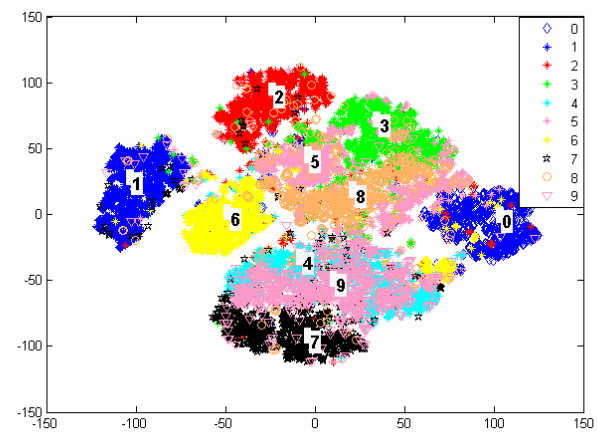
Figure 24. Sparsity of hidden units measured by the KL divergence in Eq.(27) for the MNIST dataset for  $p=0.05$ .

faces into smaller parts. It is clear that the receptive fields in Figure 28(c) are more sparse, and only show few parts of the faces. This test demonstrates that NCAE is able to extract different types of eyes, noses, mouths, etc. from the face database.

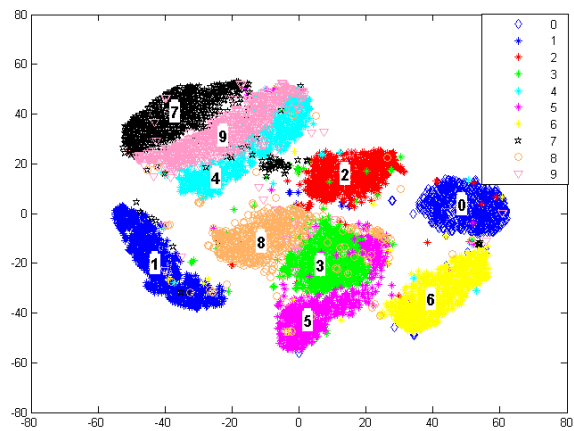
In the third experiment, the NORB normalized-uniform dataset [124] is used, which contains 24,300 training examples and 24,300 test examples. This database contains images of 50 toys from five generic categories: four-legged animals, human figures, airplanes, trucks, and cars. The training and testing sets are composed of 5 instances of each category. Each image consists of two channels, each of size  $96 \times 96$  pixels. To evaluate the performance of the method, an autoencoder is trained using 100 hidden neurons for SAE, NNSAE, and NCAE, and also NMF with 100 basis vectors. The learned features are shown as receptive fields in Figure 29. The results indicate that the receptive fields learned by NCAE are more sparse than SAE and NNSAE, since they mainly capture the edges of the toys. On the other hand, the receptive fields from SAE and NNSAE represent more holistic features. The basis images learned by NMF also show edge-like features, however, they are more holistic than the NCAE features.



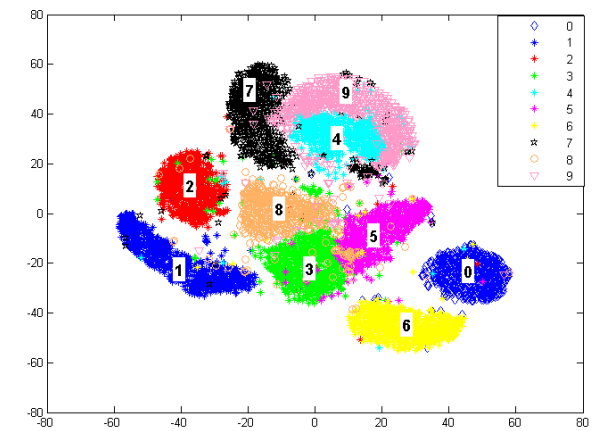
(a) SAE



(b) NNSAE



(c) NCAE\*

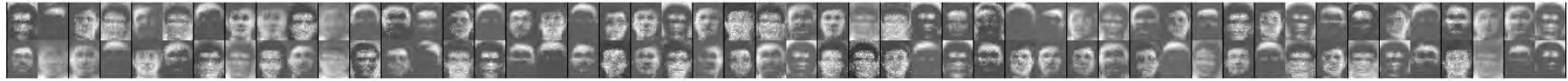


(d) NMF

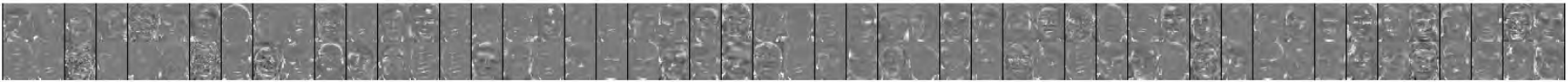
Figure 25. Visualization of MNIST handwritten digits. 196 higher representation of digits computed using (a) SAE, (b) NNSAE, (c) NCAE\*, and (d) NMF are visualized using t-SNE projection [7].



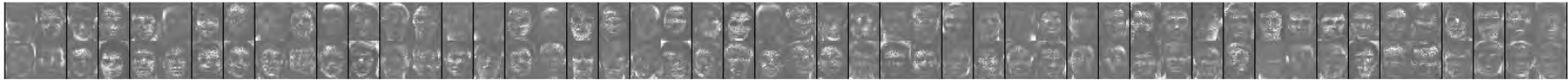
(a) SAE



(b) NNSAE



(c) NCAE\*



(d) NMF

59

Figure 26. 100 Receptive fields learned from the ORL faces data set using (a) SAE, (b) NCAE\*, and (c) NMF. Black pixels indicate negative weights, and white pixels indicate positive weights.



Figure 27. Reconstruction of the ORL Faces test data using 300 receptive fields, using SAE (error=8.6447), NNSAE (error=15.7433), NCAE\* (error=5.4944), and NMF (error=7.5653).

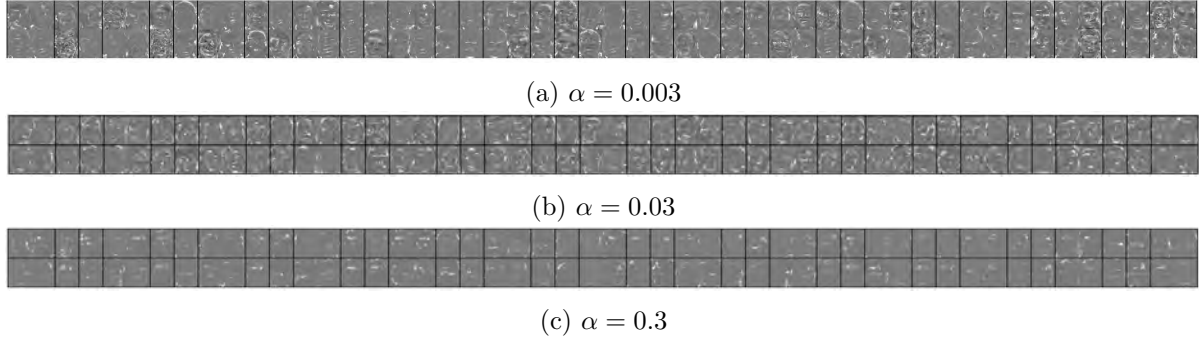
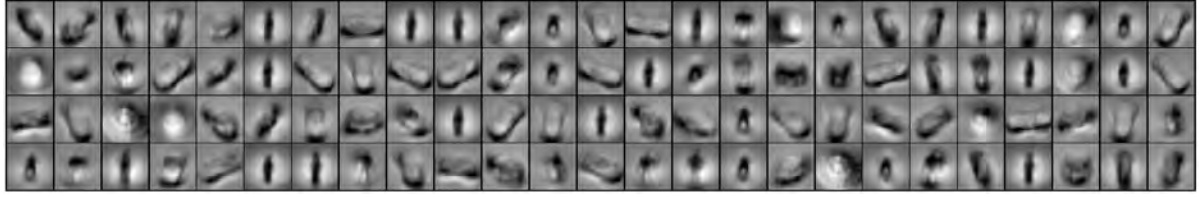


Figure 28. 100 Receptive fields learned from ORL Faces data set using NCAE for varying nonnegativity penalty coefficients ( $\alpha$ ). Brighter pixels indicate larger weights.

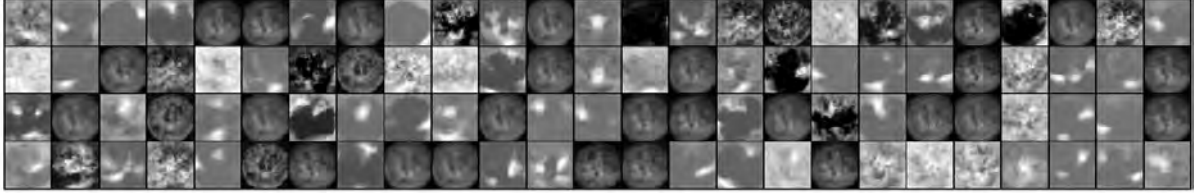
### 3 Semantic feature discovery from text data

In this experiment, the NCAE method is evaluated on extracting semantic features from text data. The documents are first converted to a TF-IDF vector space model [125]. Part-based representation in text documents is more complicated than in images, since the meaning of document can not be easily inferred by adding the words it contains. However, the topic of a document can be inferred from a group of words delivering the most information. Therefore, to detect the topic of a document, the autoencoder network should be able to extract these groups of words in its encoding layer to generate a meaningful semantic embedding. To evaluate the proposed method, the Reuters-21578 text categorization collection is used. It is composed of documents that appeared in the Reuters newswire in 1987. The ModApte split is used to limit to the 10 most frequent categories. A processed (stemming, stop-word removal) version in bag-of-words format is used, which obtained from <http://people.kyb.tuebingen.mpg.de/pgehler/rap/>.

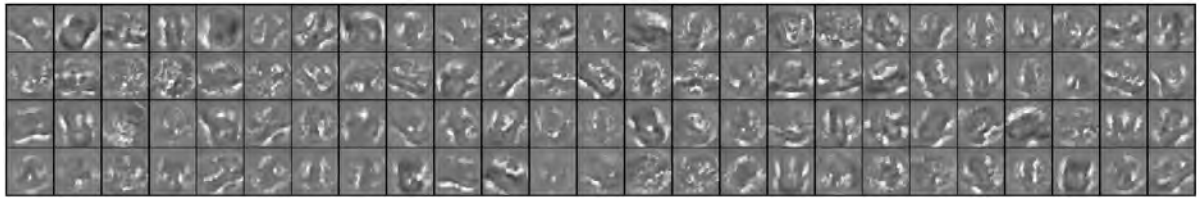
The dataset contains 11,413 documents with 12,317 words/dimensions. Two techniques were used to reduce the dimensionality of each document to contain the most informative and less correlated words. First, words were sorted based on their frequency of occurrence in the dataset. Then the words with frequency below 4 and above 70 were removed. After that, the information gain with the class attribute [89] was used to select the most informative words which do not occur in every topic. The remaining words in the dataset were sorted using this method, and the less important



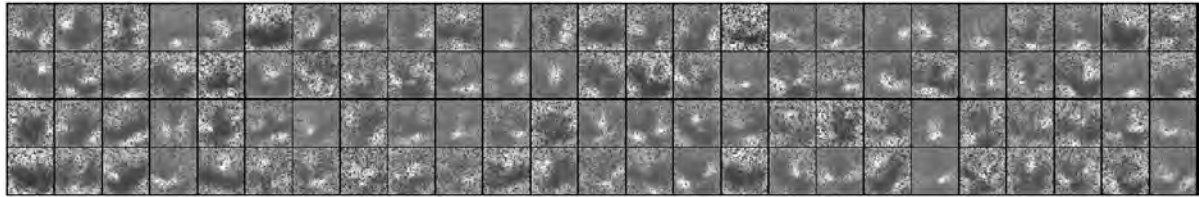
(a) SAE



(b) NNSAE



(c) NCAE\*



(d) NMF

Figure 29. 100 Receptive fields learned from small NORB data set using (a) SAE, (b) NNSAE, (c) NCAE\*, and (d) NMF. Black pixels indicate negative, and white pixels indicate positive weights.

words were removed based on the desired dimension of documents. In this experiment, the dimensionality of documents is reduced to the size [200, 300, 400].

To examine the features extracted in the encoding layer of NCAE, 20 words connecting via the highest weights to each hidden neuron were examined. The connecting weight from each word to a hidden neuron is equal to the magnitude of the association of the word to the latent feature extracted by the hidden node. Using this interpretation, a hidden node with the largest connecting weight of words related to a specific topic can be assigned as a class detector for that topic. An NCAE network is trained with 200 input neurons and 15 hidden neurons. Figure 30 depicts the selected

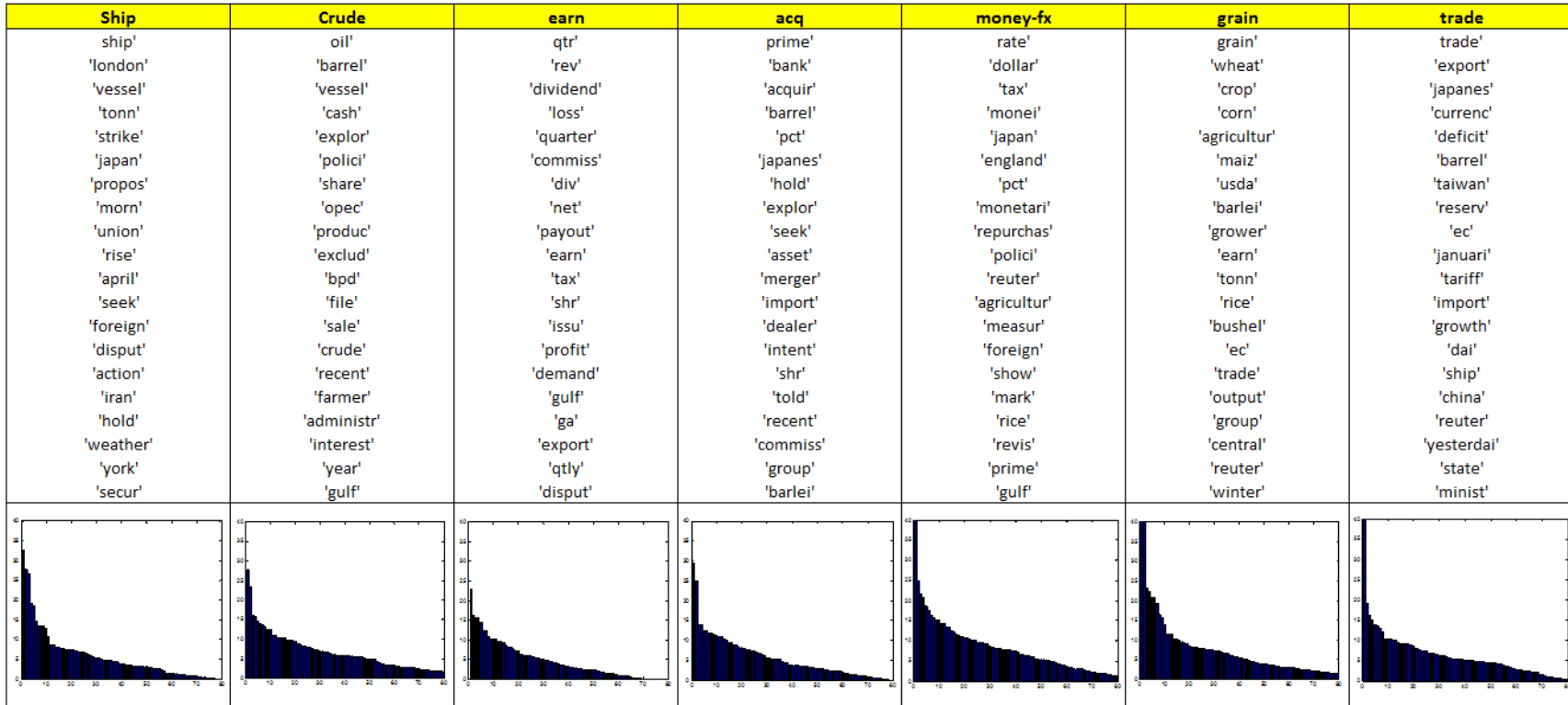
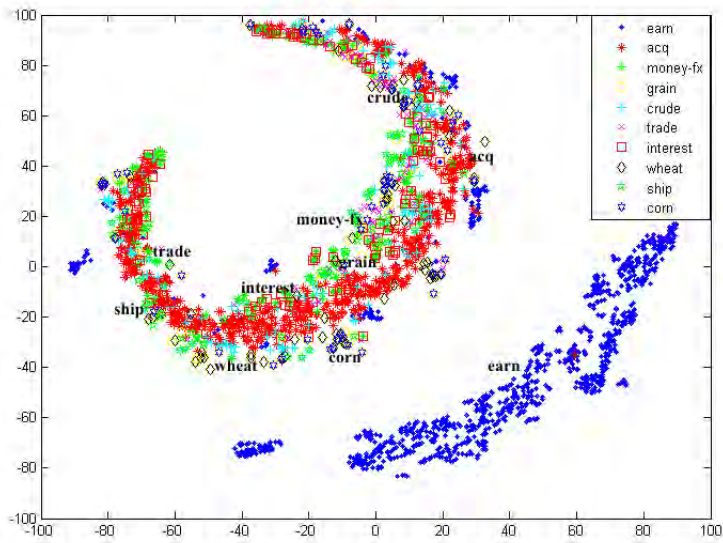
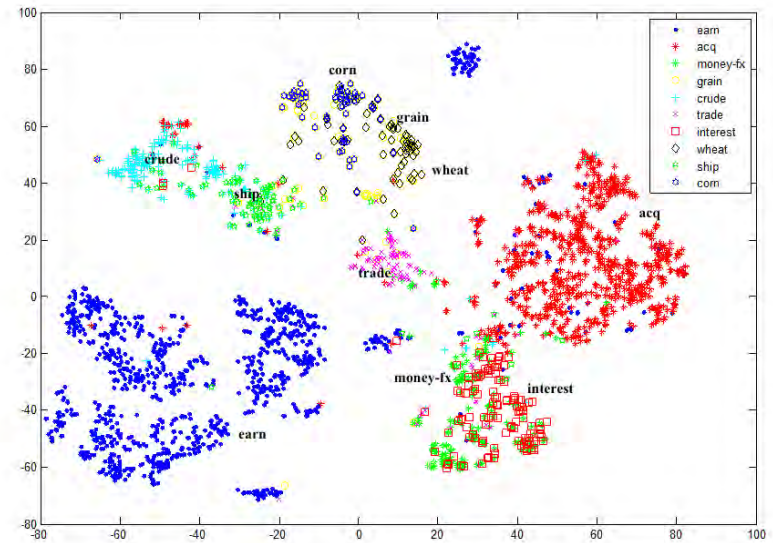


Figure 30. An example of 7 most distinguishable categories, i.e., ship, crude, earn, acq, money-fx, grain and trade associated with top 20 words (ranked by their weights) discovered from the Reuters-21578 document corpus. The charts at the bottom row illustrate the weight impact of words on the category.



(a) SAE



(b) NCAE\*

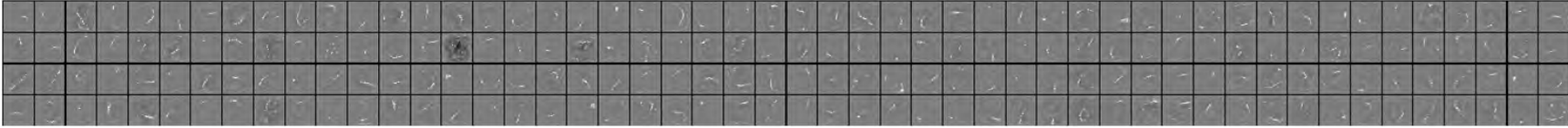
Figure 31. Visualization of the Reuters documents data based on the 15-dimensional higher representation of documents computed using (a) SAE and (b) NCAE\*. Visualization used t-SNE projection [7].

seven nodes showing the seven distinguishable topics of the dataset. The top row shows the list of words with the topic inferred from the corresponding list. The bottom row depicts the distribution of connecting weights in decreasing order. It can be concluded that the semantically related words of a topic are grouped together in each hidden node. To further evaluate the ability of the NCAE network to disentangle the semantic features (topic detector) from the dataset, the distribution of documents in the hidden layer, Figure 31(b), is compared to the SAE method, as depicted in Figure 31(a), where topic information is used for visual labeling. It is clear that NCAE is able to group the related documents together, whereas the topics which are meaningfully related are closer in the semantic space.

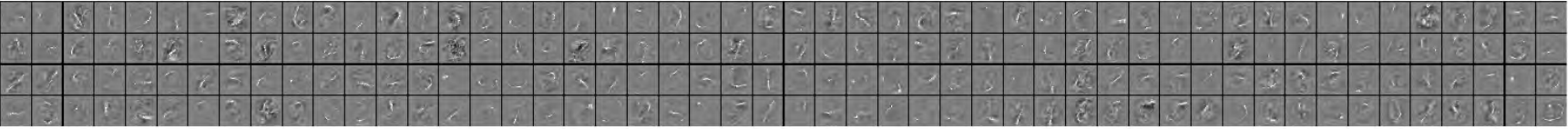
#### 4 Supervised learning

The next step is to investigate whether the ability of a deep network in to decompose data into parts, with improved ability to disentangle the hidden factors in its layers, can improve prediction performance. In this section, a deep network is pretrained by stacking several NCAE networks, trained in the previous section. Then a softmax classifier is trained using the hidden activities of the last autoencoder using Eq.(57). Finally, the deep network is fine-tuned using Eq.(58) to improve the classification accuracy. The results are compared to deep neural networks trained using SAE, NNSAE, Denoising Autoencoder (DAE) [50], and Dropout Autoencoder (DpAE) [126] on the MNIST, NORB, and the Reuters-21578. The classification results are averaged over 10 experiments to mitigate the effect of initialization.

Tables 5-7 report the classification accuracy of the deep network, pre-trained with different autoencoders. The results indicate that a deep network with NCAE yields a significantly better accuracy than other networks *before* fine-tuning for all three datasets, and *after* fine-tuning for two of the three data sets. For the NORB data set, although the NCAE network was significantly superior *before* fine-tuning, the classification results indicate no significant difference between NCAE, DAE, and DpAE networks, *after* fine-tuning. The convergence speed of the different networks were also compared based on the number of iterations during fine-tuning. These are listed



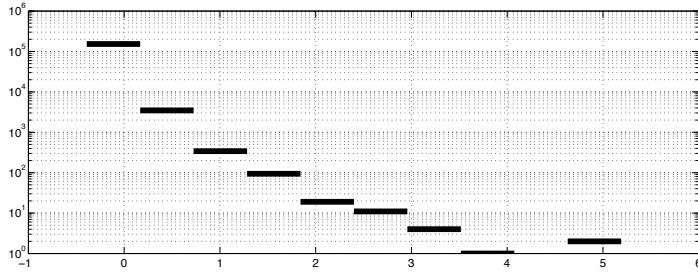
(a)



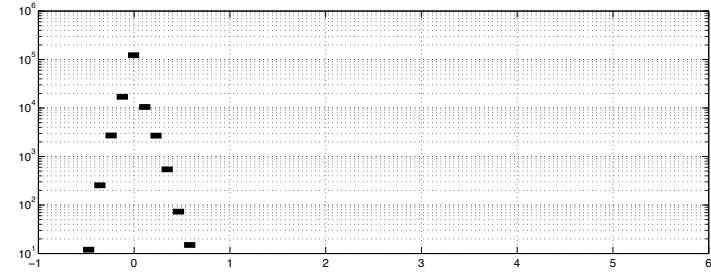
(b)

32

Figure 32. 200 Receptive fields of the first layer of the deep network after fine-tuning using (a) all weights constrained, and (b) only Softmax weights constrained. Black pixels indicate negative weights, and white pixels indicate positive weights.

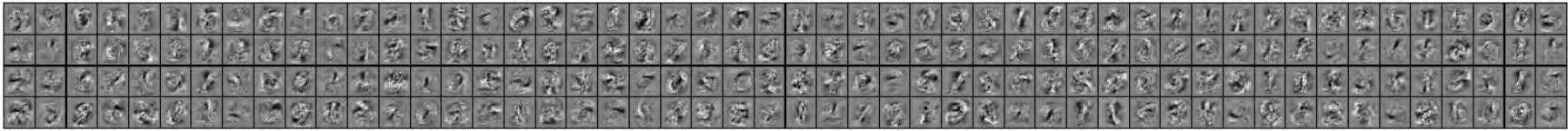


(a)

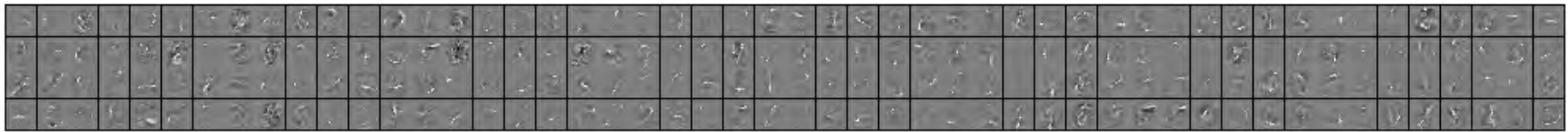


(b)

Figure 33. Weight distribution of the first layer of deep network after finetuning for (a) all weights constrained, and (b) Softmax-layer only constrained. According to histogram, 5.76% of weights become negative.



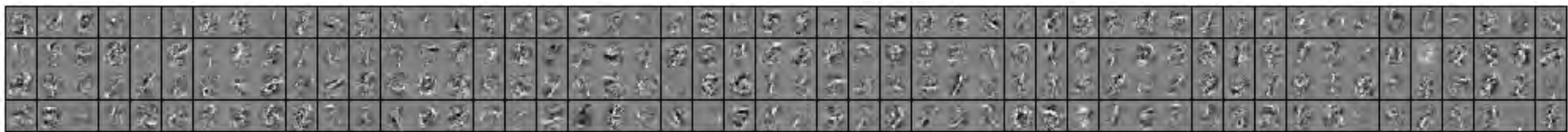
(a) SAE



(b) NCAE\*



(c) DAE



(d) NC-DAE



(d) DpAE

Figure 34. 200 Receptive fields of the first layer of the deep network after fine-tuning using (a) SAE, (b) NCAE\*, (c) DAE, (d) DpAE on the MNIST data. Black pixels indicate negative, and white pixels indicate positive weights.

alongside the error rates in Tables 5-7. It can be seen that NCAE network converges faster than other methods, since it yields better accuracy *before* fine-tuning. Note that all networks were trained for the same number of iterations (400) *before* fine-tuning. Therefore NCAE’s superior performance is not at the cost of more iterations.

TABLE 5. Performance of supervised learning methods on MNIST dataset.

Model (784-200-20-10)	Before fine-tuning		After fine-tuning		
	Mean±SD	p-value	Mean±SD	p-value	# Iterations
Deep NCAE*	<b>84.83±0.094</b>		<b>97.91±0.1264</b>		<b>97</b>
Deep SAE	52.81±0.1277	<0.0001	97.29±0.091	<0.0001	400
Deep NNSAE	69.72±0.1007	<0.0001	97.18±0.0648	<0.0001	400
Deep DAE (50% input dropout)	11.26±0.14	<0.0001	97.11±0.0808	<0.0001	400
Deep NC-DAE (50% input dropout)	84.37±0.1318	<0.0001	97.42±0.0757	<0.0001	106
Deep DpAE (50% hidden dropout)	16.77.0784	<0.0001	96.73±0.1066	<0.0001	400

TABLE 6. Performance of supervised learning methods on NORB dataset.

Model (2048-200-20-5)	Before fine-tuning		After fine-tuning		
	Mean±SD	p-value	Mean±SD	p-value	# Iterations
Deep NCAE*	<b>75.54±0.1152</b>		87.76±0.3613		<b>242</b>
Deep SAE	20.00±0.1768	<0.0001	87.26±0.3109	0.0039	400
Deep NNSAE	19.93±0.2230	<0.0001	79.00±0.0962	<0.0001	400
Deep DAE (50% input dropout)	44.03±0.1553	<0.0001	<b>88.11±0.3861</b>	0.0508	400
Deep DpAE (50% hidden dropout)	49.49±0.1437	<0.0001	87.75±0.2767	0.9454	400

TABLE 7. Performance of supervised learning methods on Reuters-21578 Dataset.

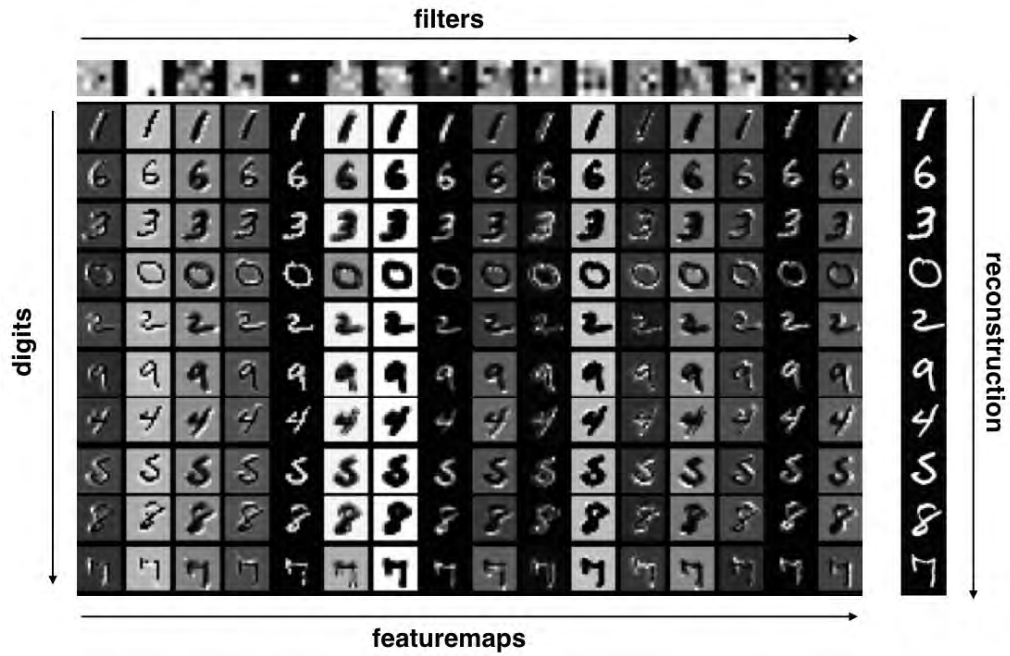
Model (200-15-10)	Before fine-tuning		After fine-tuning		
	Mean±SD	p-value	Mean±SD	p-value	# Iterations
Shallow NCAE*	<b>57.18±0.3639</b>		<b>81.15±0.1637</b>		400
Shallow SAE	39.00±0.2255	<0.0001	78.60±0.2143	<0.0001	400
Shallow DAE (50% input dropout)	39.00±0.3617	<0.0001	76.35±0.1918	<0.0001	400
Shallow DpAE (20% hidden dropout)	39.00±0.4639	<0.0001	78.04±0.1709	<0.0001	400
Shallow DpAE (50% hidden dropout)	39.00±0.3681	<0.0001	72.12±0.2901	<0.0001	400

Figure 32 shows how far the nonnegativity property enforced during the pretraining is preserved in unconstrained hidden layer during fine-tuning stage, by comparing the difference when the nonnegativity constraint is added to the weights in hidden layers. Figure 33 also depicts the deviation in weight distribution histogram due to imposing nonnegativity constraint on softmax layer only. To relate the improved accuracy to part-based decomposition of data, the first hidden layer of each deep network is depicted in Figure 34. It demonstrates that the deep network based on NCAE could decompose data into clearly distinct parts in the first layer, whereas there

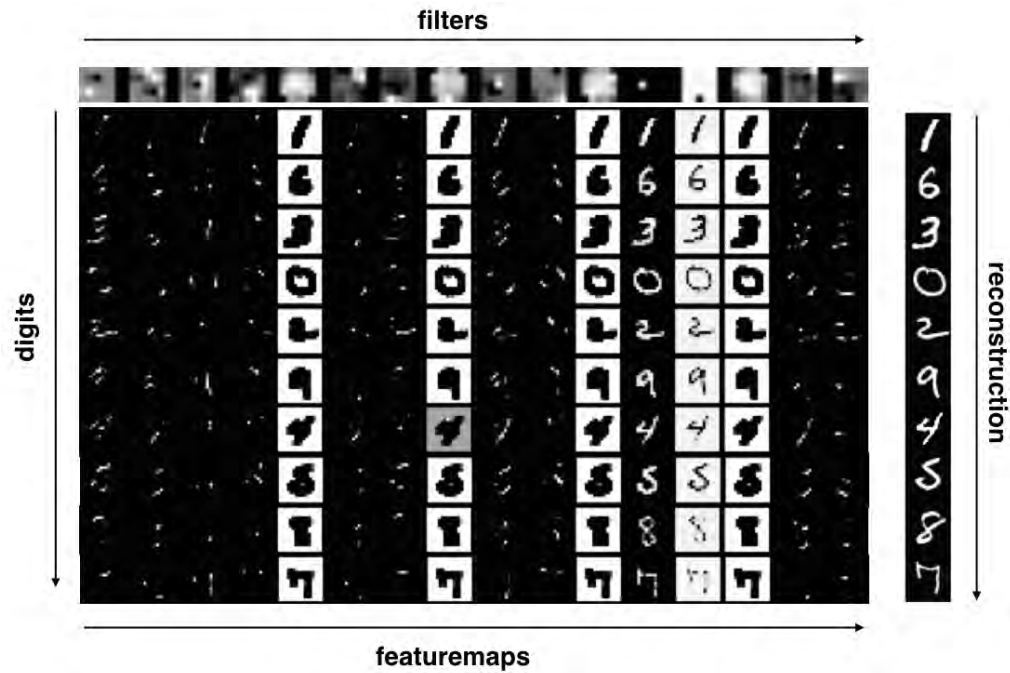
are more holistic features in other networks. This property leads to better discrimination between classes at the next layers, thus resulting in better classification. The nonnegativity constraint in the DAE network (NC-DAE) has also been tested on the MNIST dataset. The results indicate that the performance of NC-DAE improves over DAE *before* and *after* fine-tuning, since the hidden layers of NC-DAE are able to decompose data into parts, and also it converges faster *after* fine-tuning. Table 6 reports the classification results on the small NORB dataset, and demonstrates that the deep network with NCAE outperforms the other networks *before* fine-tuning. However, its performance is not significantly different from the deep networks based on DAE and DpAE, based on the  $p$ -values. Table 7 also reports the classification accuracy computed with several one-hidden-layer networks. It also indicates that the deep network built with NCAE outperforms other deep networks *before* and *after* fine-tuning by a large margin, due to their ability to extract semantic features from documents on this data.

#### D Structured Sparse Convolutional Autoencoder (SSCAE)

In this section, the model of Structured Sparse CAE (SSCAE) is described. CAE consists of convolution/pooling/nonlinearity based encoding and decoding layers, where the feature vector is represented as featuremaps, i.e. hidden output of the encoding layer.



(a) CAE



(b) SSCAE

Figure 35. 16 example filters ( $\mathbf{W}^{k \in [1, \dots, 16]} = [w_{ij}]_{5 \times 5}$ ) and featuremaps ( $\mathbf{h}^{k \in [1, \dots, 16]} = [h_{ij}^k]_{24 \times 24}$ ), with feature vectors ( $\mathbf{h}_{ij} = [h_{ij}^k]_{1 \times 16}$ ), extracted from non-whitened MNIST with sigmoid nonlinearity and no pooling using (a) CAE, (b) SSCAE. Effect of sparse feature extraction using SSCAE is shown w/o pooling layer. Digits are input pixelmaps  $28 \times 28$ ,  $n = 16$  for this example.

In CAE with  $n$  encoding filters, the featuremaps  $\mathbf{h}^{k \in n}$  are computed based on a convolution/pooling/nonlinearity layer, with nonlinear function applied on the pooled activation of convolution layer, as in Eq.(59).

$$\mathbf{h}^{k \in n} = f(x * \mathbf{W}^{k \in n} + b^{k \in n}) \quad (59)$$

where  $\mathbf{W}^{k \in n}$  and  $b^{k \in n}$  are the filter and bias of  $k$ -th featuremap, respectively.  $f(\cdot)$  indicates the nonlinearity function, e.g. sigmoid, Rectified Linear unit (ReLU).  $h_{ij}^k$  is referred to as single neuron activity in  $k$ -th featuremap  $\mathbf{h}^k$ , whereas  $\mathbf{h}_{ij} = [h_{ij}^k]^{k \in n}$  is defined as a feature vector across featuremaps  $\mathbf{h}^{k \in n}$  as illustrated in Figure 35.

In SSCAE, the featuremaps  $\mathbf{h}^{k \in n}$  are regularized and sparsified to represent three properties; (i) *Sparse feature vector*  $\mathbf{h}_{ij}$ ; (ii) *Sparse neuronal activity*  $h_{ij}^k$  within each of the  $k$ -th featuremap  $\mathbf{h}^k$ ; (iii) *Uniform distribution of feature vectors*  $\mathbf{h}_{ij}$ .

In (i), sparsity is imposed on feature vector  $\mathbf{h}_{ij}$  to increase diversity of features represented by each featuremap, i.e. each  $\mathbf{h}^{k \in n}$  should represent a distinguished and discriminative characteristic of the input, such as different parts, edges, etc. This property is exemplified in Figure 35(b) with digits decomposed into parts across featuremaps  $\mathbf{h}^{k \in n}$ . As stipulated in (ii), sparsity is imposed on each featuremap  $\mathbf{h}^{k \in n}$  to only contain few non-zero activities  $h_{ij}^k$ . This property is encouraged for each featuremap to represent a localized feature of the input. Figure 35(b) shows property (i) for MNIST dataset, where each featuremap is a localized feature of a digit, wherein Figure 35(a) shows only extracted digit shape-resemblance features, a much less successful and non-sparse outcome compared to Figure 35(b) where most featuremaps are sparsified. Figure 36 also depicts the technique for numerical sparsification of each featuremap. The property (iii) is imposed on activation features  $\mathbf{h}_{ij}$  to have similar statistics with uniform activity. In other words,  $\mathbf{h}_{ij}$  will be of nearly equal or uniform activity level, if they lie in the object spatial region, or non-active, if not. Uniform activity also improves the generic and part-based feature extraction where the contributing activation features  $\mathbf{h}_{ij}$  of digits, i.e.  $\mathbf{h}_{ij}$ , fall within convolutional region of digits and filters  $\mathbf{W}^{k \in n}$  show uniform activity level, which results in generic and part-based features.

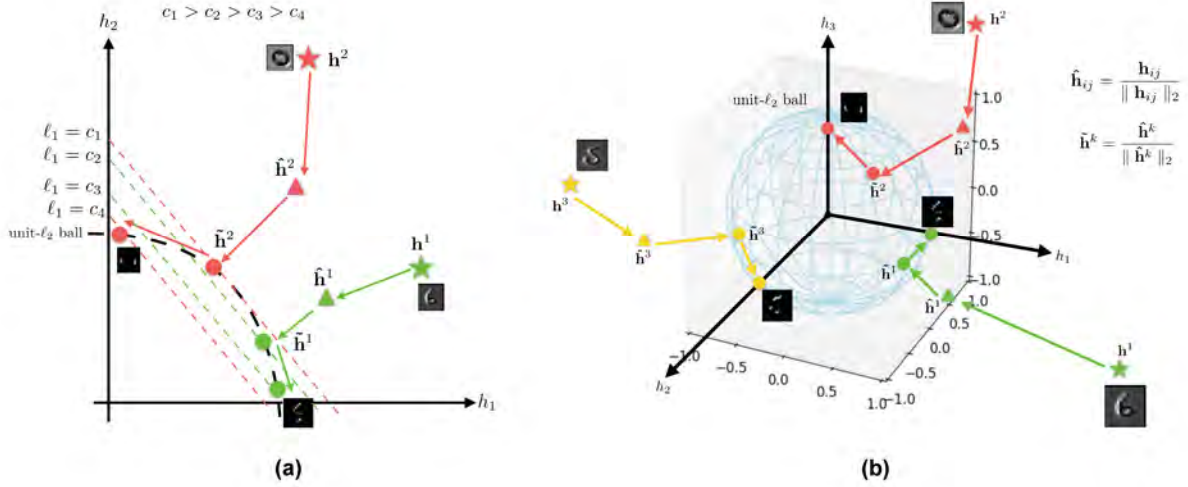


Figure 36. Structured Sparsity illustrated with on (a) two-dimensional and (b) three-dimensional space for featuremaps ( $\mathbf{h}^1$ ,  $\mathbf{h}^2$ ,  $\mathbf{h}^3$ ) of MNIST dataset. Each example is first projected onto the unit  $\ell_2$ -ball and then optimized for  $\ell_1$  sparsity. The unit  $\ell_2$ -ball is shown together with level sets of the  $\ell_1$ -norm. Notice that the sparseness of the features (in the  $\ell_1$  sense) is maximized when the examples are on the axes [9].

To enforce the aforementioned sparsity properties in CAE models, the combination of  $\ell_2$  and  $\ell_1$  normalization is used on  $\mathbf{h}^{k \in n}$  of Eq.(59), as proposed in [9], and as shown in Figure 37. In SSCAE, a normalization layer is added on the encoding layer, where the normalized featuremaps  $\tilde{\mathbf{h}}^{k \in n}$  and feature vectors  $\tilde{\mathbf{h}}_{ij}$  are imposed by two  $\ell_2$ -normalization steps, as in Eq.(61) and Eq.(60), respectively,

$$\hat{\mathbf{h}}_{ij} = \frac{\mathbf{h}_{ij}}{\|\mathbf{h}_{ij}\|_2} \quad (60)$$

$$\tilde{\mathbf{h}}^k = \frac{\hat{\mathbf{h}}^k}{\|\hat{\mathbf{h}}^k\|_2} \quad (61)$$

The final normalized featuremaps  $\tilde{\mathbf{h}}^{k \in n}$  are forwarded as inputs to the decoding layer of unpooling/deconvolution/nonlinearity to reconstruct the input  $x$  as in Eq.(62),

$$\tilde{x} = f\left(\sum_{k \in n} \tilde{\mathbf{h}}^k * \mathbf{P}^k + c^k\right) \quad (62)$$

where  $\mathbf{P}^k$  and  $c^k$  are the filters and biases of decoding layer. In order to enforce the sparsity properties of (i)-(iii), the  $\ell_1$  sparsity is applied on  $\tilde{\mathbf{h}}^{k \in n}$  as in Eq.(64), where

the averaged  $\ell_1$  sparsity over  $n$  featuremaps and  $m$  training data is minimized during the reconstruction of input  $x$ , as in Eq's.(63), (64) and (65),

$$\mathcal{L}_{L2rec} = \| \mathbf{x} - \tilde{\mathbf{x}} \|_2 \quad (63)$$

$$\mathcal{L}_{L1sp} = \frac{1}{m} \frac{1}{n} \sum_{d \in m} \sum_{k \in n} \| \tilde{\mathbf{h}}^k \|_1 \quad (64)$$

$$\mathcal{L}_{SSCAE} = \mathcal{L}_{L2rec} + \lambda_{L1sp} \mathcal{L}_{L1sp} \quad (65)$$

where  $\mathcal{L}_{L2rec}$ ,  $\mathcal{L}_{L1sp}$  and  $\mathcal{L}_{SSCAE}$  are the reconstruction, sparsity and SSCAE loss functions, respectively.  $\lambda_{L1sp}$  indicates the sparsity penalty on  $\tilde{\mathbf{h}}^{k \in n}$  and  $\tilde{\mathbf{h}}_{ij}$ . Figure 36 demonstrate the steps of normalization and sparsification by selected feature maps of MNIST data. This model has been tested on other dataset for exploring feature extraction performance, which is described in the following section.

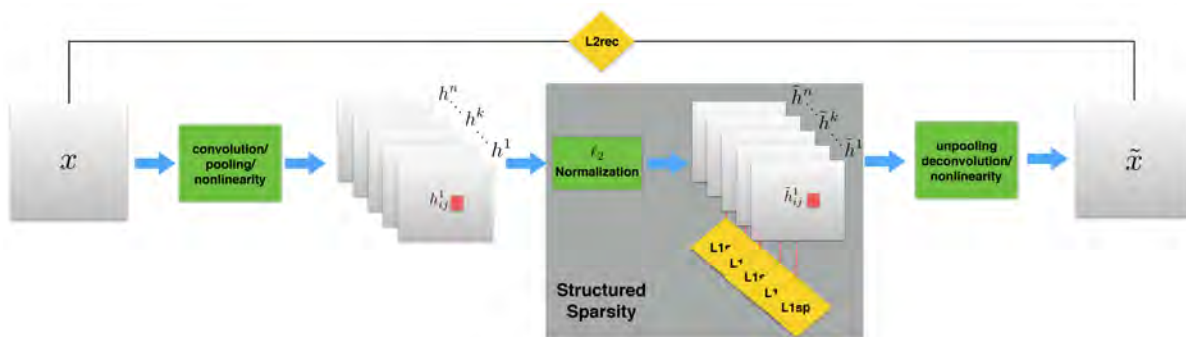


Figure 37. Model architecture of Structured Sparse Convolutional AutoEncoder (SSCAE)

## E Feature extraction by SSCAE

In this section, the performance of structured sparsity in CAE network is investigated for unsupervised feature extraction, in terms of dead filter learning, part-based learning in featuremaps and reconstruction-based learning. The

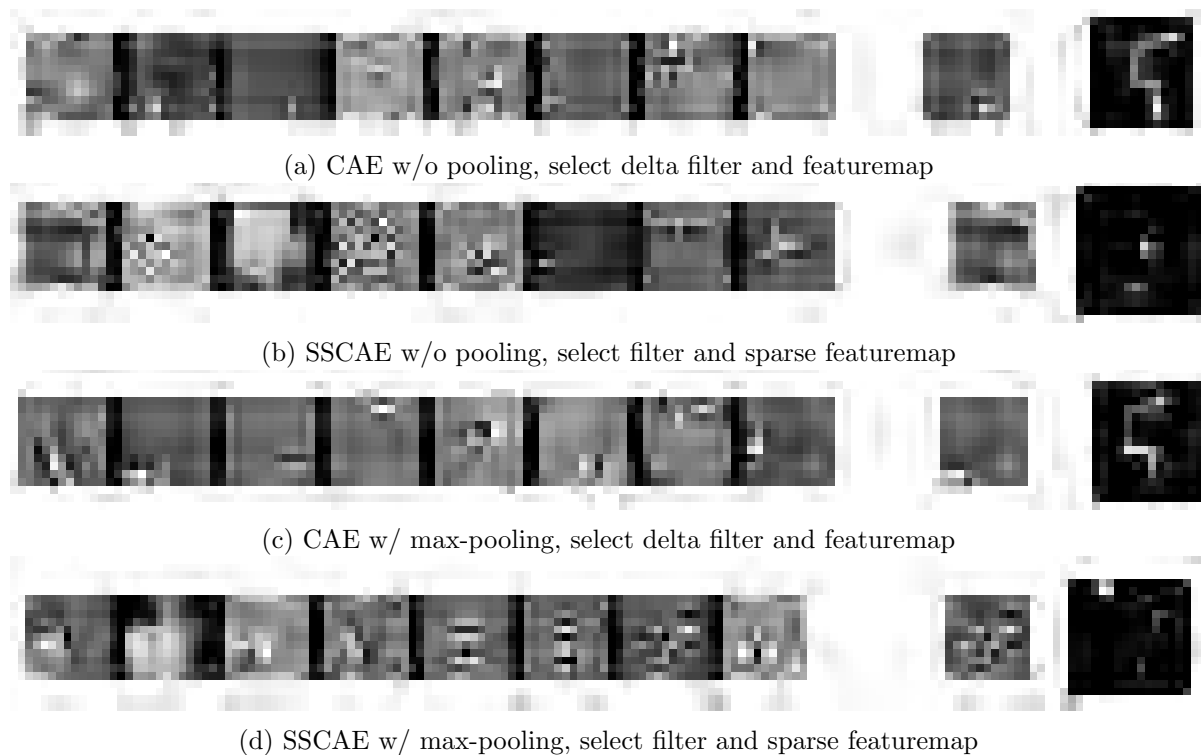
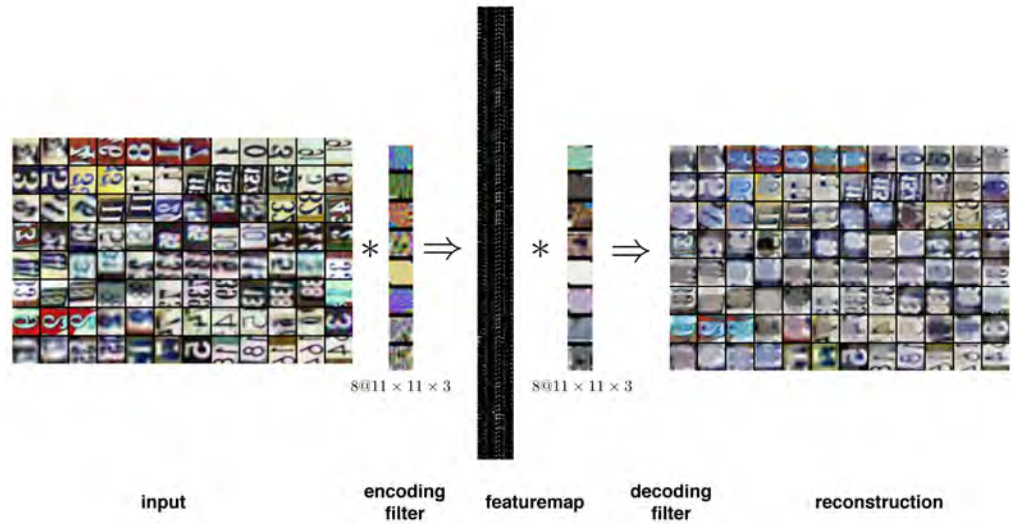


Figure 38. Comparison of 8 filters learned from MNIST by CAE and SSCAE w/o pooling (a,b) and w/ non-overlapping max-pooling (c,d) using ReLu ( $\max(0, x)$ ) nonlinearity. Select single filter and respective featuremaps shown on the digit.

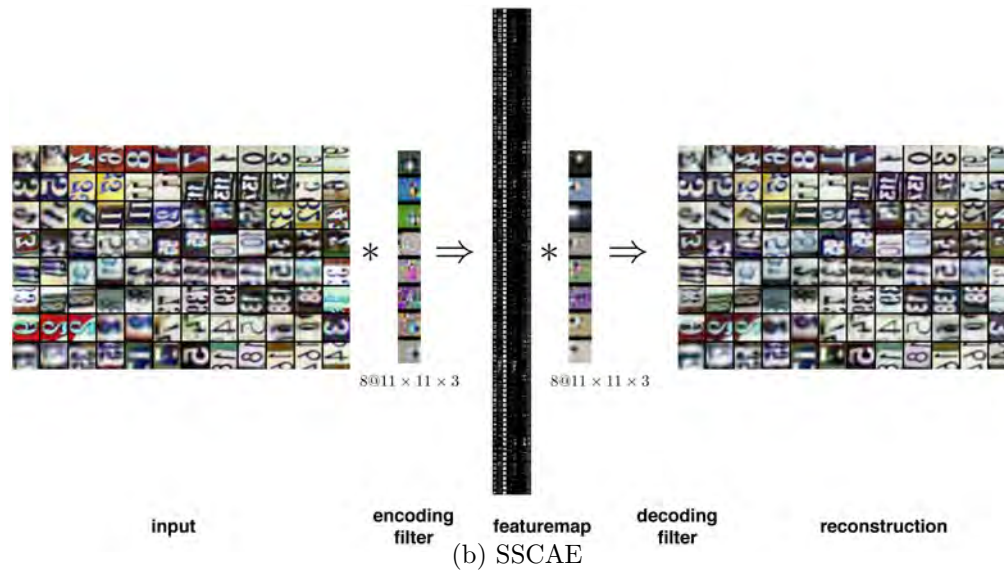
performance is evaluated on MNIST and Street View House Numbers (SVHN) datasets. ZCA normalization is also employed as whitening preprocessing step [3]. Theano [127] and Pylearn [128] are used, on Amazon EC2 g2.8xlarge instances with GPU GRID K520 for the experiments.

## 1 Minimizing dead filters

In order to compare the performance of the proposed model in minimizing dead filters by learning sparse and local filters, the trained filters of MNIST data are compared between CAE and SSCAE with and without pooling layer in Figure 38. It is shown in Figure 38(a)(c) that CAE with and without pooling layer learn some delta filters which provide simply an identity function. However, the sparsity function used in SSCAE is trying to reduce in extracting delta filters by managing the activation across featuremaps, as shown in Figure 38(b)(d).

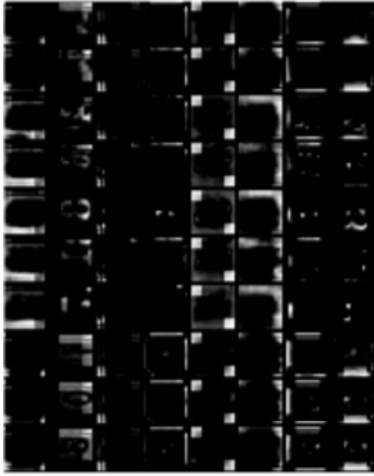


(a) CAE

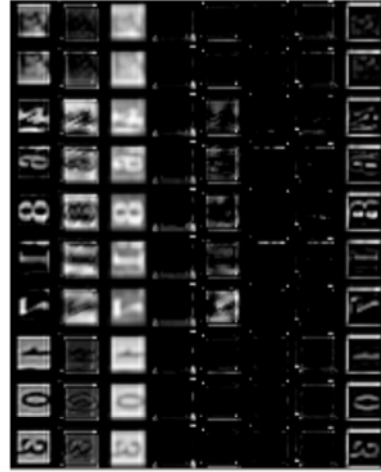


(b) SSCAE

Figure 39. SVHN data-flow visualization in (a) CAE and (b) SPCAE with 8 filters. The effect of structured sparsity is shown in encoding and decoding filters and the reconstruction. No ZCA whitening [3] is applied.



(a) CAE

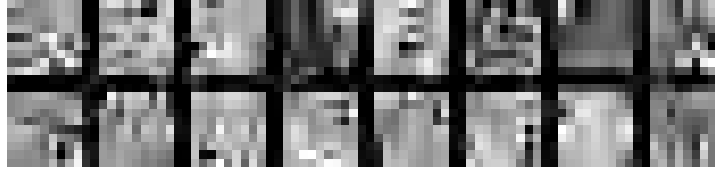


(b) SSCAE

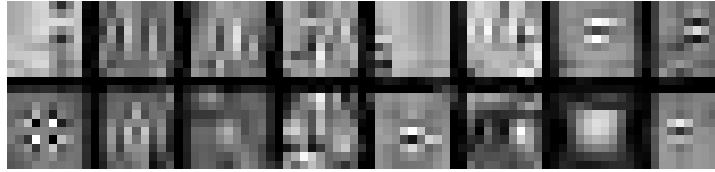
Figure 40. Selected featuremap of SVHN dataset extracted by (a) CAE, and (b) SSCAE with 8 filters of  $11 \times 11 \times 3$  size. No ZCA whitening is applied.

## 2 Improving learning of reconstruction

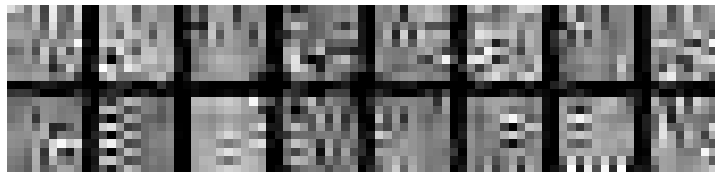
To investigate the effect of structured sparsity on learning of filters through reconstruction, the performance of CAE and SSCAE is compared on SVHN dataset, as shown in Figure 39. To show the performance of structured sparsity on reconstruction, a small CAE with 8 filters is trained on SVHN dataset. Figure 39(a) shows the performance of CAE after training which fails to extract edge-like filters and results in poor reconstruction. Figure 41 also depicts the learned 16 encoding and decoding filters on small NORB dataset, where structured sparsity improve the extraction of localized and edge-like filters. However, SSCAE outperform CAE in reconstruction due to learned edge-like filters. The selected featuremap of the two models are shown in Figure 40(a)(b). The convergence rate of reconstruction optimization for CAE and SSCAE is also compared on MNIST (Figure 42(a)), SVHN (Figure 42(b)), small NORB (Figure 42(c)), and CIFAR-10 (Figure 42(d)) datasets, which indicate faster convergence in SSCAE.



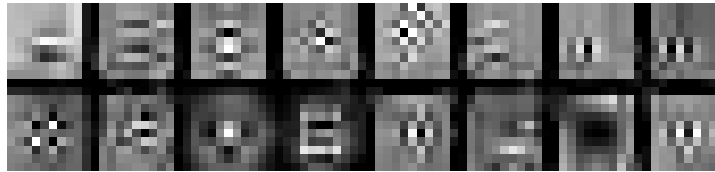
(a) CAE encoding filter



(b) SSCAE encoding filter



(c) CAE decoding filter



(d) SSCAE decoding filter

Figure 41. 16 Learnt encoding and decoding filters of (a)(c) CAE and (b)(d) SSCAE on small NORB dataset.

## F Conclusion

In this chapter, two new algorithms were introduced for part-based feature extraction.

In the first part, a new learning algorithm was developed for training a deep autoencoder network with nonnegative weights constraints, first in the unsupervised training of the NCAE autoencoder, and then in the supervised fine-tuning stage [16]. Nonnegativity has been motivated by the idea in NMF that promotes additive features and captures part-based representation of data. The performance of the proposed method, in terms of decomposing data into parts and extracting meaningful features, was compared to the SAE, NNSAE and NMF. The prediction performance of the deep network has also been compared to the SAE, NNSAE, DAE, and DpAE. The

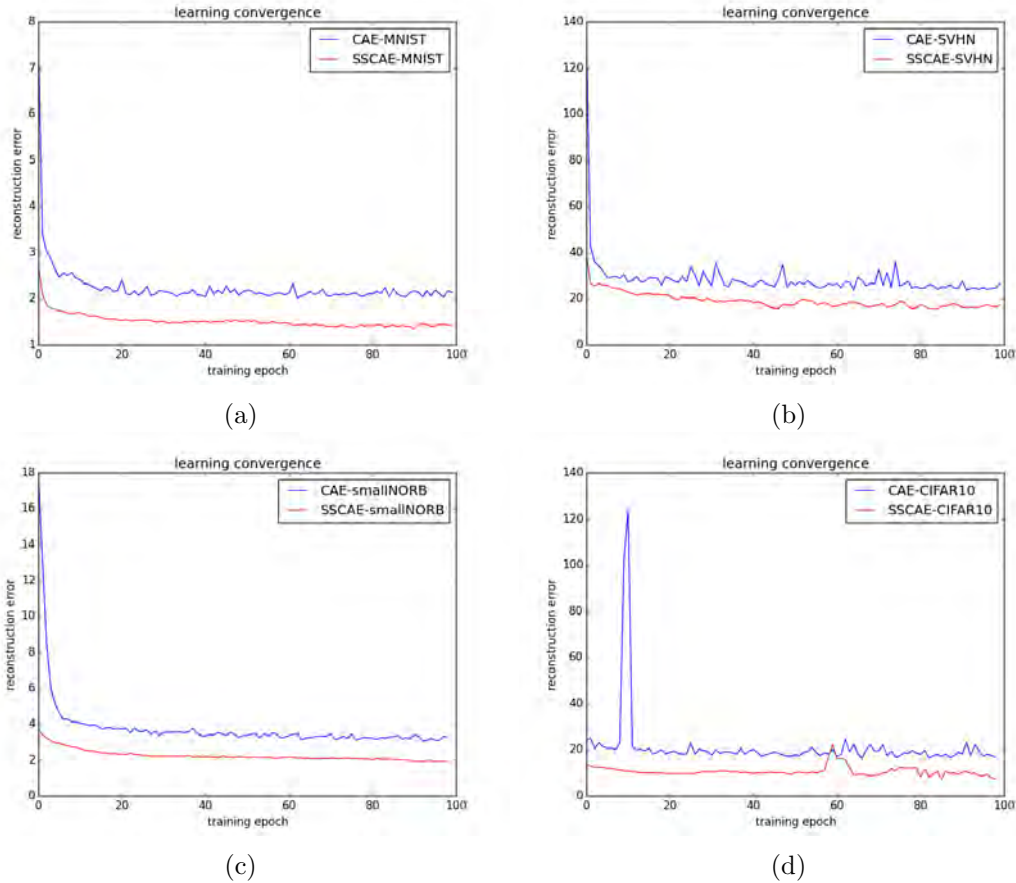


Figure 42. Learning rate convergence of CAE and SSCAE on (a) MNIST, (b) SVHN, (c) small NORB, and (d) CIFAR-10 dataset using 16 filters of  $11 \times 11 \times 3$  size.

performance is evaluated on the MNIST data set of handwritten digits, the ORL data set of faces, small NORB data set of objects, and Reuters-21578 text corpus. The results were evaluated in terms of reconstruction error, part-based representation of features, and sparseness of hidden encoding in the unsupervised learning stage. The results indicate that the nonnegativity constraints in the NCAE method force the autoencoder to learn features that capture a part-based representation of data, while achieving lower reconstruction error and better sparsity in the hidden encoding as compared with SAE and NMF. The numerical classification results also reveal that a deep network trained by restricting the number of nonnegative weights in the autoencoding and softmax classification layer achieves better performance. This is due to decomposing data into parts, hierarchically in its hidden layers, which helps the classification layer to discriminate between classes.

In the second part, a structured sparsity model was proposed using  $\ell_2$  and  $\ell_1$  normalization. SSCAE was demonstrated to extract sparse features by decoupling the spatial interaction in neighboring hidden neurons. It was also indicated that due to structuring the distribution of neuronal activity, the problem of dead filter extraction can be mitigated. The comparison with CAE indicated the ability of SSCAE in simplifying the model structure and preserving the reconstruction performance.

## CHAPTER VI

### DEEP NETWORK FOR ALZHEIMER'S DISEASE DETECTION

Deep learning methods have recently made notable advances in the tasks of classification and representation learning. These tasks are important for brain imaging and neuroscience discovery. The medical images are very high-dimensional and they should be represented in their intrinsic dimensionality to detect patterns. On the other hand, each image could also be identified by a smaller set of parameters that describe shape variations and patterns that are common for a particular group of images. This makes deep learning methods suitable for the dimension of image can be reduced in hierarchical layer to capture anatomical variation related to clinical diagnosis [129].

CNN are supervised networks proven to preserve the inputs neighborhood relations and spatial locality in their latent higher-level feature representations. While the common fully connected deep architectures do not scale well to realistic-sized high-dimensional images in terms of computational complexity, CNN do, since the number of free parameters describing their shared weights does not depend on input dimensionality [4]. On the other hand, CAE is a hierarchical unsupervised feature extractor that scales well to high-dimensional 2D images [58–60]. It learns non-trivial features using plain stochastic gradient descent, and discovers good CNN initializations that avoid the numerous distinct local minima of highly non-convex objective functions arising in virtually all deep learning problems.

#### A Alzheimer's Disease (AD)

AD is a progressive brain disorder and the most common case of dementia in the late life. AD leads to the death of nerve cells and tissue loss throughout the brain. AD results in the reducing of the brain volume in size dramatically through time, and

affects most of its functions [130], as shown in Figure 43. It is estimated that the number of affected people will double in the next 20 years, where 1 in 85 people will have AD by 2050 [131]. While the cost of care of AD patients is expected to rise dramatically, the demand for developing a CAD system for early and accurate diagnosis of AD becomes imperative [132].

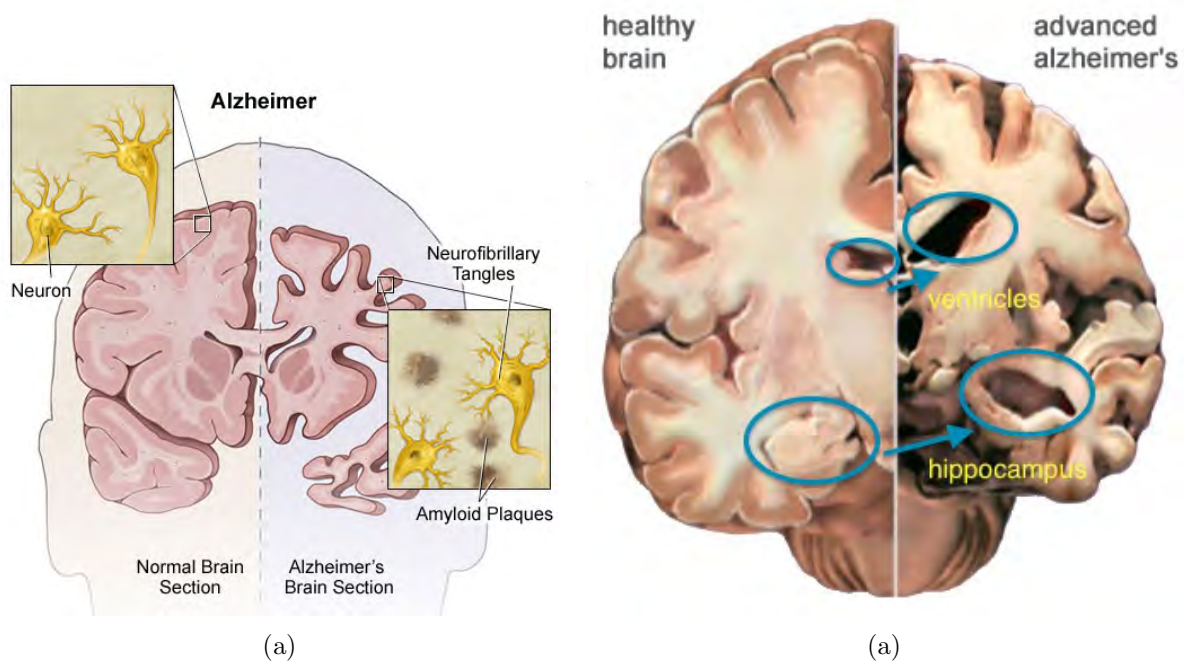


Figure 43. In the Alzheimer's brain: (a) The cortex shrivels up, damaging areas involved in thinking, planning and remembering. (b) Shrinkage is especially severe in the hippocampus, an area of the cortex that plays a key role in formation of new memories. Ventricles (fluid-filled spaces within the brain) grow larger. Copyright©2011 Alzheimer's Association®.

The early diagnosis of AD is primarily associated to the detection of Mild Cognitive Impairment (MCI), a prodromal stage of AD. Though the memory complaints and deficits of MCI do not notably affect the patients daily activities, it has been reported that MCI has a high risk of progression to AD or other forms of dementia [3]. The accurate diagnosis of AD plays a significant role in patient care, especially at the early stage, because the consciousness of the severity and the progression risks allows the patients to take prevention measures before irreversible brain damages are shaped.

## B AD diagnosis

Several machine learning based approaches are also proposed to leverage the multi-view information, i.e. MRI, PET, CSM modalities to predict the Alzheimer. Liu et al. [133] extracted multi-view features of MRI subjects using several selected template in dataset. Then subjects are clustered within each classes using generated tissue density map of each template, and the encoding feature of each subject is extracted. Finally an ensemble of SVM classifiers are used to compute the class of subject.

Deep learning refers to the methods that extract the hierarchical features from data by training several layers of feature extractors. Deep learning aims to decrease the use of domain expert in designing and extracting discriminative features, which make them appropriate for brain imaging, e.g. sMRI, fMRI, etc. [134]. Several deep network based models are proposed for Alzheimer diagnosis using different image modalities and clinical data. Suk et al. [135] used stacked autoencoder to extract features from MRI, PET, and CSF images separately. Then a feature selection method is employed to derive features from combination of the learned features from previous step each with clinical scores (MMSE ADAS-cog). Finally they used a multi-kernel SVM for Alzheimer classification using three extracted features of MRI, PET and CSF. In [136], a multimodal Deep Boltzmann Machine (DBM) is used to extract  $k$  feature from selected  $k$  patches of MRI and PET scans. then an ensemble of SVM classifiers are used for Alzheimer prediction. Liu et al. [137] extracted 83 Region Of Interests (ROI) of MRI and PET scans and used multimodal fusion to create one set of feature to train stacked layers of denoising autoencoders. Li et al. [138] developed a multi-task deep learning (MTL) for class, MMSE and ADAS-cog modeling by multimodal fusion of MRI and PET features into the deep network, which is pretrained by RBM. MTL was used to leverage the label information of MMSE and ADAS-cog for AD classification.

## C MRI imaging for AD diagnosis

To develop a CAD system, several neuroimaging biomarkers, e.g. structural MRI (sMRI), functional MRI (fMRI), and PET, were investigated and considered as diagnosis criteria for AD [139,140]. Among them, sMRI is recognized as a non-invasive and a widely-available neuroimaging, which is also a good indicator of AD progression [132,141]. sMRI can be considered the preferred neuroimaging examination for AD because it allows for accurate measurement of the 3D volume of brain structures, especially the size of the hippocampus and related regions. Some examples of brain MRI images revealing the AD symptoms vs. normal patients are shown in Figure 44



Figure 44. T1-weighted MRI images. (a) Sagittal section of dilated ventricular system of AD patient, compared to ventricular system of normal subject. (b) Coronal section through the hippocampus. AD patients have shrunken hippocampus and enlarged ventricles relative to healthy age-matched controls [10,11].

In voxel-based features, after registration of all brain, each image is assumed as a high-dimensional vector with each voxel as a feature. Klöppel et al. [142] used the voxels of the gray matter (GM) as the features and trained a Support Vector Machine (SVM) for classification of AD and NC subjects. In [143], the brain volume is segmented to GM, white matter (WM), and cerebrospinal fluid (CSF), followed by voxel-wise density calculation for each material. Thus, each voxel has a three-valued feature representing tissue density of GM, WM and CSF used for classification. For feature extraction method based on cortical thickness, Lerch et al. [144] segmented the registered brain MRI into GM, WM, and CSF. Then GM and WM surfaces are fitted

using deformable models. A surface deformation algorithm is then used to expand WM surface to GM-CSF intersection to calculate the distance between corresponding vertices of WM and GM surfaces as a measure of cortical thickness, and these features are used for classification. In the third method, the shape characteristic of hippocampus is modeled for feature extraction. Gerardin et al. [145] segmented the hippocampus region and spatially aligned them among various subjects. Then the shape of the hippocampus is mathematically modeled using a series of spherical harmonics. The coefficients of the series are then normalized to eliminate the effects of rotation and translation. Finally, the coefficients of the hippocampus model are used as features for training SVM classifier.

The performance of the aforementioned feature extraction methods for AD classification are evaluated and compared with several studies [132, 146–148]. It turns out that the developed approaches suffer from some limitations: *(i)* voxel-based features generate a high-dimensional and noisy features from brain sMRI, which make it feasible for classification only after designing a smoothing and feature clustering method to extract higher-level features in reduced dimension [143]; *(ii)* feature extraction based on cortical thickness and hippocampus modeling results in neglecting the correlated shape variations of the whole brain structure affected by AD in other ROI, e.g. ventricle’s volume; *(iii)* the appropriateness of the extracted features is highly dependent on a preprocessing stage due to high dimensionality, registration error, and noise, which requires the use of domain expert to design a feature engineering algorithm; *(iv)* Most of the proposed machine learning models are biased toward the dataset is used for training and testing, i.e. the feature extracted for classification are specific to the dataset.

In this section, a new deep network is proposed to learn generic and transferable features to detect AD biomarkers, and to predict task-specific classification of AD in target dataset, based on stacking generic feature extraction layers and task-specific fine-tuned layers [149]. A 3D CAE model (3D-CAE) is trained for learning generic and transferable features which represents AD biomarkers, to overcome the aforementioned limitations in feature extraction from brain sMRI for AD classification. It automatically

extracts discriminative features that capture anatomical variations of AD. The trained convolutional filters of 3D-CAE can be adapted to other domain dataset, e.g. between CADDementia and Alzheimer’s Disease Neuroimaging Initiative (ADNI) datasets. Then a 3D CNN network (3D-CNN) is developed by stacking pretrained layers of 3D-CAE, follows by fine-tuning of fully-connected layers. To boost the performance of 3D-CNN in task-specific classification, a deep supervision method [150] is employed on fully-connected layers. Due to adaptability of learned generic features, the proposed model is called Deeply Supervised Adapted 3D CNN (DSA-3D-CNN).

## D Proposed model

The proposed model for AD diagnosis composes two part; (i) Feature extraction of brain MRI using 3D Convolutional Autoencoder; (ii) Task-specific classification using deep-supervised 3D Convolutional Neural Network. Section 1 explains the architecture of 3D-CAE and the framework of AD using deeply-supervised 3D-CNN is illustrated in Section 2.

### 1 3D Convolutional autoencoder

This section explains the procedure for extracting low-dimensional feature from high-dimensional 3D image. A 3D extension of CAE in [58] is developed, as shown in Figure 45, where 3D feature maps are extracted aiming to capture the patterns of variation in a 3D image ( $i \in [1, 2, 3]$  for the diagram shown). The hidden node activities of the input image  $\mathbf{x}$  is computed,

$$\mathbf{h}^i = f(\mathbf{W}^i * \mathbf{x} + b^i) \quad (66)$$

where  $\mathbf{h}^i$  is the 3D feature map of the  $i$ -th hidden node,  $\mathbf{W}^i$  is the 3D encoding filter (input weights) of the  $i$ -th node convolved ( $*$ ) with  $\mathbf{x}$ ,  $b^i$  is the  $i$ -node bias value, which is propagated over the feature map. The activation function  $f(\cdot)$  can be selected as linear, sigmoidal or Rectified Linear unit (ReLU) functions [151]. The encoding layer generate  $k$  feature maps, which are assumed as extracted features from the input image. The reconstructed image  $\hat{\mathbf{x}}$  is computed as

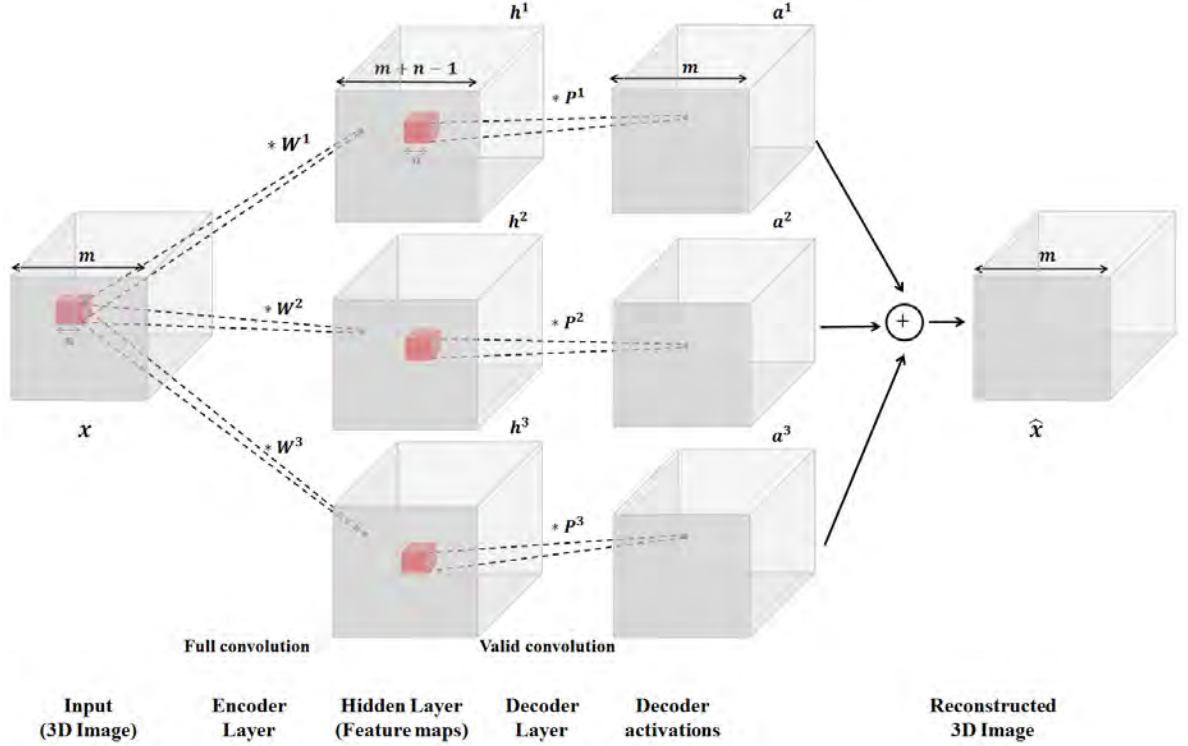


Figure 45. Schematic diagram of 3D-CAE for feature extraction of a 3D Image based on reconstructing the input. Note that the image dimension increases in the encoding layer due to full convolution, and decreases to original dimension by valid convolution.

$$\hat{\mathbf{x}} = \sum_{i=1}^k g(\mathbf{P}^i * \mathbf{h}^i + c^i) \quad (67)$$

where  $\mathbf{P}^i$  is the 3D decoding filter (output weight) convolved with the  $i$ -th feature map ( $\mathbf{h}^i$ ), and  $c^i$  is the  $i$ -th decoding bias. The 3D-CAE is trained based on minimization of the mean squared error,

$$\mathcal{L}_E(\theta) = \|\hat{\mathbf{x}} - \mathbf{x}\|_2^2 \quad (68)$$

where  $\theta$  is the parameters of 3D-CAE, i.e.  $\{\mathbf{P}^i, \mathbf{W}^i, b^i, c^i\}$ , and  $\|\cdot\|_2^2$  denotes the Frobenius norm summed over all training data. To reduce the number of free parameters in the cost function minimization of Eq.(68), the 3D-CAE is used with tied weights, where  $\mathbf{P}^i = \tilde{\mathbf{W}}^i$  (flip operation over all dimensions of  $\mathbf{W}^i$ ) [58]. To minimize Eq.(68) using stochastic gradient descent, error backpropagation is used,

$$\begin{aligned}
\frac{\partial E}{\partial \mathbf{w}^i} &= x * \delta \mathbf{h}^i + \tilde{\mathbf{h}}^i * \delta \hat{\mathbf{x}}; \\
\delta \hat{\mathbf{x}} &= (\hat{x} - x) g'(\cdot); \\
\delta \mathbf{h}^i &= (\tilde{\mathbf{W}}^i * \delta \mathbf{x}) f'(\cdot)
\end{aligned} \tag{69}$$

where  $\delta \hat{\mathbf{x}}$  and  $\delta \mathbf{h}^i$  are the deltas of output and hidden activities, and  $g'(\cdot)$  and  $f'(\cdot)$  indicate the derivatives of activation functions with respect to their input.

In order to obtain translation-invariant and feature maps, a max-pooling function is applied to  $\mathbf{h}^i$ , which downsample the feature maps by extracting the maximum value of non-overlapping sub-regions. In order to extract higher level feature maps with reduced dimensions which entangle the shape variation, the output of the max-pooling function is used as the training data for CAE in the higher layer, as shown in Figure 46. The procedure of stacking the encoding layers of 3D-CAE's (3D-CAES) reduces the feature map dimensionality by half in each hierarchical layer [58].

## 2 Deeply Supervised Adaptive 3D-CNN (DSA-3D-CNN)

In this section, a 3D-CNN is developed. The proposed 3D-CNN is based on generalization of first layers and task-specification of final layers as proposed by Long et al. [149]. The developed 3D-CNN is composed two parts: (i) Pretrained generic feature extraction layers using stacked layers of pretrained 3D-CAE; (ii) Fine-tuned fully-connected layers for task-specific classification, i.e. for multi-class or binary classification of sMRI.

The 3D-CNN is composed of stacked encoding layer of pretrained 3D-CAE for generic feature extraction, connected by fully-connected layers for task-specific classification. The extracted feature maps from the encoding layer of the highest layer of 3D-CAE are used as features yielding AD biomarkers used for classifying brain sMRI's. In order to create a feature vector for classification, the extracted  $k$  feature maps of the highest layer, e.g. 3rd 3D-CAE in Figure 45(b), are vectorized and concatenated together (as shown in concatenation layer in Figure 46). Thus, a high-dimensional brain sMRI is represented as a low-dimensional feature vector which possibly captures its anatomical shape variations. To discriminate between brain

sMRI’s for AD, MCI, and NC group, several fully connected layers are stacked and fine tuned for any task specific classification.

$$\mathcal{NLL}_s(\theta_s, \mathcal{D}_s) = - \sum_{i=0}^{|\mathcal{D}_s|} \log P(Y = y^{(i)} | x^{(i)}, \theta) \quad (70)$$

where  $\mathcal{NLL}_s$  is the task-specific negative log-likelihood cost function,  $\theta_s$  and  $\mathcal{D}_s$  are the weights of the fully connected layers and training data of a task-specific classification, respectively.

In order to boost the performance of the 3D-CNN network for task-specific classification, the method of deep supervision proposed in [150] is used, where several negative log likelihood loss are used for fully connected and softmax layers, to improve the discriminativeness of specific feature learning, as shown in Figure 46. Therefore, the proposed method is called Deeply Supervised and Adapted 3D-CNN Network (DSA-3D-CNN) and it is trained using the combined negative-log likelihood loss,

$$\mathcal{L}_{\mathcal{D}_s}(W_s) = \sum_{l=0}^n \lambda^{(l)} \mathcal{NLL}_s(\mathbf{W}_s^{(l)}) + \mathcal{NLL}_s(\mathbf{W}_s^o) \quad (71)$$

where  $\mathcal{L}_{\mathcal{D}_s}$  is the loss of DSA-3D-CNN,  $\lambda^l$  is the loss penalty of  $l$ -th fully connected layer,  $\mathbf{W}_s^l$  and  $\mathbf{W}_s^o$  are the weights of fully-connected and output layers, respectively.

The DSA-3D-CNN uses ReLu activation function in the convolution and fully connected layers, and a softmax output layer (Figure 46) to predict the probability of brain sMRI belonging to AD, MCI or NC group.

## E Experiment

The performance of the proposed DSA-3D-CNN model is evaluated for Alzheimer diagnosis of 210 subjects in ADNI dataset. The demographic information of the selected subjects from ADNI dataset is shown in Table. 8. Five classification tasks are considered, i.e. four binary classifications of AD vs. NC (AD/NC), AD and MCI vs. NC(AD+MCI/NC), AD vs. MCI (AD/MCI), MCI vs NC (MCI/NC), and 3-way classification of AD vs. MCI vs. NC (AD/MCI/NC). *Ten-fold crossvalidation* is used to evaluate the test classification accuracy. Theano [127] library is used to develop the

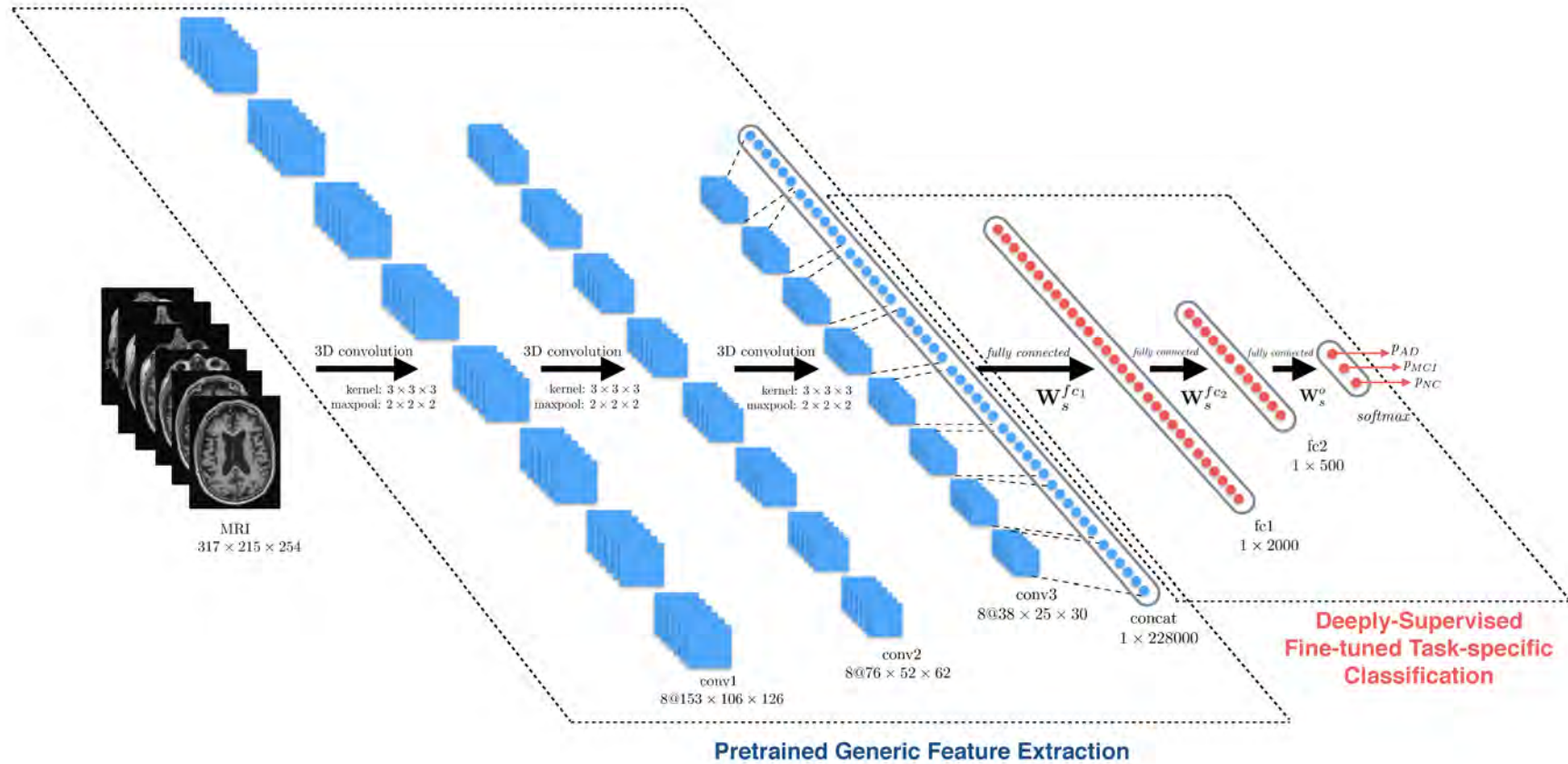


Figure 46. Proposed 3D-CNN network for AD.

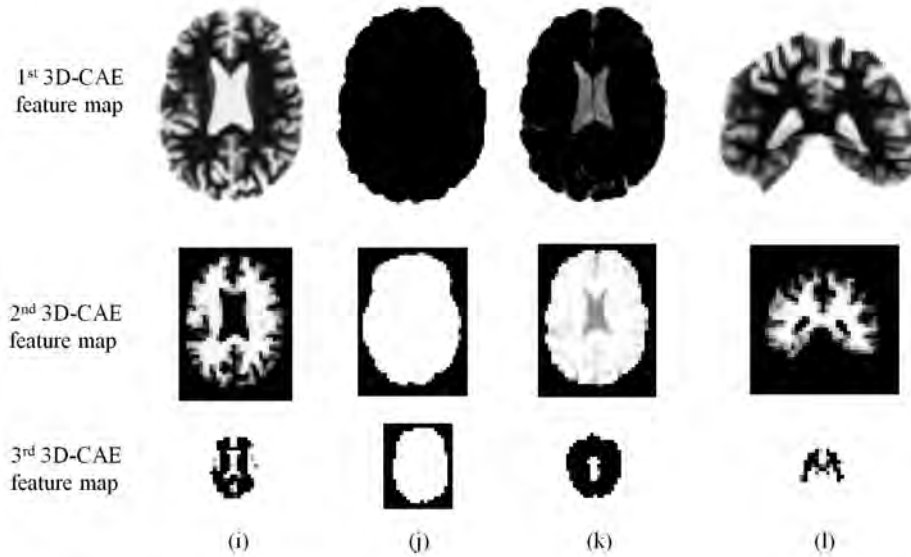


Figure 47. Selected slices of hierarchical 3D feature maps in (i,j,k) axial and (l) sagittal view extracted at 3 layers of stacked 3D-CAE indicating (i) cortex thickness and volume, (j) brain size, (k) ventricle size, and (l) hippocampus model, extracted from the brain structural MRI. The feature maps are downsampled at each layer using max-pooling to reduce the size and detect higher level features.

deep network, and Amazon EC2 g2.8xlarge instances with GPU GRID K520 for the experiments.

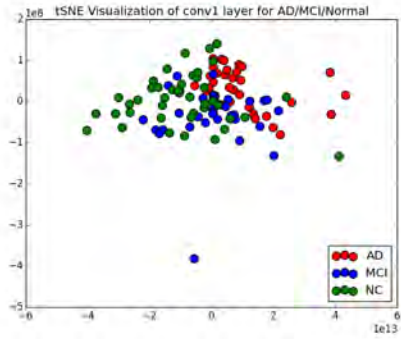
TABLE 8. Demographic information of 210 studied subjects from the ADNI dataset.

Diagnosis	AD	MCI	NC
Subject number	70	70	70
Male/Female	36/34	50/20	37/33
Age (Mean $\pm$ SD)	75.01 $\pm$ 7.87	75.90 $\pm$ 7.65	74.63 $\pm$ 6.07

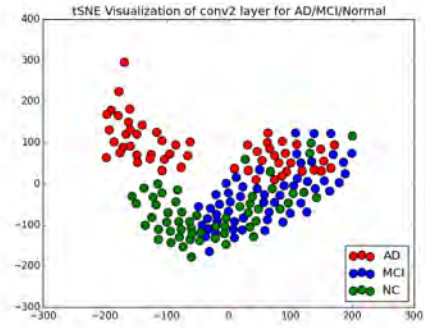
Note: Values are denoted as mean  $\pm$  deviation; M and F represent male and female, respectively.

## 1 Generic and task-specific feature evaluation:

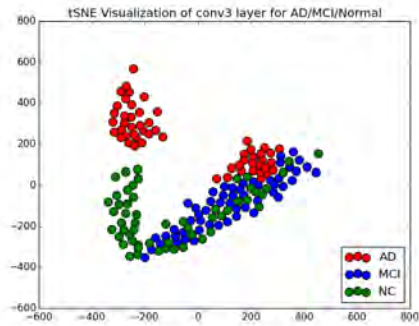
In order to evaluate the generalization of pretrained layers and specification of fine-tuned layers of DSA-3D-CNN, the extracted features in each layer are visualized. To evaluate the ability of learned generic convolutional filters in capturing features related to AD biomarkers, e.g. ventricle size, cortex thickness, and hippocampus model, selected slices of the three feature maps from each layer of 3D-CAES are shown



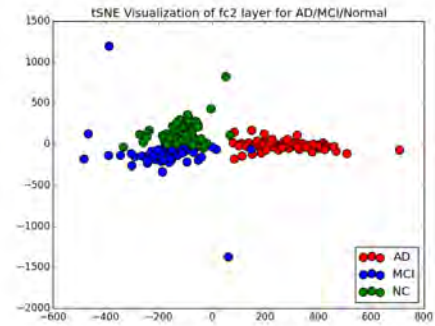
(a) conv1 hidden



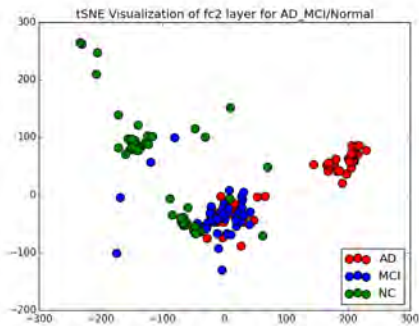
(b) conv2 hidden



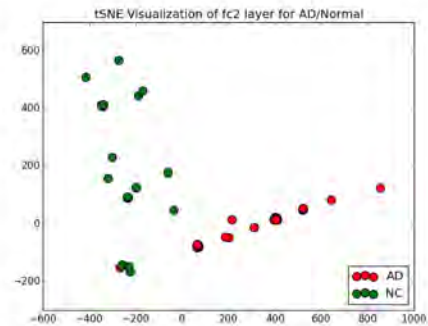
(c) conv3 hidden



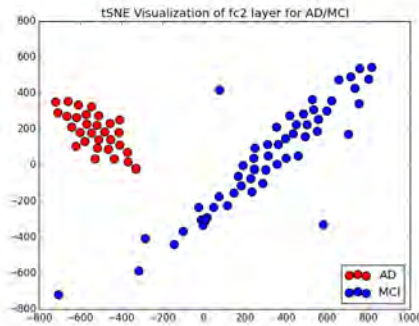
(d) fc2 hidden - AD/MCI/NC



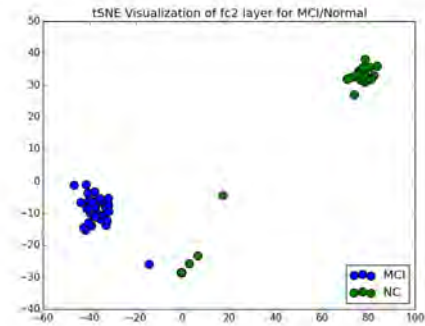
(e) fc2 hidden - AD+MCI/NC



(f) fc2 hidden - AD/NC



(g) fc2 hidden - AD/MCI



(h) fc2 hidden - MCI/NC

Figure 48. Manifold visualization of training data in (a,b,c) pretrained generic layers and (d,e,f,g,h) fine-tuned task-specific layers, using t-SNE projection [7].

in Figure 47. The featuremaps are generated by pretrained 3D-CAE on ADNI dataset. According to Figure 47(i), the cortex thickness is extracted in the first layer of 3D-CAES as a discriminative feature of AD. Figure 47(j,k,l) depict the brain size (related to patient gender), ventricles size, and hippocampus model, respectively. In each layer of 3D-CAES, the extracted feature maps of the lower layer are combined to train higher-level features characterizing the anatomical variation in the brain sMRI. According to Figure 47, features of ventricle size and cortex thickness are combined to extract a conceptually higher feature in higher layers. To express the ability of the extracted features in discriminating brain sMRI of AD patients from NC, the images are visualized using low-dimensional features extracted at the third layer of 3D-CAES (Figure 47), which indicates that AD brain sMRI's are differentiated from NC images in the low-dimensional feature space. It means that extracting higher-level features in higher layers of 3D-CAES contributes to distinguish brain sMRI's of AD from NC group.

To evaluate the discriminativeness of generic and task-specific features, the distributions of training ADNI MRI's are visualized in hidden layers of DSA-3D-CNN on Figure 48. The manifold distribution generated by generic layers, i.e. conv1, conv2, and conv3 layers (Figure 48(a-c)) indicates the gradual discrimination of AD, MCI and NC in hierarchical layers. Moreover, the subsequent layers of task-specific classification encompass the discrimination between the target classes in the dataset, i.e. ADNI, as shown in Figure 48(d-h). To highlight the generated task-specific features, Figure 48(e) depicts the distribution of three classes for AD\_MCI vs. NC classification, where AD and MCI brain MRI are projected in closer distances than NC data.

Finally, the manifold distribution of three classes in test mode is shown for AD vs. MCI vs. NC classification model is shown in Figure 49, which indicates the discriminativeness of learned features to distinguish between different classes. The manifold distribution of subjects in Figure 49 indicates the correlation of disease severity with the extracted features, where the most severe AD subjects are located at the right-most of the AD manifold, and the most normal NC subjects are located to the bottom of NC manifold.

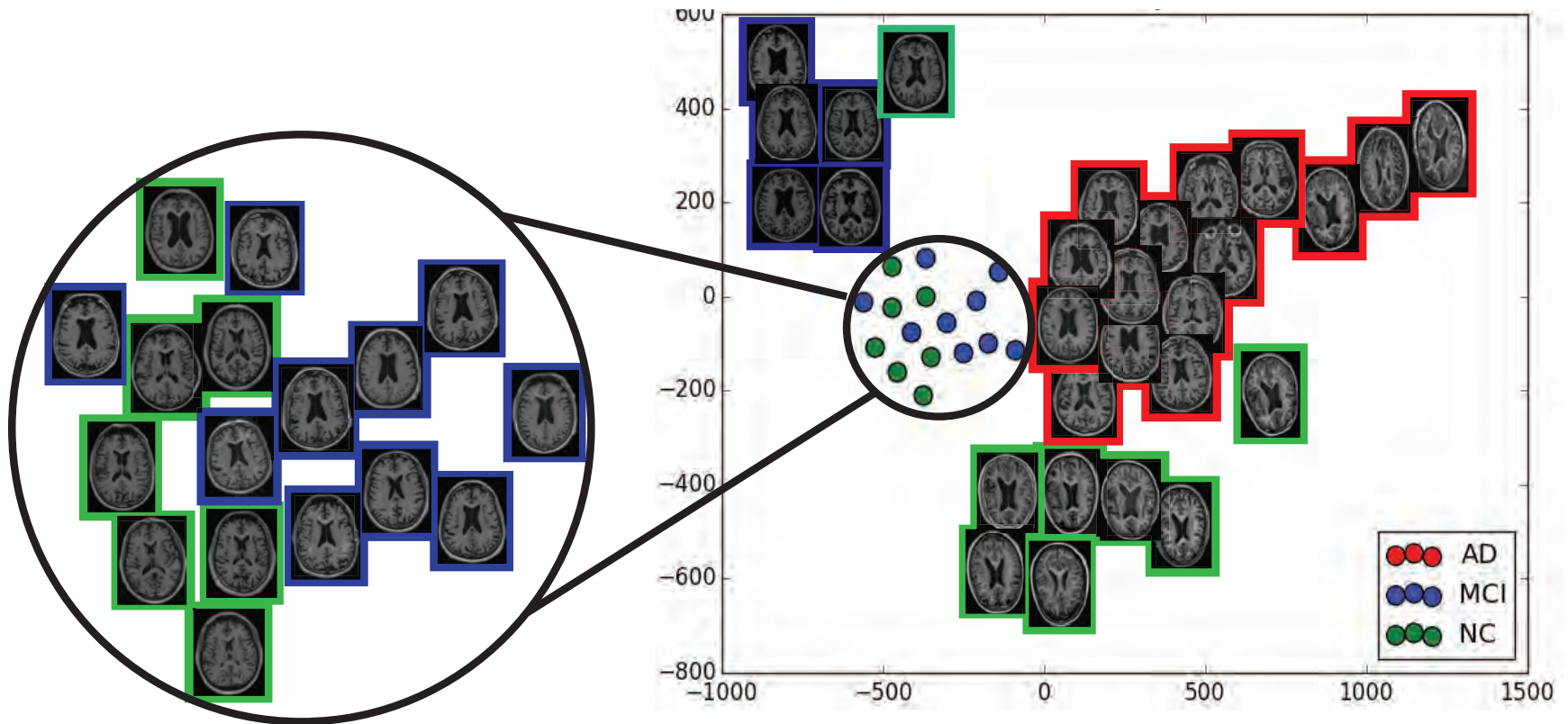
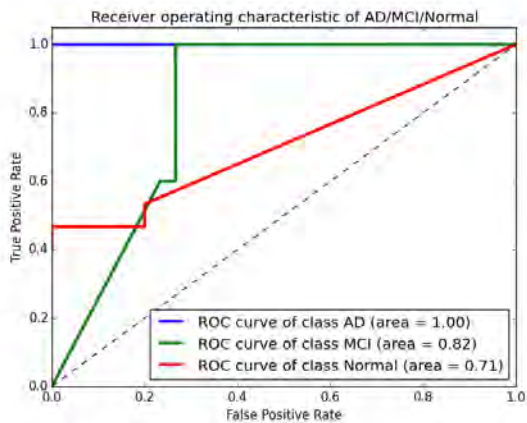
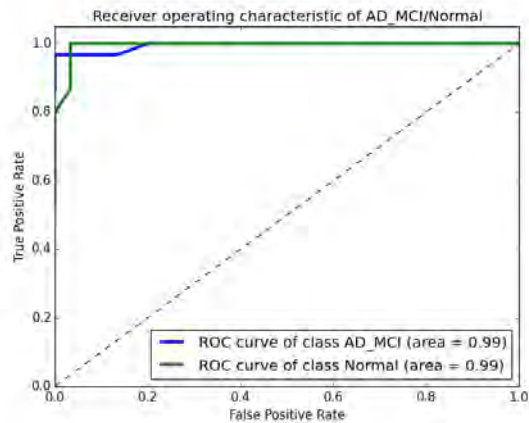


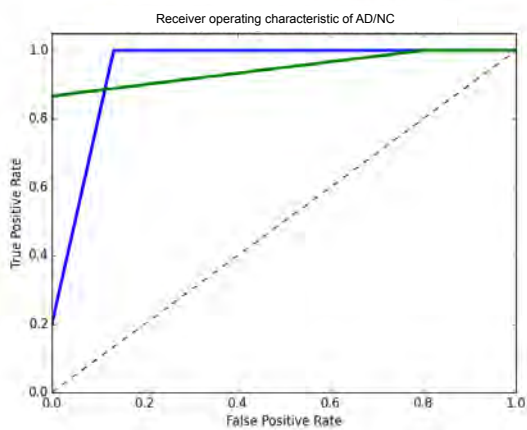
Figure 49. Manifold visualization of train data in (a,b,c) pretrained generic layers and (d,e,f,g,h) fine-tuned task-specific layers, using t-SNE projection [7].



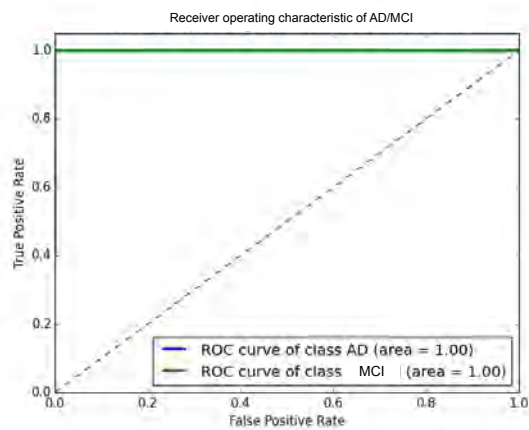
(a)



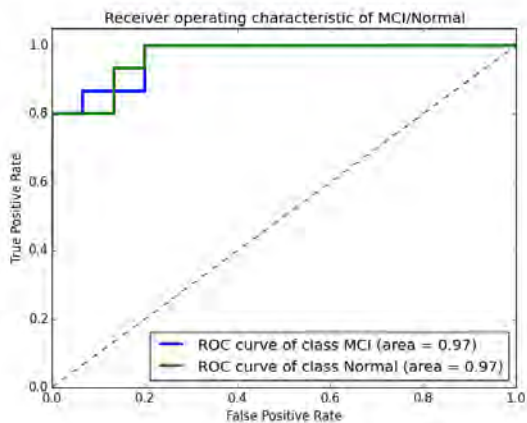
(b)



(c)



(d)



(e)

Figure 50. Receiver Operating Curve (ROC) and Area Under Curve (AUC) performance of (a) AD/MCI/NC, (b) AD+MCI/NC, (c) AD/NC, (d) AD/MCI and (e) MCI/NC task-specific classification models.

TABLE 9

Task-Specific Classification Models Performance for a selected fold of *crossvalidation* for the proposed model.

Class	AD/MCI/NC			AD+MCI/NC			AD/NC			AD/MCI			MCI/NC		
	Precision	Recall	F1-score	Precision	Recall	F1-score	Precision	Recall	F1-score	Precision	Recall	F1-score	Precision	Recall	F1-score
AD	1.00	1.00	1.00	-	-	-	0.88	1.00	0.94	1.00	1.00	1.00	-	-	-
MCI	0.60	0.80	0.69	-	-	-	-	-	-	1.00	1.00	1.00	0.83	1.00	0.91
AD+MCI	-	-	-	0.94	0.97	0.95	-	-	-	-	-	-	-	-	-
NC	0.70	0.47	0.56	0.93	0.87	0.90	1.00	0.87	0.93	-	-	-	1.00	0.80	0.89
Average	0.77	0.76	0.75	0.93	0.93	0.93	0.94	0.93	0.93	1.00	1.00	1.00	0.92	0.90	0.90

TABLE 10

Classification performance evaluation of the proposed model [mean(std)%]

Task	ACC	SEN	SPE	BAC	PPV	NPV	AUC	F1-score
AD/MCI/NC	94.76(2.60)	-	-	-	-	-	-	-
AD+MCI/NC	95.72(3.10)	94.8(4.08)	97.20(3.83)	96.00(2.89)	98.40(2.19)	91.00(6.81)	96.07(2.93)	93.88(4.36)
AD/NC	99.28(1.59)	100(0)	98.60(3.13)	99.30(1.56)	98.60(3.13)	100(0)	99.28(1.95)	99.40(1.34)
AD/MCI	100(0)	100(0)	100(0)	100(0)	100(0)	100(0)	100(0)	100(0)
MCI/NC	91.43(1.95)	86.00(5.71)	96.50(4.04)	91.25(2.02)	96.25(4.34)	87.50(4.50)	91.43(1.95)	91.81(1.65)

TABLE 11

Performance comparison (ACC%) of the competing methods.

Models	Modalities	Task-Specific Classification [mean(std)%]				
		AD/MCI/NC	AD+MCI/NC	AD/NC	AD/MCI	MCI/NC
(1) Suk et al. [135]	PET+MRI+CSF	-	-	0.959(0.011)	-	0.850(0.012)
(2) Suk et al. [136]	PET+MRI	-	-	95.35(5.23)	-	85.67(5.22)
(3) Zhu et al. [152]	PET+MRI+CSF	-	-	95.9	-	82
(4) Zu et al. [153]	PET+MRI	-	-	95.95	-	80.26
(5) Liu et al. [137]	PET+MRI	53.79(4.76)	-	91.40(5.56)	-	82.10(4.91)
(6) Liu et al. [133]	MRI	-	-	93.83	-	89.09
(7) Li et al. [138]	PET+MRI+CSF	-	-	91.4(1.8)	70.1(2.3)	77.4(1.7)
Proposed	MRI	<b>94.78(2.60)</b>	<b>95.72(3.10)</b>	<b>99.28(1.42)</b>	<b>100(0)</b>	<b>91.43(1.95)</b>

## 2 Classification performance evaluation

In this section, the classification performance of each task-specific classification of the proposed model in Section 1 is evaluated and also compared to the competing models [133, 135–138, 152]. Let TP, TN, FP, and FN denote, respectively, True Positive, True Negative, False Positive, and False Negative. The classification performance of each task-specific model is measured with seven metrics [154]:

- ACCuracy (ACC) =  $(TP + TN) / (TP + TN + FP + FN)$
- SENSitivity (SEN) =  $TP / (TP + FN)$
- SPEcificity (SPE) =  $TN / (TN + FP)$
- Balanced ACCuracy (BAC) =  $(SEN + SPE) / 2$
- Positive Predictive Value (PPV) =  $TP / (TP + FP)$
- Negative Predictive Value (NPV) =  $TN / (TN + FN)$
- F1-score =  $2TP / (2TP + FP + FN)$
- Area Under the receiver operating characteristic Curve (AUC)

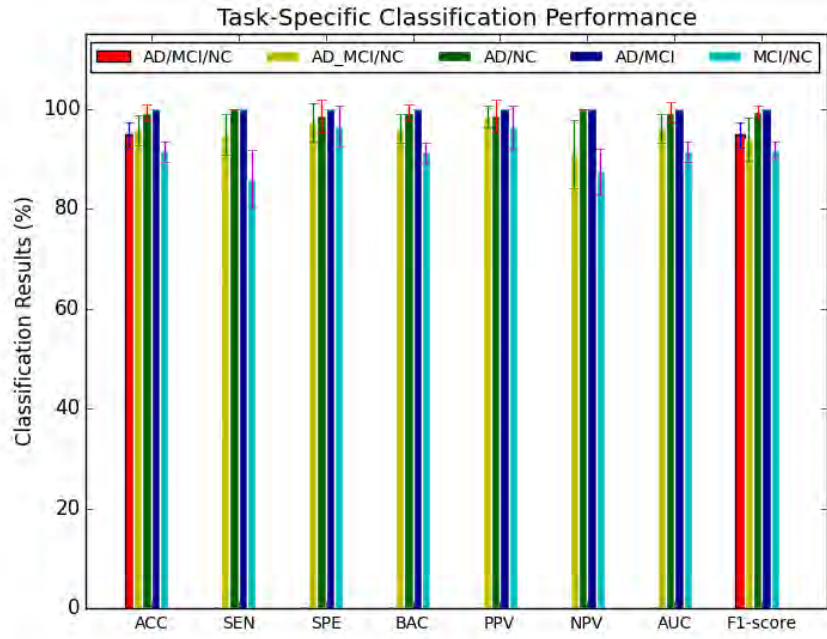
Table 9 indicates the precision, recall and F1-score for each classes within each task-specific classification of DSA-3D-CNN of a selected fold of *crossvalidation*. To measure the robustness of task-specific classification models, ROC and AUC of the selected fold is also shown in Figure 50, where it indicates the robustness and high-confidence of predicted classes within each model. Table 10 lists the mean and standard deviation of the seven metrics for each task-specific model, while Figure 51(a) also depicts the values as error bar.

The ACC performance of the proposed method is compared to several alzheimer’s diagnosis models, with the comparison of imaging modalities or clinical data employed by each method in Table 11 and Figure 51(b). The results of the proposed model are the average of *ten-fold crossvalidation* experiments. They indicate

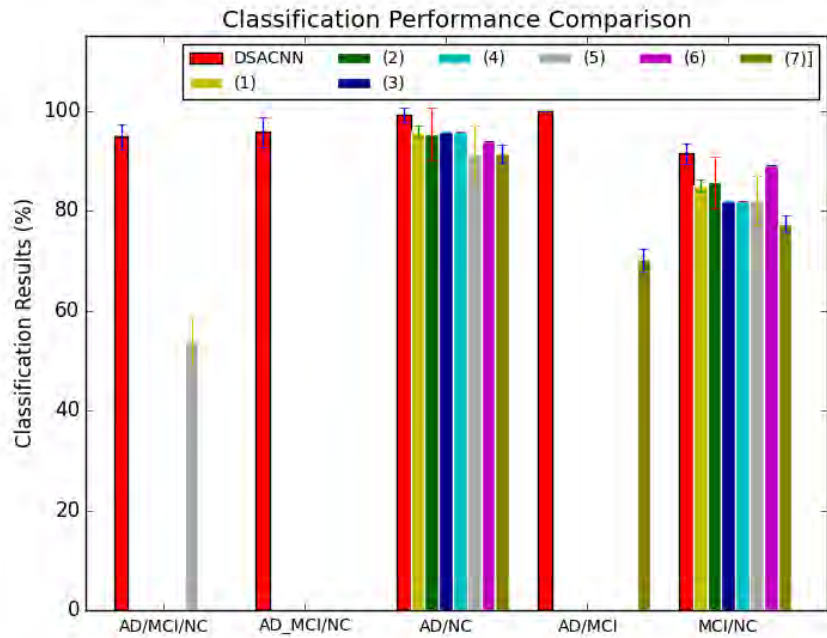
that DSA-3D-CNN achieves better accuracy performance in five task-specific classifications, where a single image modality (sMRI) with no skull-stripping is used.

## F Conclusion

In this chapter, a 3D-CNN is proposed for AD classification on structural brain MRI scans. To enhance the generality of features in capturing AD biomarkers, a transfer learning approach was used, where three stacked 3D-CAE network were pretrained on CADDementia Dataset. Then the learned features are extracted and used as AD biomarkers detectors in lower layers of a 3D-CNN network. Then three fully connected layers are stacked on top of the lower layers to perform AD classification on 210 subjects of ADNI dataset. To boost the classification performance, negative log-likelihood loss was imposed on the fully connected hidden layers, in addition to the output layers. The results demonstrate that hierarchical feature extraction improved in hidden layers of 3D-CNN by discriminating between AD, MCI, and NC subjects, (Table 10). Seven classification performance measures were computed using *ten-fold crossvalidation*, were compared to the state-of-the-art models, and demonstrated the out-performance of the proposed 3D-CNN.



(a)



(b)

Figure 51. Averaged classification results achieved by different methods.

## CHAPTER VII

### CONCLUSION AND FUTURE WORK

In this work, sparse feature learning has been evaluated for lung segmentation, part-based feature extraction for image classification, and AD classification. The results are summarized in Sections A, B, and C, respectively.

#### A Lung Segmentation

A novel lung segmentation framework was proposed using NMF as the unsupervised feature learning part [2]. NMF was used to extract spatial features of voxels from CT scans. The extracted features demonstrated the discrimination and separability between lung and chest voxels in the learned space. Then a K-means clustering algorithm was applied to group the voxels in two clusters. The group of voxels belonging to lung was indicated by the distance of the cluster center to the origin. Since lung voxels have lower intensity value, their cluster center in the feature space is closer to the origin.

There are initially unknown number of clusters representing main objects in CT scans, such as lungs, chest tissues, veins, arteries etc. The main limitation of NMF-based approach was that the actual number of body organs should be known prior to feature extraction and clustering. Moreover, due to the slice-wise segmentation, the inter-slice signal dependencies were taken into account only implicitly, via the context. To overcome the NMF-based approach limitations, a modified incremental learning algorithm was proposed for NMF (INMF) that automatically detect the number of clusters in the feature space using voxel-by-voxel mode [1]. Then K-means algorithm was applied based on the computed number of clusters by INMF, to group the voxels. The results demonstrated that different objects

such as tumor and cancer can be detected and segmented. The final lung segmentation was computed by merging the small clusters based on similarity distance, and forming two major clusters representing lung and chest.

INMF-based approach turned out to be computationally expensive for large 3D images due to voxel-by-voxel modeling. Moreover, the number of clusters are highly sensitive to the distance parameter of the algorithm, which is directly related to the sparseness and smoothness of the learned feature space. To overcome the INMF-based approach limitations, a new slice-wise incremental learning was combined with constrained NMF (ICNMF) [6]. The method applies smoothness constraints to learn the features, which are more robust to lung tissue inhomogeneities and thus help to better segment internal lung pathologies than the known state-of-the-art techniques. Compared to the latter, the ICNMF depends less on the domain expert knowledge and is more easily tuned due to only a few control parameters. Also, the proposed slice-wise incremental learning with due regard for inter-slice signal dependencies decreases the computational complexity of the NMF-based segmentation and is scalable to very large 3D lung images.

The ICNMF method was quantitatively validated on simulated realistic lung phantoms that mimic different lung pathologies (7 datasets), *in vivo* data sets for 17 subjects, and 55 data sets from the LOLA11 study. For the *in vivo* data, the accuracy of ICNMF segmentation w.r.t. the ground truth is 0.96 by the DSC, 9.0 *mm* by the MHD, and 0.87% by the ALVD, which is significantly better than for the NMF-based segmentation. In spite of not being designed for lungs with severe pathologies and of no agreement between radiologists on the ground truth in such cases, the ICNMF with its total accuracy of 0.965 was ranked 5<sup>th</sup> among all others in the LOLA11. After excluding the nine too pathological cases from the LOLA11 dataset, the ICNMF accuracy increased to 0.986.

## B Image Classification

To improve image classification performance in deep models, interpretability was considered as a key characteristic. Autoencoder (AE) [16,108] and Convolutional

Autoencoder (CAE) were investigated for interpretable feature learning in deep models. The goal was to extract meaningful representation of high-dimensional data.

Part-based feature extraction was defined as the meaningful and interpretable characteristic of deep model, and autoencoder was employed as the core feature extraction model [16, 108]. Inspired by the idea of sparsity and NMF, nonnegativity was used as a constraint in learning autoencoder (NCAE). The performance of the novel NCAE demonstrated in extracting parts of objects, e.g. digit, face, object, or semantic features of text data. A deep model was created from stacked layers of NCAE, and then fine-tuned using labeled data. To impose part-based decomposition in hidden layers of deep model during fine-tuning, a nonnegativity constrained back-propagation algorithm was proposed. The trained stacked NCAE demonstrated better performance than deep models using stacked layers of SAE, DpAE, and DAE.

To impose part-based feature extraction in CAE, a novel structured sparsity was proposed. Due to the convolutional layer, the learned featuremaps yield non-sparse features, reflecting spatial interaction of neighboring neurons in hidden layer. To overcome this problem, a structured normalization approach was proposed using  $\ell_2$  and  $\ell_1$  constraints. Using the proposed SSCAE, the feature maps were normalized to have a unit- $\ell_2$  norm, and  $\ell_1$  norm of the featuremap was minimized to break the spatial interaction of neighboring neurons in hidden featuremaps. The results demonstrated that objects were decomposed into parts across featuremap, with improved reconstruction and learning rate convergence than CAE.

## C Disease Diagnosis

To develop an accurate AD classification, a new 3D-CNN was proposed for AD diagnosis from structural MRI scans [17]. To incorporate a new knowledge sharing across domains, two dataset for AD classification, i.e. CADDementia and ADNI, were selected. CADDementia dataset was used for extracting features related to different AD biomarkers, e.g. cortical thickness, hippocampus shape, and ventricular size. 3D-CAE was used for extracting the aforementioned features in three stacked layers. Then, the trained 3D-CAES were used as feature extraction layers in lower layers of

3D-CNN. Then, three fully connected layers are initialized and stacked as higher layers to the aforementioned layers. For each task-specific classifications, i.e. AD vs NC, AD vs MCI, MCI vs NC, and AD and MCI vs NC, the higher fully connected layers were fine-tuned with labeled data from ADNI dataset.

Due to layer-wise training, i.e. pretraining of lower convolutional layers on CADDementia MRI data, and fine-tuning of higher fully connected layers on ADNI MRI data, the proposed 3D-CNN demonstrated superior performance than recently developed models. The main advantage of the proposed 3D-CNN was the use of a single image modality of structural MRI, compared to other multi-modality approaches, and no skull-stripping preprocess in AD classification.

#### D Future work

The future application of the proposed NMF-based segmentation of lung combined with proposed convolutional neural network for AD diagnosis can be extended to the following problems,

- Testing the proposed segmentation and diagnosis model in other clinical studies, such as detection of lung cancer using CE computed tomography (CE-CT) images [70, 72, 73, 75, 83, 155–178], brain cancer using DCE-MRI [179–215], kidney transplantation prediction using DW-MRI [216–237], prostate cancer detection from DCE- and DW-MRI [238–249], and colon cancer using CE-CT images.
- A future work of this dissertation is to investigate the integration of the proposed work with the BioImaging lab work for the detection of other brain disorders such as autism [192, 196, 200, 204–206, 209, 212–214, 250–254] and dyslexia [179, 197–199, 201–203, 207, 255].

## REFERENCES

- [1] E. Hosseini-Asl, J. M. Zurada, and A. El-Baz, “Automatic segmentation of pathological lung using incremental nonnegative matrix factorization,” in *Image Processing (ICIP), 2015 IEEE International Conference on*, 2015, pp. 3111–3115.
- [2] ———, “Lung segmentation based on nonnegative matrix factorization,” in *Image Processing (ICIP), 2014 IEEE International Conference on*, Oct 2014, pp. 877–881.
- [3] A. Ng, “Sparse autoencoder,” in *CS294A Lecture notes*. URL [http://www.stanford.edu/class/cs294a/sparseAutoencoder\\_2011new.pdf](http://www.stanford.edu/class/cs294a/sparseAutoencoder_2011new.pdf): Stanford University, 2011.
- [4] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
- [5] Y. Bengio, I. J. Goodfellow, and A. Courville, “Deep learning,” 2015, book in preparation for MIT Press. [Online]. Available: <http://www.iro.umontreal.ca/bengioy/dlbook>
- [6] E. Hosseini-Asl, J. Zurada, G. Gimel farb, and A. El-Baz, “3D lung segmentation by incremental constrained nonnegative matrix factorization,” *Biomedical Engineering, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2015. doi: 10.1109/TBME.2015.2482387
- [7] L. Van der Maaten and G. Hinton, “Visualizing data using t-SNE.” *Journal of Machine Learning Research*, vol. 9, no. 11, 2008.
- [8] A. Lemme, R. F. Reinhart, and J. J. Steil, “Online learning and generalization of parts-based image representations by non-negative sparse autoencoders,” *Neural Networks*, vol. 33, pp. 194–203, 2012.

- [9] J. Ngiam, Z. Chen, S. A. Bhaskar, P. W. Koh, and A. Y. Ng, “Sparse filtering,” in *Advances in Neural Information Processing Systems*, 2011, pp. 1125–1133.
- [10] C. R. Jack, M. A. Bernstein, N. C. Fox, P. Thompson, G. Alexander, D. Harvey, B. Borowski, P. J. Britson, J. L Whitwell, C. Ward *et al.*, “The Alzheimer’s disease neuroimaging initiative (ADNI): MRI methods,” *Journal of Magnetic Resonance Imaging*, vol. 27, no. 4, pp. 685–691, 2008.
- [11] P. Scheltens, N. Fox, F. Barkhof, and C. De Carli, “Structural magnetic resonance imaging in the practical assessment of dementia: beyond exclusion,” *The Lancet Neurology*, vol. 1, no. 1, pp. 13 – 21, 2002.
- [12] Y. Bengio, A. Courville, and P. Vincent, “Representation learning: A review and new perspectives,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 35, no. 8, pp. 1798–1828, 2013.
- [13] D. D. Lee and H. S. Seung, “Learning the parts of objects by non-negative matrix factorization,” *Nature*, vol. 401, no. 6755, pp. 788–791, 1999.
- [14] G. E. Hinton and R. R. Salakhutdinov, “Reducing the dimensionality of data with neural networks,” *Science*, vol. 313, no. 5786, pp. 504–507, 2006.
- [15] Y. Bengio, “Learning deep architectures for AI,” *Foundations and trends in Machine Learning*, vol. 2, no. 1, pp. 1–127, 2009.
- [16] E. Hosseini-Asl, J. Zurada, and O. Nasraoui, “Deep learning of part-based representation of data using sparse autoencoders with nonnegativity constraints,” *Neural Networks and Learning Systems, IEEE Transactions on*, vol. PP, no. 99, pp. 1–13, 2015. doi: 10.1109/TNNLS.2015.2479223
- [17] E. Hosseini-Asl, J. M. Zurada, and A. El-Baz, “A deeply supervised adapted 3D convolutional network for alzheimer diagnosis,” in *in preparation for submission to IEEE Transactions on Pattern Recognition and Machine Intelligence*.

- [18] P. O. Hoyer, “Non-negative sparse coding,” in *Proceedings of the 12th IEEE Workshop on Neural Networks for Signal Processing*. IEEE, 2002, pp. 557–565.
- [19] G. F. Harpur and R. W. Prager, “Development of low entropy coding in a recurrent network,” *Network: computation in neural systems*, vol. 7, no. 2, pp. 277–284, 1996.
- [20] B. A. Olshausen and D. J. Field, “Sparse coding with an overcomplete basis set: A strategy employed by V1?” *Vision research*, vol. 37, no. 23, pp. 3311–3325, 1997.
- [21] D. D. Lee and H. S. Seung, “Algorithms for non-negative matrix factorization,” in *Advances in neural information processing systems*. MIT Press, 2000, pp. 556–562.
- [22] C. Févotte, N. Bertin, and J. L. Durrieu, “Nonnegative matrix factorization with the itakura-saito divergence: With application to music analysis,” *Neural computation*, vol. 21, no. 3, pp. 793–830, 2009.
- [23] I. S. Dhillon and S. Sra, “Generalized nonnegative matrix approximations with bregman divergences,” in *NIPS*, vol. 18, 2005.
- [24] R. Kompass, “A generalized divergence measure for nonnegative matrix factorization,” *Neural computation*, vol. 19, no. 3, pp. 780–791, 2007.
- [25] E. Hosseini-Asl and J. M. Zurada, “Nonnegative matrix factorization for document clustering: A survey,” in *Artificial Intelligence and Soft Computing*, ser. Lecture Notes in Computer Science, vol. 8468. Springer International Publishing, 2014, pp. 726–737.
- [26] R. Zdunek and A. Cichocki, “Non-negative matrix factorization with quasi-newton optimization,” in *Artificial Intelligence and Soft Computing–ICAISC 2006*. Springer, 2006, pp. 870–879.

- [27] P. O. Hoyer, “Non-negative matrix factorization with sparseness constraints,” *J. Mach. Learn. Res.*, vol. 5, pp. 1457–1469, 2004.
- [28] F. Shahnaz, M. W. Berry, V. P. Pauca, and R. J. Plemmons, “Document clustering using nonnegative matrix factorization,” *Information Processing & Management*, vol. 42, no. 2, pp. 373–386, 2006.
- [29] M. W. Berry, M. Browne, A. N. Langville, V. P. Pauca, and R. J. Plemmons, “Algorithms and applications for approximate nonnegative matrix factorization,” *Computational Statistics & Data Analysis*, vol. 52, no. 1, pp. 155–173, 2007.
- [30] C. J. Lin, “Projected gradient methods for nonnegative matrix factorization,” *Neural computation*, vol. 19, no. 10, pp. 2756–2779, 2007.
- [31] H. Kim and H. Park, “Nonnegative matrix factorization based on alternating nonnegativity constrained least squares and active set method,” *SIAM Journal on Matrix Analysis and Applications*, vol. 30, no. 2, pp. 713–730, 2008.
- [32] J. Kim and H. Park, “Fast nonnegative matrix factorization: An active-set-like method and comparisons,” *SIAM J. Sci. Comput.*, vol. 33, no. 6, pp. 3261–3281, 2011.
- [33] C. Févotte and J. Idier, “Algorithms for nonnegative matrix factorization with the  $\beta$ -divergence,” *Neural Computation*, vol. 23, no. 9, pp. 2421–2456, 2011.
- [34] W. Liu, P. P. Pokharel, and J. C. Principe, “Correntropy: properties and applications in non-gaussian signal processing,” *IEEE Trans. Signal Process.*, vol. 55, no. 11, pp. 5286–5298, 2007.
- [35] K. H. Jeong and J. C. Principe, “Enhancing the correntropy MACE filter with random projections,” *Neurocomputing*, vol. 72, no. 1, pp. 102–111, 2008.
- [36] T. Ensari, J. Chorowski, and J. M. Zurada, “Correntropy-based document clustering via nonnegative matrix factorization,” in *Artificial Neural Networks and Machine Learning–ICANN 2012*. Springer, 2012, pp. 347–354.

- [37] —, “Occluded face recognition using correntropy-based nonnegative matrix factorization,” in *11th International Conference on Machine Learning and Applications (ICMLA)*, vol. 1. IEEE, 2012, pp. 606–609.
- [38] M. Schmidt, “Matlab software,” URL [www.di.ens.fr/~mschmidt/Software/minConf.html](http://www.di.ens.fr/~mschmidt/Software/minConf.html), 2008.
- [39] L. Du, X. Li, and Y. D. Shen, “Robust nonnegative matrix factorization via half-quadratic minimization,” in *ICDM*, 2012, pp. 201–210.
- [40] E. Hosseini-Asl and J. M. Zurada, “Multiplicative algorithm for correntropy-based nonnegative matrix factorization,” *Journal of Applied Computer Science Methods*, vol. 5, no. 2, pp. 89–104, 2013.
- [41] Y. Bengio and Y. LeCun, “Scaling learning algorithms towards AI,” *Large-Scale Kernel Machines*, vol. 34, pp. 1–41, 2007.
- [42] L. Deng, “A tutorial survey of architectures, algorithms, and applications for deep learning,” *APSIPA Transactions on Signal and Information Processing*, vol. 3, p. e2, 2014.
- [43] S. Bengio, L. Deng, H. Larochelle, H. Lee, and R. Salakhutdinov, “Guest editors’ introduction: Special section on learning deep architectures,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 35, no. 8, pp. 1795–1797, 2013.
- [44] B. Hutchinson, L. Deng, and D. Yu, “Tensor deep stacking networks,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 35, no. 8, pp. 1944–1957, 2013.
- [45] Y. Bengio, P. Lamblin, D. Popovici, H. Larochelle *et al.*, “Greedy layer-wise training of deep networks,” *Advances in neural information processing systems*, vol. 19, p. 153, 2007.

- [46] G. E. Hinton, S. Osindero, and Y. W. Teh, “A fast learning algorithm for deep belief nets,” *Neural computation*, vol. 18, no. 7, pp. 1527–1554, 2006.
- [47] M. Ranzato, Y. L. Boureau, and Y. LeCun, “Sparse feature learning for deep belief networks,” *Advances in neural information processing systems*, vol. 20, pp. 1185–1192, 2007.
- [48] R. Raina, A. Battle, H. Lee, B. Packer, and A. Y. Ng, “Self-taught learning: transfer learning from unlabeled data,” in *Proceedings of the 24th international conference on Machine learning*. ACM, 2007, pp. 759–766.
- [49] D. Erhan, Y. Bengio, A. Courville, P. A. Manzagol, P. Vincent, and S. Bengio, “Why does unsupervised pre-training help deep learning?” *The Journal of Machine Learning Research*, vol. 11, pp. 625–660, 2010.
- [50] P. Vincent, H. Larochelle, Y. Bengio, and P. A. Manzagol, “Extracting and composing robust features with denoising autoencoders,” in *Proceedings of the 25th international conference on Machine learning*. ACM, 2008, pp. 1096–1103.
- [51] G. E. Hinton, “Training products of experts by minimizing contrastive divergence,” *Neural computation*, vol. 14, no. 8, pp. 1771–1800, 2002.
- [52] G. E. Hinton and R. S. Zemel, “Autoencoders, minimum description length, and helmholtz free energy,” *Advances in neural information processing systems*, pp. 3–3, 1994.
- [53] C. Poultney, S. Chopra, Y. L. Cun *et al.*, “Efficient learning of sparse representations with an energy-based model,” in *Advances in neural information processing systems*, 2006, pp. 1137–1144.
- [54] H. Lee, C. Ekanadham, and A. Y. Ng, “Sparse deep belief net model for visual area V2.” in *NIPS*, vol. 7, 2007, pp. 873–880.
- [55] V. Nair and G. E. Hinton, “3D object recognition with deep belief nets.” in *NIPS*, 2009, pp. 1339–1347.

- [56] Y. A. LeCun, L. Bottou, G. B. Orr, and K. R. Müller, “Efficient backprop,” in *Neural networks: Tricks of the trade*. Springer, 2012, pp. 9–48.
- [57] J. E. Moody, S. J. Hanson, A. Krogh, and J. A. Hertz, “A simple weight decay can improve generalization,” *Advances in neural information processing systems*, vol. 4, pp. 950–957, 1995.
- [58] J. Masci, U. Meier, D. Cireşan, and J. Schmidhuber, “Stacked convolutional auto-encoders for hierarchical feature extraction,” in *Artificial Neural Networks and Machine Learning–ICANN 2011*. Springer, 2011, pp. 52–59.
- [59] A. Makhzani and B. Frey, “A winner-take-all method for training sparse convolutional autoencoders,” *arXiv preprint arXiv:1409.2752*, 2014.
- [60] B. Leng, S. Guo, X. Zhang, and Z. Xiong, “3D object retrieval with stacked local convolutional autoencoder,” *Signal Processing*, vol. 112, no. 0, pp. 119 – 128, 2015.
- [61] D. H. Hubel and T. N. Wiesel, “Receptive fields and functional architecture of monkey striate cortex,” *The Journal of physiology*, vol. 195, no. 1, pp. 215–243, 1968.
- [62] S. Thrun, “Is learning the n-th thing any easier than learning the first?” *Advances in neural information processing systems*, pp. 640–646, 1996.
- [63] R. Caruana, “Multitask learning,” *Machine learning*, vol. 28, no. 1, pp. 41–75, 1997.
- [64] R. Raina, A. Y. Ng, and D. Koller, “Constructing informative priors using transfer learning,” in *Proceedings of the 23rd international conference on Machine learning*. ACM, 2006, pp. 713–720.
- [65] J. Baxter, “A bayesian/information theoretic model of learning to learn via multiple task sampling,” *Machine Learning*, vol. 28, no. 1, pp. 7–39, 1997.

- [66] J. S. Bridle and S. Cox, “Recnorm: Simultaneous normalisation and classification applied to speech recognition.” in *NIPS*. Morgan Kaufmann, 1990, pp. 234–240.
- [67] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan, “A theory of learning from different domains,” *Machine learning*, vol. 79, no. 1-2, pp. 151–175, 2010.
- [68] K. Crammer, M. Kearns, and J. Wortman, “Learning from multiple sources,” *The Journal of Machine Learning Research*, vol. 9, pp. 1757–1774, 2008.
- [69] G. Mesnil, Y. Dauphin, X. Glorot, S. Rifai, Y. Bengio, I. J. Goodfellow, E. Lavoie, X. Muller, G. Desjardins, D. Warde-Farley *et al.*, “Unsupervised and transfer learning challenge: a deep learning approach.” *ICML Unsupervised and Transfer Learning*, vol. 27, pp. 97–110, 2012.
- [70] A. El-Baz and J. S. Suri, *Lung Imaging and Computer Aided Diagnosis*. CRC Press, 2011.
- [71] A. A. Farag, A. El-Baz, G. Gimel’farb, and R. Falk, “Detection and recognition of lung abnormalities using deformable templates,” in *Proc. 17<sup>th</sup> Intern. Conf. on Pattern Recognition (ICPR 2004)*, vol. 3. IEEE, 2004, pp. 738–741.
- [72] A. El-Baz, A. A. Farag, R. Falk, and R. La Rocca, “A unified approach for detection, visualization, and identification of lung abnormalities in chest spiral CT scans,” in *International Congress Series*, vol. 1256. Elsevier, 2003, pp. 998–1004.
- [73] A. El-Baz, A. Farag, R. Falk, and R. La Rocca, “Detection, visualization and identification of lung abnormalities in chest spiral CT scan: Phase-I,” in *Proc. Intern. Conf. on Biomedical Engineering, Cairo, Egypt*, vol. 12, no. 1, 2002.
- [74] A. A. Farag *et al.*, “Appearance models for robust segmentation of pulmonary nodules in 3D LDCT chest images,” in *Proc. Intern. Conf. on Medical Image Computing and Computer-Assisted Intervention (MICCAI 2006)*. Springer, 2006, pp. 662–670.

- [75] A. El-Baz *et al.*, “Automatic analysis of 3D low dose CT images for early diagnosis of lung cancer,” *Pattern Recognition*, vol. 42, no. 6, pp. 1041–1051, 2009.
- [76] S. Hu, E. A. Hoffman, and J. M. Reinhardt, “Automatic lung segmentation for accurate quantitation of volumetric x-ray ct images,” *Medical Imaging, IEEE Transactions on*, vol. 20, no. 6, pp. 490–498, 2001.
- [77] Y. Itai, H. Kim, S. Ishikawa, A. Yamamoto, and K. Nakamura, “A segmentation method of lung areas by using snakes and automatic detection of abnormal shadow on the areas,” *International Journal of Innovative Computing, Information, and Control*, vol. 3, no. 2, pp. 277–284, 2007.
- [78] M. Silveira, J. Nascimento, and J. Marques, “Automatic segmentation of the lungs using robust level sets,” in *Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE*. IEEE, 2007, pp. 4414–4417.
- [79] C. H. Won, D. H. Kim, J. H. Lee, K. W. Yoon, S. H. Woo, Y. H. Yoon, M. K. Kim, and J. H. Cho, “Lung segmentation by new curve stopping function using geodesic active contour model,” *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. 89, no. 6, pp. 1727–1729, 2006.
- [80] Y. Shi, F. Qi, Z. Xue, L. Chen, K. Ito, H. Matsuo, and D. Shen, “Segmenting lung fields in serial chest radiographs using both population-based and patient-specific shape statistics,” *Medical Imaging, IEEE Transactions on*, vol. 27, no. 4, pp. 481–494, 2008.
- [81] I. C. Sluimer, M. Niemeijer, and B. Van Ginneken, “Lung field segmentation from thin-slice ct scans in presence of severe pathology,” in *Medical Imaging 2004*. International Society for Optics and Photonics, 2004, pp. 1447–1455.

- [82] I. Sluimer, M. Prokop, and B. Van Ginneken, "Toward automated segmentation of the pathological lung in ct," *Medical Imaging, IEEE Transactions on*, vol. 24, no. 8, pp. 1025–1038, 2005.
- [83] A. El-Baz, G. Gimelfarb, R. Falk, T. Holland, and T. Shaffer, "A new stochastic framework for accurate lung segmentation," in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2008*. Springer, 2008, pp. 322–330.
- [84] A. El-Baz, A. Farag, A. Ali, G. Gimelfarb, and M. Casanova, "A framework for unsupervised segmentation of multi-modal medical images," in *Computer vision approaches to medical image analysis*. Springer, 2006, pp. 120–131.
- [85] A. Soliman, F. Khalifa, A. Alansary, G. Gimel'farb, and A. El-Baz, "Segmentation of lung region based on using parallel implementation of joint mgrf: Validation on 3d realistic lung phantoms," in *Biomedical Imaging (ISBI), 2013 IEEE 10th International Symposium on*. IEEE, 2013, pp. 864–867.
- [86] Y. Xie, J. Ho, and B. C. Vemuri, "Nonnegative factorization of diffusion tensor images and its applications," in *Information Processing in Medical Imaging*. Springer, 2011, pp. 550–561.
- [87] C. Lazar, A. Doncescu, and N. Kabbaj, "Non negative matrix factorisation clustering capabilities; application on multivariate image segmentation," *International Journal of Business Intelligence and Data Mining*, vol. 5, no. 3, pp. 285–296, 2010.
- [88] R. Sandler and M. Lindenbaum, "Nonnegative matrix factorization with earth mover's distance metric for image analysis," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 33, no. 8, pp. 1590–1602, 2011.
- [89] T. Pang-Ning, M. Steinbach, V. Kumar *et al.*, "Introduction to data mining," *WP Co*, 2006.
- [90] A. Cichocki, R. Zdunek, A. H. Phan, and S. Amari, *Nonnegative matrix and*

*tensor factorizations: applications to exploratory multi-way data analysis and blind source separation.* John Wiley & Sons, 2009.

- [91] S. S. Bucak and B. Günsel, “Incremental subspace learning via non-negative matrix factorization,” *Pattern recognition*, vol. 42, no. 5, pp. 788–797, 2009.
- [92] T. Pang-Ning, M. Steinbach, and V. Kumar, *Introduction to Data Mining, (First Edition)*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 2005. ISBN 0321321367
- [93] H. Kim and H. Park, “Sparse non-negative matrix factorizations via alternating non-negativity-constrained least squares for microarray data analysis,” *Bioinformatics*, vol. 23, no. 12, pp. 1495–1502, 2007.
- [94] V. P. Pauca, J. Piper, and R. J. Plemmons, “Nonnegative matrix factorization for spectral data analysis,” *Linear Algebra and Its Applications*, vol. 416, no. 1, pp. 29–47, 2006.
- [95] D. Cai, X. He, J. Han, and T. S. Huang, “Graph regularized nonnegative matrix factorization for data representation,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 33, no. 8, pp. 1548–1560, 2011.
- [96] N. Ahuja, A. Rosenfeld, and R. M. Haralick, “Neighbor gray levels as features in pixel classification,” *Pattern Recognition*, vol. 12, no. 4, pp. 251–260, 1980.
- [97] L. R. Dice, “Measures of the amount of ecologic association between species,” *Ecology*, vol. 26, no. 3, pp. 297–302, 1945.
- [98] G. Gerig, M. Jomier, and M. Chakos, “Valmet: A new validation tool for assessing and improving 3d object segmentation,” in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2001*. Springer, 2001, pp. 516–523.
- [99] C. Bouman and K. Sauer, “A generalized gaussian image model for edge-preserving map estimation,” *Image Processing, IEEE Transactions on*, vol. 2, no. 3, pp. 296–310, 1993.

- [100] A. Mansoor, U. Bagci, Z. Xu, B. Foster, K. N. Olivier, J. M. Elinoff, A. F. Suffredini, J. K. Udupa, and D. J. Mollura, “A generic approach to pathological lung segmentation,” *IEEE Transactions on Medical Imaging*, vol. 33, pp. 2293–2310, 2014.
- [101] B. Abdollahi, A. Soliman, A. C. Civelek, X. F. Li, G. Gimel’farb, and A. El-Baz, “A novel gaussian scale space-based joint mgrf framework for precise lung segmentation,” in *Image Processing (ICIP), 2012 19th IEEE International Conference on*. IEEE, 2012, pp. 2029–2032.
- [102] C. Bouman and B. Liu, “Multiple resolution segmentation of textured images,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 13, no. 2, pp. 99–113, 1991.
- [103] C. Xu and J. L. Prince, “Snakes, shapes, and gradient vector flow,” *Image Processing, IEEE Transactions on*, vol. 7, no. 3, pp. 359–369, 1998.
- [104] B. Lassen, J. M. Kuhnigk, M. Schmidt, S. Krass, and H. O. Peitgen, “Lung and lung lobe segmentation methods at fraunhofer mevis,” in *Proc. 4th Intern. MICCAI Workshop on Pulmonary Image Analysis, Toronto, Canada*, 2011.
- [105] J. Pu, J. Roos, C. A. Yi, S. Napel, G. D. Rubin, and D. S. Paik, “Adaptive border marching algorithm: automatic lung segmentation on chest CT images,” *Computerized Medical Imaging and Graphics*, vol. 32, no. 6, pp. 452–462, 2008.
- [106] M. S. Brown, M. F. Mcnitt-Gray, N. J. Mankovich, J. G. Goldin, J. Hiller, L. S. Wilson, and D. Aberie, “Method for segmenting chest ct image data using an anatomical model: preliminary results,” *Medical Imaging, IEEE Trans. on*, vol. 16, no. 6, pp. 828–839, 1997.
- [107] O. Weinheimer, T. Achenbach, C. P. Heussel, and C. Düber, “Automatic lung segmentation in MDCT images,” in *Proc. 4th Intern. MICCAI Workshop on Pulmonary Image Analysis, Toronto, Canada*, 2011, pp. 241–255.

- [108] B. Ayinde, E. Hosseini-Asl, and J. M. Zurada, “Visualizing and understanding nonnegativity constrained sparse autoencoder in deep architecture,” in *submitted to ICAISC*, 2016.
- [109] A. Makhzani and B. Frey, “k-sparse autoencoders,” *arXiv preprint arXiv:1312.5663*, 2013.
- [110] B. A. Olshausen *et al.*, “Emergence of simple-cell receptive field properties by learning a sparse code for natural images,” *Nature*, vol. 381, no. 6583, pp. 607–609, 1996.
- [111] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Advances in neural information processing systems*, 2012, pp. 1097–1105.
- [112] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, “Decaf: A deep convolutional activation feature for generic visual recognition,” *arXiv preprint arXiv:1310.1531*, 2013.
- [113] M. D. Zeiler, D. Krishnan, G. W. Taylor, and R. Fergus, “Deconvolutional networks,” in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 2528–2535.
- [114] M. D. Zeiler and R. Fergus, “Visualizing and understanding convolutional networks,” in *Computer Vision–ECCV 2014*. Springer, 2014, pp. 818–833.
- [115] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: A simple way to prevent neural networks from overfitting,” *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.
- [116] I. J. Goodfellow, D. Warde-Farley, M. Mirza, A. Courville, and Y. Bengio, “Maxout networks,” *arXiv preprint arXiv:1302.4389*, 2013.
- [117] F. Bach, R. Jenatton, J. Mairal, G. Obozinski *et al.*, “Structured sparsity through convex optimization,” *Statistical Science*, vol. 27, no. 4, pp. 450–468, 2012.

- [118] J. Chorowski and J. M. Zurada, “Learning understandable neural networks with nonnegative weight constraints,” *Neural Networks and Learning Systems, IEEE Transactions on*, vol. PP, no. 99, 2014.
- [119] S. J. Wright and J. Nocedal, *Numerical optimization*. Springer New York, 1999, vol. 2.
- [120] T. D. Nguyen, T. Tran, D. Phung, and S. Venkatesh, “Learning parts-based representations with nonnegative restricted boltzmann machine,” in *Asian Conference on Machine Learning*, 2013, pp. 133–148.
- [121] J. M. Zurada, *Introduction to Artificial Neural Systems*. St. Paul, MN, USA: West Publishing Co., 1992.
- [122] R. H. Byrd, P. Lu, J. Nocedal, and C. Zhu, “A limited memory algorithm for bound constrained optimization,” *SIAM Journal on Scientific Computing*, vol. 16, no. 5, pp. 1190–1208, 1995.
- [123] F. S. Samaria and A. C. Harter, “Parameterisation of a stochastic model for human face identification,” in *Applications of Computer Vision*. IEEE, 1994, pp. 138–142.
- [124] Y. LeCun, F. J. Huang, and L. Bottou, “Learning methods for generic object recognition with invariance to pose and lighting,” in *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, vol. 2. IEEE, 2004, pp. II–97.
- [125] K. Sparck Jones, “A statistical interpretation of term specificity and its application in retrieval,” *Journal of documentation*, vol. 28, no. 1, pp. 11–21, 1972.
- [126] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, “Improving neural networks by preventing co-adaptation of feature detectors,” *arXiv preprint arXiv:1207.0580*, 2012.

- [127] F. Bastien, P. Lamblin, R. Pascanu, J. Bergstra, I. J. Goodfellow, A. Bergeron, N. Bouchard, and Y. Bengio, “Theano: new features and speed improvements,” Deep Learning and Unsupervised Feature Learning NIPS 2012 Workshop, 2012.
- [128] I. J. Goodfellow, D. Warde-Farley, P. Lamblin, V. Dumoulin, M. Mirza, R. Pascanu, J. Bergstra, F. Bastien, and Y. Bengio, “Pylearn2: a machine learning research library,” *arXiv preprint arXiv:1308.4214*, 2013. [Online]. Available: <http://arxiv.org/abs/1308.4214>
- [129] S. M. Plis, D. Hjelm, R. Salakhutdinov, E. A. Allen, H. J. Bockholt, J. D. Long, H. J. Johnson, J. Paulsen, J. A. Turner, and V. D. Calhoun, “Deep learning for neuroimaging: a validation study,” *Frontiers in Neuroscience*, vol. 8, no. 229, 2014.
- [130] G. McKhann, D. Drachman, M. Folstein, R. Katzman, D. Price, and E. M. Stadlan, “Clinical diagnosis of Alzheimer’s disease: Report of the NINCDSADRDA Work Group\* under the auspices of Department of Health and Human Services Task Force on Alzheimer’s Disease,” *Neurology*, vol. 34, no. 7, p. 939, 1984.
- [131] A. Association *et al.*, “2014 alzheimer’s disease facts and figures,” *Alzheimer’s & Dementia*, vol. 10, no. 2, pp. e47–e92, 2014.
- [132] E. E. Bron, M. Smits, W. M. van der Flier, H. Vrenken, F. Barkhof, P. Scheltens, J. M. Papma, R. M. Steketee, C. M. Orellana, R. Meijboom *et al.*, “Standardized evaluation of algorithms for computer-aided diagnosis of dementia based on structural MRI: The CADDementia challenge,” *NeuroImage*, 2015.
- [133] M. Liu, D. Zhang, E. Adeli-Mosabbab, and D. Shen, “Inherent structure based multi-view learning with multi-template feature representation for alzheimer’s disease diagnosis,” *Biomedical Engineering, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2015.

- [134] S. M. Plis, D. R. Hjelm, R. Salakhutdinov, E. A. Allen, H. J. Bockholt, J. D. Long, H. J. Johnson, J. S. Paulsen, J. A. Turner, and V. D. Calhoun, “Deep learning for neuroimaging: a validation study,” *Frontiers in neuroscience*, vol. 8, 2014.
- [135] H. Suk and D. Shen, “Deep learning-based feature representation for ad/mci classification,” in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2013*. Springer, 2013, pp. 583–590.
- [136] H. Suk, S. W. Lee, D. Shen, A. D. N. Initiative *et al.*, “Hierarchical feature representation and multimodal fusion with deep learning for ad/mci diagnosis,” *NeuroImage*, vol. 101, pp. 569–582, 2014.
- [137] S. Liu, S. Liu, W. Cai, H. Che, S. Pujol, R. Kikinis, F. Dagan, and M. Fulham, “Multimodal neuroimaging feature learning for multiclass diagnosis of alzheimer’s disease,” *Biomedical Engineering, IEEE Transactions on*, vol. 62, no. 4, pp. 1132–1140, 2015.
- [138] F. Li, L. Tran, K. H. Thung, S. Ji, D. Shen, and J. Li, “A robust deep model for improved classification of AD/MCI patients,” *Biomedical and Health Informatics, IEEE Journal of*, vol. 19, no. 5, pp. 1610–1616, 2015.
- [139] C. R. Jack, M. S. Albert, D. S. Knopman, G. M. McKhann, R. A. Sperling, M. C. Carrillo, B. Thies, and C. H. Phelps, “Introduction to the recommendations from the National Institute on Aging-Alzheimer’s Association workgroups on diagnostic guidelines for Alzheimer’s disease,” *Alzheimer’s & Dementia*, vol. 7, no. 3, pp. 257–262, 2011.
- [140] G. M. McKhann, D. S. Knopman, H. Chertkow, B. T. Hyman, C. R. Jack, C. H. Kawas, W. E. Klunk, W. J. Koroshetz, J. J. Manly, R. Mayeux *et al.*, “The diagnosis of dementia due to Alzheimers disease: Recommendations from the National Institute on Aging-Alzheimers Association workgroups on diagnostic guidelines for Alzheimer’s disease,” *Alzheimer’s & Dementia*, vol. 7, no. 3, pp. 263–269, 2011.

- [141] C. R. Jack, D. S. Knopman, W. J. Jagust, R. C. Petersen, M. W. Weiner, P. S. Aisen, L. M. Shaw, P. Vemuri, H. J. Wiste, S. D. Weigand *et al.*, “Tracking pathophysiological processes in Alzheimer’s disease: an updated hypothetical model of dynamic biomarkers,” *The Lancet Neurology*, vol. 12, no. 2, pp. 207–216, 2013.
- [142] S. Klöppel, C. M. Stonnington, C. Chu, B. Draganski, R. I. Scahill, J. D. Rohrer, N. C. Fox, C. R. Jack, J. Ashburner, and R. S. Frackowiak, “Automatic classification of MR scans in Alzheimer’s disease,” *Brain*, vol. 131, no. 3, pp. 681–689, 2008.
- [143] Y. Fan, D. Shen, R. C. Gur, R. E. Gur, and C. Davatzikos, “COMPARE: classification of morphological patterns using adaptive regional elements,” *Medical Imaging, IEEE Transactions on*, vol. 26, no. 1, pp. 93–105, 2007.
- [144] J. P. Lerch, J. Pruessner, A. P. Zijdenbos, D. L. Collins, S. J. Teipel, H. Hampel, and A. C. Evans, “Automated cortical thickness measurements from MRI can accurately separate Alzheimer’s patients from normal elderly controls,” *Neurobiology of aging*, vol. 29, no. 1, pp. 23–30, 2008.
- [145] E. Gerardin, G. Chételat, M. Chupin, R. Cuingnet, B. Desgranges, H. S. Kim, M. Niethammer, B. Dubois, S. Lehéricy, L. Garnero *et al.*, “Multidimensional classification of hippocampal shape features discriminates alzheimer’s disease and mild cognitive impairment from normal aging,” *Neuroimage*, vol. 47, no. 4, pp. 1476–1486, 2009.
- [146] R. Cuingnet, E. Gerardin, J. Tessieras, G. Auzias, S. Lehéricy, M. O. Habert, M. Chupin, H. Benali, O. Colliot, A. D. N. Initiative *et al.*, “Automatic classification of patients with Alzheimer’s disease from structural MRI: a comparison of ten methods using the ADNI database,” *NeuroImage*, vol. 56, no. 2, pp. 766–781, 2011.
- [147] F. Falahati, E. Westman, and A. Simmons, “Multivariate Data Analysis and Machine Learning in Alzheimer’s Disease with a Focus on Structural Magnetic

- Resonance Imaging,” *Journal of Alzheimer’s Disease*, vol. 41, no. 3, pp. 685–708, 2014.
- [148] M. R. Sabuncu and E. Konukoglu, “Clinical Prediction from Structural Brain MRI Scans: A Large-Scale Empirical Study,” *Neuroinformatics*, vol. 13, no. 1, pp. 31–46, 2015.
- [149] M. Long and J. Wang, “Learning transferable features with deep adaptation networks,” *arXiv preprint arXiv:1502.02791*, 2015.
- [150] C. Y. Lee, S. Xie, P. Gallagher, Z. Zhang, and Z. Tu, “Deeply-supervised nets,” *arXiv preprint arXiv:1409.5185*, 2014.
- [151] X. Glorot and Y. Bengio, “Understanding the difficulty of training deep feedforward neural networks,” in *International conference on artificial intelligence and statistics*, 2010, pp. 249–256.
- [152] X. Zhu, H. Suk, and D. Shen, “A novel matrix-similarity based loss function for joint regression and classification in ad diagnosis,” *NeuroImage*, vol. 100, pp. 91–105, 2014.
- [153] C. Zu, B. Jie, M. Liu, S. Chen, D. Shen, D. Zhang, A. D. N. Initiative *et al.*, “Label-aligned multi-task feature learning for multimodal classification of alzheimers disease and mild cognitive impairment,” *Brain imaging and behavior*, pp. 1–12, 2015.
- [154] R. H. Fletcher, S. W. Fletcher, and G. S. Fletcher, *Clinical epidemiology: the essentials*. Lippincott Williams & Wilkins, 2012.
- [155] A. El-Baz, G. Gimel’farb, R. Falk, and M. A. El-Ghar, “A novel level set-based CAD system for automatic detection of lung nodules in low dose chest CT scans,” in *Lung Imaging and Computer Aided Diagnosis*, A. El-Baz and J. S. Suri, Eds. Taylor , Francis, 2011, vol. 1, ch. 10, pp. 221–238. ISBN 978-1-4398-4558-5

- [156] —, “A new CAD system for early diagnosis of detected lung nodules,” in *Image Processing, 2007. IICIP 2007. IEEE International Conference on*, vol. 2. IEEE, 2007, pp. II–461.
- [157] A. El-Baz, M. Nitzken, E. Vanbogaert, G. Gimel’Farb, R. Falk, and M. Abo El-Ghar, “A novel shape-based diagnostic approach for early diagnosis of lung nodules,” in *Biomedical Imaging: From Nano to Macro, 2011 IEEE International Symposium on*. IEEE, 2011, pp. 137–140.
- [158] A. El-Baz, P. Sethu, G. Gimel’farb, F. Khalifa, A. Elnakib, R. Falk, and M. A. El-Ghar, “A new validation approach for the growth rate measurement using elastic phantoms generated by state-of-the-art microfluidics technology,” in *Proceedings of IEEE International Conference on Image Processing, (ICIP’10)*, Hong Kong, September 26–29, 2010, pp. 4381–4383.
- [159] A. El-Baz, P. Sethu, G. Gimel’farb, F. Khalifa, A. Elnakib, R. Falk, and M. A. E.-G. J. Suri, “Validation of a new imaged-based approach for the accurate estimating of the growth rate of detected lung nodules using real CT images and elastic phantoms generated by state-of-the-art microfluidics technology,” in *Handbook of Lung Imaging and Computer Aided Diagnosis*, A. El-Baz and J. S. Suri, Eds. Taylor & Francis, New York, 2011, vol. 1, ch. 18, pp. 405–420. ISBN 978-1-4398-4557-8
- [160] A. El-Baz, M. Nitzken, F. Khalifa, A. Elnakib, G. Gimel’farb, R. Falk, and M. A. El-Ghar, “3D shape analysis for early diagnosis of malignant lung nodules,” in *Proceedings of International Conference on Information Processing in Medical Imaging, (IPMI’11)*, Monastery Irsee, Germany (Bavaria), July 3–8, 2011, pp. 772–783.
- [161] A. El-Baz, M. Nitzken, A. Elnakib, F. Khalifa, G. Gimel’farb, R. Falk, and M. A. El-Ghar, “3D shape analysis for early diagnosis of malignant lung nodules,” in *Proceedings of International Conference Medical Image Computing and*

*Computer-Assisted Intervention, (MICCAI'11)*, Toronto, Canada, September 18–22, 2011, pp. 175–182.

- [162] A. A. Farag, A. El-Baz, G. Gimel'farb, and R. Falk, "Detection and recognition of lung abnormalities using deformable templates," in *Proceedings of the 17<sup>th</sup> International Conference on Pattern Recognition, (ICPR'04)*, vol. 3. IEEE, 2004, pp. 738–741.
- [163] A. El-Bazl, A. A. Farag, R. Falk, and R. La Rocca, "Automatic identification of lung abnormalities in chest spiral CT scans," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, (ICASSP'03)*, vol. 2. IEEE, 2003, pp. II–261.
- [164] A. El-Baz, G. Gimel'farb, R. Falk, and M. A. El-Ghar, "A novel approach for automatic follow-up of detected lung nodules," in *Proceedings of IEEE International Conference on Image Processing, (ICIP'07)*, vol. 5. IEEE, 2007, pp. V–501.
- [165] A. El-Baz, G. L. Gimel'farb, R. Falk, D. Heredis, and M. Abou El-Ghar, "A novel approach for accurate estimation of the growth rate of the detected lung nodules," in *Proceedings of International Workshop on Pulmonary Image Analysis*, 2008, pp. 33–42.
- [166] A. El-Baz, G. Gimel'farb, R. Falk, M. A. El-Ghar, and H. Refaie, "Promising results for early diagnosis of lung cancer," in *Proceedings of IEEE International Symposium on Biomedical Imaging: From Nano to Macro, (ISBI'08)*. IEEE, 2008, pp. 1151–1154.
- [167] A. El-Baz, G. Gimelfarb, R. Falk, M. A. El-Ghar, S. Rainey, D. Heredia, and T. Shaffer, "Toward early diagnosis of lung cancer," in *Proceedings of Medical Image Computing and Computer-Assisted Intervention, (MICCAI'09)*. Springer, 2009, pp. 682–689.

- [168] A. El-Baz, G. Gimel'farb, R. Falk, and M. El-Ghar, "Appearance analysis for diagnosing malignant lung nodules," in *Proceedings of IEEE International Symposium on Biomedical Imaging: From Nano to Macro (ISBI'10)*. IEEE, 2010, pp. 193–196.
- [169] A. El-Baz, G. Gimelfarb, R. Falk, M. A. El-Ghar, and J. Suri, "Appearance analysis for the early assessment of detected lung nodules," in *Lung Imaging and Computer Aided Diagnosis*. chapter, 2011, ch. 17, pp. 395–404.
- [170] A. El-Baz, P. Sethu, G. Gimel'farb, F. Khalifa, A. Elnakib, R. Falk, and M. A. El-Ghar, "Elastic phantoms generated by microfluidics technology: Validation of an imaged-based approach for accurate measurement of the growth rate of lung nodules," *Biotechnology journal*, vol. 6, no. 2, pp. 195–203, 2011.
- [171] A. El-Baz, A. Farag, G. Gimelfarb, R. Falk, and M. A. El-Ghar, "A novel level set-based computer-aided detection system for automatic detection of lung nodules in low dose chest computed tomography scans," *Lung Imaging and Computer Aided Diagnosis*, vol. 10, pp. 221–238, 2011.
- [172] A. El-Baz, M. Nitzken, G. Gimelfarb, E. Van Bogaert, R. Falk, M. A. El-Ghar, and J. Suri, "Three-dimensional shape analysis using spherical harmonics for early assessment of detected lung nodules," in *Lung Imaging and Computer Aided Diagnosis*. chapter, 2011, ch. 19, pp. 421–438.
- [173] A. El-Baz, G. Gimel'farb, M. Abou El-Ghar, and R. Falk, "Appearance-based diagnostic system for early assessment of malignant lung nodules," in *Proceedings of IEEE International Conference on Image Processing, (ICIP'12)*. IEEE, 2012, pp. 533–536.
- [174] A. El-Baz, A. Soliman, P. McClure, G. Gimel'farb, M. A. El-Ghar, and R. Falk, "Early assessment of malignant lung nodules based on the spatial analysis of detected lung nodules," in *Proceedings of IEEE International Symposium on Biomedical Imaging: From Nano to Macro, (ISBI'12)*. IEEE, 2012, pp. 1463–1466.

- [175] A. El-Baz, A. Elnakib, M. Abou El-Ghar, G. Gimel'farb, R. Falk, and A. Farag, "Automatic detection of 2D and 3D lung nodules in chest spiral CT scans," *International Journal of Biomedical Imaging*, vol. 2013, 2013.
- [176] A. El-Baz, G. M. Beache, G. Gimel'farb, K. Suzuki, K. Okada, A. Elnakib, A. Soliman, and B. Abdollahi, "Computer-aided diagnosis systems for lung cancer: Challenges and methodologies," *International journal of biomedical imaging*, vol. 2013, 2013.
- [177] A. A. Farag, A. El-Baz, G. G. Gimelfarb, R. Falk, and S. G. Hushek, "Automatic detection and recognition of lung abnormalities in helical CT images using deformable templates," in *Proceedings of Medical Image Computing and Computer-Assisted Intervention, (MICCAI'04)*. Springer, 2004, pp. 856–864.
- [178] M. Kondapaneni, M. Nitzken, E. Bogaert, G. Gimelfarb, R. Falk, M. A. El-Ghar, and A. El-Baz, "A novel shape-based diagnostic approach for early diagnosis of lung nodules," *CHEST Journal*, vol. 140, no. 4\_MeetingAbstracts, pp. 655A–655A, 2011.
- [179] M. F. Casanova, A. El-Baz, A. Elnakib, J. Giedd, J. M. Rumsey, E. L. Williams, and A. E. Switala, "Corpus callosum shape analysis with application to dyslexia," *Translational Neuroscience*, vol. 1, no. 2, pp. 124–130, 2010.
- [180] A. El-Baz, A. Elnakib, F. Khalifa, M. A. El-Ghar, P. McClure, A. Soliman, and G. Gimel'farb, "Precise segmentation of 3-D magnetic resonance angiography," *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 7, pp. 2019–2029, 2012.
- [181] A. Alansary, A. Soliman, F. Khalifa, A. Elnakib, M. Mostapha, M. Nitzken, M. Casanova, and A. El-Baz, "MAP-based framework for segmentation of mr brain images based on visual appearance and prior shape," *MIDAS J*, vol. 1, p. 1, 2013.

- [182] A. Alansary, A. Soliman, M. Nitzken, F. Khalifa, A. Elnakib, M. Mostapha, M. Casanova, and A. El-Baz, “An integrated geometrical and stochastic approach for accurate infant brain extraction,” in *Image Processing (ICIP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 3542–3546.
- [183] A. S. Chowdhury, A. K. Rudra, M. Sen, A. Elnakib, and A. El-Baz, “Cerebral white matter segmentation from MRI using probabilistic graph cuts and geometric shape priors.” in *ICIP*, 2010, pp. 3649–3652.
- [184] A. El-Baz, A. Farag, G. Gimel’farb, and S. Hushek, “Automatic cerebrovascular segmentation by accurate probabilistic modeling of tof-mra images,” in *Proc. of International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI’05)*, vol. 1. Palm Springs, 2005, pp. 34–42.
- [185] A. El-Baz, A. Farag, and G. Gimelfarb, “Cerebrovascular segmentation by accurate probabilistic modeling of tof-mra images,” in *Image Analysis*, vol. 3540. Springer, 2005, pp. 1128–1137.
- [186] A. El-Baz, A. Farag, G. Gimel’farb, M. Abou El-Ghar, and T. Eldiasty, “Fast unsupervised segmentation of 3D magnetic resonance angiography,” in *Proc. of IEEE International Conference on Image Processing (ICIP’06)*, 2006, pp. 93–96.
- [187] A. El-Baz, A. A. Farag, G. Gimelfarb, M. A. El-Ghar, and T. Eldiasty, “A new adaptive probabilistic model of blood vessels for segmenting mra images,” in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2006*, vol. 4191. Springer, 2006, pp. 799–806.
- [188] A. El-Baz, A. A. Farag, G. L. Gimel’farb, M. A. El-Ghar, and T. Eldiasty, “Probabilistic modeling of blood vessels for segmenting mra images.” in *ICPR (3)*, 2006, pp. 917–920.
- [189] A. El-Baz, G. Gimelfarb, R. Falk, M. A. El-Ghar, V. Kumar, and D. Heredia, “A novel 3D joint Markov-gibbs model for extracting blood vessels from PC-mra

- images,” in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2009*, vol. 5762. Springer, 2009, pp. 943–950.
- [190] M. Mostapha, A. Alansary, A. Soliman, F. Khalifa, M. Nitzken, R. Khodeir, M. F. Casanova, and A. El-Baz, “Atlas-based approach for the segmentation of infant dti mr brain images,” in *Biomedical Imaging (ISBI), 2014 IEEE 11<sup>th</sup> International Symposium on*. IEEE, 2014, pp. 1255–1258.
- [191] M. Mostapha, A. Soliman, F. Khalifa, A. Elnakib, A. Alansary, M. Nitzken, M. F. Casanova, and A. El-Baz, “A statistical framework for the classification of infant dt images,” in *Image Processing (ICIP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 2222–2226.
- [192] M. Nitzken, M. F. Casanova, F. Khalifa, G. Sokhadze, and A. El-Baz, “Shape-based detection of cortex variability for more accurate discrimination between autistic and normal brains,” in *Handbook of Multi-Modality State-of-the-Art Medical Image Segmentation and Registration Methodologies*, A. El-Baz, R. Acharya, A. Laine, and J. Suri, Eds. New York: Springer Verlag, Mar. 2011, vol. 2, ch. 7, pp. 161–185. ISBN 978-1-4419-8203-2
- [193] A. K. Rudra, M. Sen, A. S. Chowdhury, A. Elnakib, and A. El-Baz, “3D graph cut with new edge weights for cerebral white matter segmentation,” *Pattern Recognition Letters*, vol. 32, no. 7, pp. 941–947, 2011.
- [194] M. Sen, A. K. Rudra, A. S. Chowdhury, A. Elnakib, and A. El-Baz, “Cerebral white matter segmentation using probabilistic graph cut algorithm,” in *Multi Modality State-of-the-Art Medical Image Segmentation and Registration Methodologies*. Springer, 2011, pp. 41–67.
- [195] B. Dombroski, M. Nitzken, A. Elnakib, F. Khalifa, A. El-Baz, and M. F. Casanova, “Cortical surface complexity in a population-based normative sample,” *Translational Neuroscience*, vol. 5, no. 1, pp. 17–24, 2014.

- [196] A. El-Baz, M. F. Casanova, G. Gimel'farb, M. Mott, and A. E. Switwala, "A new image analysis approach for automatic classification of autistic brains," in *Proc. IEEE International Symposium on Biomedical Imaging: From Nano to Macro (ISBI'2007)*. IEEE, 2007, pp. 352–355.
- [197] A. El-Baz, M. Casanova, G. Gimel'farb, M. Mott, and A. Switala, "An MRI-based diagnostic framework for early diagnosis of dyslexia," *International Journal of Computer Assisted Radiology and Surgery*, vol. 3, no. 3-4, pp. 181–189, 2008.
- [198] A. El-Baz, M. Casanova, G. Gimel'farb, M. Mott, A. Switala, E. Vanbogaert, and R. McCracken, "A new CAD system for early diagnosis of dyslexic brains," in *Proc. International Conference on Image Processing (ICIP'2008)*. IEEE, 2008, pp. 1820–1823.
- [199] A. El-Baz, M. Casanova, G. Gimel'farb, M. Mott El-Baz, G. Aymanand Gimel'farb, M. Mott, A. Switala, E. Vanbogaert, and R. McCracken, "Dyslexia diagnostics by 3D texture analysis of cerebral white matter gyrifications," in *Proc. International Conference on Pattern Recognition (ICPR'2008)*. IEEE, 2008, pp. 1–4.
- [200] A. El-Baz, A. Farag, A. Elnakib, M. F. Casanova, G. Gimel'farb, A. E. Switala, D. Jordan, and S. Rainey, "Accurate automated detection of autism related corpus callosum abnormalities," *Journal of Medical Systems*, vol. 35, no. 5, pp. 929–939, 2011.
- [201] A. Elnakib, A. El-Baz, M. F. Casanova, and A. E. Switala, "Dyslexia diagnostics by centerline-based shape analysis of the corpus callosum," in *Proc. International Conference on Pattern Recognition (ICPR'2010)*. IEEE, 2010, pp. 261–264.
- [202] A. Elnakib, A. El-Baz, M. F. Casanova, G. Gimel'farb, and A. E. Switala, "Image-based detection of corpus callosum variability for more accurate discrimination between dyslexic and normal brains," in *Proc. IEEE International*

- Symposium on Biomedical Imaging: From Nano to Macro (ISBI'2010)*. IEEE, 2010, pp. 109–112.
- [203] A. Elnakib, M. F. Casanova, G. Gimel'farb, A. E. Switala, and A. El-Baz, "Dyslexia diagnostics by 3-D shape analysis of the corpus callosum," *IEEE Transactions on Information Technology in Biomedicine*, vol. 16, no. 4, pp. 700–708, 2012.
- [204] A. Elnakib, M. F. Casanova, G. Gimel'farb, and A. El-Baz, "Autism diagnostics by 3D shape analysis of the corpus callosum," in *Machine Learning in Computer-aided Diagnosis: Medical Imaging Intelligence and Analysis*, K. Suzuki, Ed. Berlin: IGI Global, 2012, ch. 15, pp. 315–335.
- [205] A. Elnakib, A. El-Baz, M. F. Casanova, G. Gimel'farb, and A. E. Switala, "Image-based detection of corpus callosum variability for more accurate discrimination between autistic and normal brains," in *Proc. IEEE International Conference on Image Processing (ICIP'2010)*. IEEE, 2010, pp. 4337–4340.
- [206] A. Elnakib, M. F. Casanova, G. Gimel'farb, A. E. Switala, and A. El-Baz, "Autism diagnostics by centerline-based shape analysis of the corpus callosum," in *Proc. IEEE International Symposium on Biomedical Imaging: From Nano to Macro (ISBI'2011)*. IEEE, 2011, pp. 1843–1846.
- [207] M. Nitzken, M. Casanova, G. Gimel'farb, A. Elnakib, F. Khalifa, A. Switala, and A. El-Baz, "3D shape analysis of the brain cortex with application to dyslexia," in *Image Processing (ICIP), 2011 18<sup>th</sup> IEEE International Conference on*. Brussels, Belgium: IEEE, Sep. 2011, pp. 2657–2660, (Selected for oral presentation. Oral acceptance rate is 10 percent and the overall acceptance rate is 35 percent).
- [208] B. A. Dombroski, A. E. Switala, A. S. El-Baz, and M. F. Casanova, "Gyral window mapping of typical cortical folding using MRI," *Translational neuroscience*, vol. 2, no. 2, pp. 142–147, 2011.

- [209] A. El-Baz, M. F. Casanova, G. Gimel'farb, M. Mott, and A. E. Switala, "Autism diagnostics by 3D texture analysis of cerebral white matter gyrifications," in *Proc. International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI'2007)*. Springer, 2007, pp. 882–890.
- [210] A. El-Baz, M. Casanova, G. Gimel'farb, M. Mott, and A. Switala, "A new image-based diagnostic framework for early diagnosis of dyslexic brains," in *Proc. of Computer Assisted Radiology and Surgery (CARS'08)*, 2008, pp. 46–47.
- [211] A. Elnakib, M. Nitzken, M. Casanova, H. Park, G. Gimel'farb, and A. El-Baz, "Quantification of age-related brain cortex change using 3D shape analysis," in *Pattern Recognition (ICPR), 2012 21<sup>st</sup> International Conference on*. IEEE, 2012, pp. 41–44.
- [212] R. Fahmi, A. Elbaz, H. Hassan, A. A. Farag, and M. F. Casanova, "Structural MRI-based discrimination between autistic and typically developing brain," in *Proc. Computer Assisted Radiology and Surgery (CARS'2007)*, 2007, pp. 24–26.
- [213] A. A. Farag, R. Fahmi, M. F. Casanova, A. E. Abdel-Hakim, H. A. El-Munim, and A. El-Baz, "Robust neuroimaging-based classification techniques of autistic vs. typically developing brain," in *Deformable Models*. Springer, 2007, ch. 16, pp. 535–566.
- [214] M. Nitzken, M. Casanova, G. Gimel'farb, F. Khalifa, A. Elnakib, A. E. Switala, and A. El-Baz, "3D shape analysis of the brain cortex with application to autism," in *Biomedical Imaging: From Nano to Macro, 2011 IEEE International Symposium on*. Chicago, Illinois, USA: IEEE, Mar. 2011, pp. 1847–1850.
- [215] M. J. Nitzken, M. F. Casanova, and A. El-Baz, "Spharm analysis of the brain cortex for diagnosing dyslexia," in *Biomedical Imaging (ISBI), 2014 IEEE 11<sup>th</sup> International Symposium on*. Beijing, China: IEEE, Apr. 2014.
- [216] A. M. Ali, A. A. Farag, and A. El-Baz, "Graph cuts framework for kidney segmentation with prior shape constraints," in *Proceedings of International*

*Conference on Medical Image Computing and Computer-Assisted Intervention, (MICCAI'07)*, vol. 1, Brisbane, Australia, October 29–November 2, 2007, pp. 384–392.

- [217] F. Khalifa, A. El-Baz, G. Gimel'farb, R. Ouseph, and M. A. El-Ghar, "Shape-appearance guided level-set deformable model for image segmentation," in *Proceedings of IAPR International Conference on Pattern Recognition, (ICPR'10)*, Istanbul, Turkey, August 23–26, 2010, pp. 4581–4584.
- [218] F. Khalifa, G. Gimel'farb, M. A. El-Ghar, G. Sokhadze, S. Manning, P. McClure, R. Ouseph, and A. El-Baz, "A new deformable model-based segmentation approach for accurate extraction of the kidney from abdominal CT images," in *Proceedings of IEEE International Conference on Image Processing, (ICIP'11)*, Brussels, Belgium, September 11–14, 2011, pp. 3393–3396.
- [219] F. Khalifa, A. Elnakib, G. M. Beache, G. Gimel'farb, M. A. El-Ghar, G. Sokhadze, S. Manning, P. McClure, and A. El-Baz, "3D kidney segmentation from CT images using a level set approach guided by a novel stochastic speed function," in *Proceedings of International Conference Medical Image Computing and Computer-Assisted Intervention, (MICCAI'11)*, Toronto, Canada, September 18–22, 2011, pp. 587–594.
- [220] A. Rudra, A. Chowdhury, A. Elnakib, F. Khalifa, A. Soliman, G. M. Beache, and A. El-Baz, "Kidney segmentation using graph cuts and pixel connectivity," *Pattern Recognition Letters*, vol. 34, no. 13, pp. 1470–1475, 2013.
- [221] M. Shehata, F. Khalifa, A. Soliman, R. Alrefai, M. A. El-Ghar, A. C. Dwyer, R. Ouseph, and A. El-Baz, "A novel framework for automatic segmentation of kidney from DW-MRI," in *Proc. IEEE International Symposium on Biomedical Imaging: From Nano to Macro (ISBI'15)*, New York, USA, April 16–19, 2015. IEEE, 2015, p. In press.
- [222] S. E. Yuksel, A. El-Baz, and A. A. Farag, "A kidney segmentation framework for dynamic contrast enhanced magnetic resonance imaging," in *Proceedings of*

*International Symposium on Mathematical Methods in Engineering, (MME'06)*, Ankara, Turkey, April, 27-29, 2006, pp. 55–64.

- [223] S. E. Yuksel, A. El-Baz, A. A. Farag, M. El-Ghar, T. Eldiasty, and M. A. Ghoneim, “A kidney segmentation framework for dynamic contrast enhanced magnetic resonance imaging,” *Journal of Vibration and Control*, vol. 13, no. 9-10, pp. 1505–1516, 2007.
- [224] F. Khalifa, M. A. El-Ghar, B. Abdollahi, H. B. Frieboes, T. El-Diasty, and A. El-Baz, “Dynamic contrast-enhanced MRI-based early detection of acute renal transplant rejection,” in *2014 Annual Scientific Meeting and Educational Course Brochure of the Society of Abdominal Radiology, (SAR'14)*, Boca Raton, Florida, March 23–28, 2014, p. CID: 1855912.
- [225] A. El-Baz, A. A. Farag, S. E. Yuksel, M. E. A. El-Ghar, T. A. Eldiasty, and M. A. Ghoneim, “Application of deformable models for the detection of acute renal rejection,” in *Deformable Models*, A. A. Farag and J. S. Suri, Eds., 2007, vol. 1, ch. 10, pp. 293–333.
- [226] A. El-Baz, G. Gimel'farb, and M. A. El-Ghar, “A novel image analysis approach for accurate identification of acute renal rejection,” in *Proceedings of IEEE International Conference on Image Processing, (ICIP'08)*, San Diego, California, USA, October 12–15, 2008, pp. 1812–1815.
- [227] A. El-Baz, A. Farag, R. Fahmi, S. Yuksel, M. A. El-Ghar, and T. Eldiasty, “Image analysis of renal DCE MRI for the detection of acute renal rejection,” in *Proceedings of IAPR International Conference on Pattern Recognition (ICPR'06)*, Hong Kong, August 20-24, 2006, pp. 822–825.
- [228] A. El-Baz, G. Gimel'farb, and M. A. El-Ghar, “Image analysis approach for identification of renal transplant rejection,” in *Proceedings of IAPR International Conference on Pattern Recognition, (ICPR'08)*, Tampa, Florida, USA, December 8-11, 2008, pp. 1–4.

- [229] A. Farag, A. El-Baz, S. Yuksel, M. A. El-Ghar, and T. Eldiasty, "A framework for the detection of acute rejection with Dynamic Contrast Enhanced Magnetic Resonance Imaging," in *Proceedings of IEEE International Symposium on Biomedical Imaging: From Nano to Macro, (ISBI'06)*, Arlington, Virginia, USA, April 6-9, 2006, pp. 418–421.
- [230] A. El-Baz, A. Farag, R. Fahmi, S. Yuksel, W. Miller, M. A. El-Ghar, T. El-Diasty, and M. Ghoneim, "A new CAD system for the evaluation of kidney diseases using DCE-MRI," in *Proceedings of International Conference on Medical Image Computing and Computer-Assisted Intervention, (MICCAI'08)*, Copenhagen, Denmark, October 1-6, 2006, pp. 446–453.
- [231] F. Khalifa, A. El-Baz, G. Gimel'farb, and M. A. El-Ghar, "Non-invasive image-based approach for early detection of acute renal rejection," in *Proceedings of International Conference Medical Image Computing and Computer-Assisted Intervention, (MICCAI'10)*, Beijing, China, September 20–24, 2010, pp. 10–18.
- [232] F. Khalifa, M. A. El-Ghar, B. Abdollahi, H. Frieboes, T. El-Diasty, and A. El-Baz, "A comprehensive non-invasive framework for automated evaluation of acute renal transplant rejection using DCE-MRI," *NMR in Biomedicine*, vol. 26, no. 11, pp. 1460–1470, 2013.
- [233] F. Khalifa, G. M. Beache, M. A. El-Ghar, T. El-Diasty, G. Gimel'farb, M. Kong, and A. El-Baz, "Dynamic contrast-enhanced MRI-based early detection of acute renal transplant rejection," *IEEE Transactions on Medical Imaging*, vol. 32, no. 10, pp. 1910–1927, 2013.
- [234] M. Mostapha, F. Khalifa, A. Alansary, A. Soliman, J. Suri, and A. El-Baz, "Computer-aided diagnosis systems for acute renal transplant rejection: Challenges and methodologies," in *Abdomen and Thoracic Imaging*, A. El-Baz and L. saba J. Suri, Eds. Springer, 2014, pp. 1–35.
- [235] S. E. Yuksel, A. El-Baz, A. A. Farag, M. E. Abo El-Ghar, T. A. Eldiasty, and

- M. A. Ghoneim, "Automatic detection of renal rejection after kidney transplantation," in *International Congress Series*, vol. 1281, 2005, pp. 773–778.
- [236] L. Mackelaite, R. Ouseph, A. El-Baz, and A. Gaweda, "Cortical CT perfusion of the live donor kidneys as a predictor of post transplant graft function," in *AMERICAN JOURNAL OF TRANSPLANTATION*, vol. 12. WILEY-BLACKWELL COMMERCE PLACE, 350 MAIN ST, MALDEN 02148, MA USA, 2012, pp. 329–329.
- [237] M. Mostapha, F. Khalifa, A. Alansary, A. Soliman, G. Gimel'farb, and A. El-Baz, "Dynamic MRI-based computer aided diagnostic systems for early detection of kidney transplant rejection: A survey," in *2013 INTERNATIONAL SYMPOSIUM ON COMPUTATIONAL MODELS FOR LIFE SCIENCES*, vol. 1559, no. 1. AIP Publishing, 2013, pp. 297–306.
- [238] M. Abou El-Ghar, A. El-Baz, F. Khalifa, A. Elnakib, A. Firjani, and T. El-Diasty, "Non-invasive image-based approach for early diagnosis of prostate cancer," in *Proceedings of 18<sup>th</sup> Symposium of the European Society of Urogenital Radiology, (ESUR'11)*, Dubrovnik, Croatia, October 13–16, 2011.
- [239] A. Firjani, F. Khalifa, A. Elnakib, G. Gimel'farb, M. A. El-Ghar, A. Elmaghraby, and A. El-Baz, "Non-invasive image-based approach for early detection of prostate cancer," in *Proceedings of Fourth International Conference on Developments in eSystems Engineering, (DeSE'11)*, Dubai, UAE, December 6–8, 2011, pp. 172–177.
- [240] A. Firjani, F. Khalifa, A. Elnakib, G. Gimel'farb, A. Elmaghraby, and A. El-Baz, "A novel image-based approach for early detection of prostate cancer," in *Proceedings of IEEE International Conference on Image Processing, (ICIP'12)*, Lake Buena Vista, Florida, September 30–October 3, 2012, pp. 2849–2852.
- [241] A. Firjani, A. Elmaghraby, and A. El-Baz, "MRI-based diagnostic system for early detection of prostate cancer," in *Biomedical Sciences and Engineering Conference (BSEC), 2013*, May 2013, pp. 1–4.

- [242] A. Firjani, A. Elnakib, F. Khalifa, G. Gimelfarb, M. Abou El-Ghar, A. Elmaghraby, and A. El-Baz, “A diffusion-weighted imaging based diagnostic system for early detection of prostate cancer,” *Journal of Biomedical Science and Engineering*, vol. 6, no. 3A, pp. 346–356, 2013.
- [243] A. Firjani, F. Khalifa, A. Elnakib, G. Gimel’farb, M. El-Ghar, A. Elmaghraby, and A. El-Baz, “A novel image-based approach for early detection of prostate cancer using DCe-MRI,” in *Computational Intelligence in Biomedical Imaging*, K. Suzuki, Ed. Springer New York, 2014, pp. 55–82. ISBN 978-1-4614-7244-5
- [244] A. Firjani, A. Elnakib, F. Khalifa, A. El-Baz, G. Gimel’farb, M. A. El-Ghar, and A. Elmaghraby, “A novel 3D segmentation approach for segmenting the prostate from dynamic contrast enhanced MRI using current appearance and learned shape prior,” in *Proceedings of IEEE International Symposium on Signal Processing and Information Technology, (ISSPIT’10)*, Luxor, Egypt, December 15–18, 2010, pp. 137–143.
- [245] A. Firjany, A. Elnakib, A. El-Baz, G. Gimelfarb, M. A. El-Ghar, and A. Elmagharby, “Novel stochastic framework for accurate segemntation of prostate in dynamic contrast-enhanced MRI,” in *Proceedings of the International Wokshop on Prostate Cancer Imaging: Computer-Aided Diagnosis, Prognosis, and Intervention*, Beijing, China, Sptember 24, 2010, pp. 121–130.
- [246] A. Firjani, A. Elnakib, F. Khalifa, G. Gimel’farb, M. A. El-Ghar, A. Elmaghraby, and A. El-Baz, “A new 3D automatic segmentation framework for accurate extraction of prostate from diffusion imaging,” in *Proceedings of Biomedical Science and Engineering Conference–Image Informatics and Analytics in Biomedicine, (BSEC’11)*, Knoxville, Tennessee, March 15–17, 2011, pp. 1306–1309.
- [247] A. Firjani, F. Khalifa, A. Elnakib, G. Gimel’farb, M. A. El-Ghar, A. Elmaghraby, and A. El-Baz, “3D automatic approach for precise segmentation of the prostate from diffusion-weighted magnetic resonance imaging,” in *Proceedings of IEEE*

*International Conference on Image Processing, (ICIP'11)*, Brussels, Belgium, September 11–14, 2011, pp. 2285–2288.

- [248] A. Firjani, A. Elnakib, F. Khalifa, G. Gimel'farb, M. A. El-Ghar, J. Suri, A. Elmaghraby, and A. El-Baz, "A new 3D automatic segmentation framework for accurate extraction of prostate from DCE-MRI," in *Proceedings of IEEE International Symposium on Biomedical Imaging: From Nano to Macro, (ISBI'11)*, Chicago, Illinois, March 30–April 2, 2011, pp. 1476–1479.
- [249] A. Firjany, A. Elnakib, A. El-Baz, G. Gimel'farb, M. El-Ghar, and A. Elmagharby, "Novel stochastic framework for accurate segmentation of prostate in dynamic contrast enhanced MRI," in *Prostate Cancer Imaging. Computer-Aided Diagnosis, Prognosis, and Intervention*, ser. Lecture Notes in Computer Science, A. Madabhushi, J. Dowling, P. Yan, A. Fenster, P. Abolmaesumi, and N. Hata, Eds. Springer Berlin Heidelberg, 2010, vol. 6367, pp. 121–130. ISBN 978-3-642-15988-6
- [250] M. F. Casanova, A. El-Baz, A. Elnakib, A. E. Switala, E. L. Williams, D. L. Williams, N. J. Minshew, and T. E. Conturo, "Quantitative analysis of the shape of the corpus callosum in patients with autism and comparison individuals," *Autism*, vol. 15, no. 2, pp. 223–238, 2011.
- [251] R. Fahmi, A. S. El-Baz, H. Abd El Munim, A. A. Farag, and M. F. Casanova, "Classification techniques for autistic vs. typically developing brain using MRI data," in *Proc. IEEE International Symposium on Biomedical Imaging: From Nano to Macro (ISBI'2007)*. IEEE, 2007, pp. 1348–1351.
- [252] M. F. Casanova, A. El-Baz, M. Mott, G. Mannheim, H. Hassan, R. Fahmi, J. Giedd, J. M. Rumsey, A. E. Switala, and A. Farag, "Reduced gyral window and corpus callosum size in autism: Possible macroscopic correlates of a minicolumnopathy," *Journal of Autism and Developmental Disorders*, vol. 39, no. 5, pp. 751–764, 2009.

- [253] M. F. Casanova, A. El-Baz, E. Vanbogaert, P. Narahari, and A. Switala, “A topographic study of minicolumnar core width by lamina comparison between autistic subjects and controls: Possible minicolumnar disruption due to an anatomical element in-common to multiple laminae,” *Brain Pathology*, vol. 20, no. 2, pp. 451–458, 2010.
- [254] M. F. Casanova, A. S. El-Baz, S. S. Kamat, B. A. Dombroski, F. Khalifa, A. Elnakib, A. Soliman, A. Allison-McNutt, and A. E. Switala, “Focal cortical dysplasias in autism spectrum disorders,” *Acta Neuropathologica Communications*, vol. 1, no. 1, p. 67, 2013.
- [255] E. L. Williams, A. El-Baz, M. Nitzken, A. E. Switala, and M. F. Casanova, “Spherical harmonic analysis of cortical complexity in autism and dyslexia,” *Translational Neuroscience*, vol. 3, no. 1, pp. 36–40, 2012.

## APPENDIX A

### LIST OF ACRONYMS

	Page	
3D-CAES	Stacked 3D CAE	86
3D-CNN	3D Convolutional Neural Network	84
ACC	ACCuracy	96
AD	Alzheimer's Disease	1
ADNI	Alzheimer's Disease Neuroimaging Initiative	84
AE	Autoencoder	1
AI	Artifitial Intelligence	1
ALS	Alternating Least Squares	6
ALS-PGD	ALS method based on Projected Gradient Descent	7
ALVD	Absolute Lung Volume Difference	36
ANLS	Alternating Nonnegative Least Square	7
ANLS-AS	ANLS based on Active Set	7
ANLS-BP	ANLS based on Block Pivoting	7
AUC	Area Under the ROC Curve	40
BAC	BAlanced Accuracy	96
Beta-ME	$\beta$ -divergence ME	8
CAE	Convolutional Autoencoder	1
CGD	Constrained Gradient Descent	8
CLS	Constrained Least Squares	6
CNMF	Constrained NMF	22
CNN	Convolutional Neural Network	1
CSF	Cerebrospinal Fluid	81

DBM	Deep Boltzmann Machine	81
DpAE	Dropout Autoencoder	64
DSA-3D-CNN	Deeply Supervised Adapted 3D-CNN	84
DSC	Dice Similarity Measure	36
DTI	Diffusion Tensor Image	22
fMRI	functional MRI	82
GD-CLS	Gradient Descent Least Squares	6
GM	Gray Matter	82
GVF	Gradient Vector Flow	39
ICA	Independent Component Analysis	2
ICNMF	Incremental Constrained NMF	22
INMF	Incremental NMF	22
KL divergence	Kullback-Leibler divergence	4
LOLA11	Lobe and Lung Analysis 2011	40
MAP	Maximum A Posteriori	22
MCI	Mild Cognitive Impairment	80
ME	Majorization Equalization	8
MHD	Modified Hausdorff distance	36
MRS	Multiple Resolution Segmentation	39
MTL	Multi-Task Learning	81
NC-DAE	Nonnegativity Constrained DAE	68
NCAE	Nonnegative Constrained Autoencoder	46
NMF	Nonnegative Matrix Factorization	1
NNLS	Non-Negative Least Squares	7
NPV	Negative Predictive Value	96
NTD	Nonnegative Tensor Decomposition	24
PCA	Principal Component Analysis	2
PGD	Projected Gradient Descent	24
PPV	Positive Predictive Value	96
RBM	Restricted Boltzmann Machine	9

ReLU	Rectified Linear unit	70
ROC	Receiver Operating Curve	40
ROI	Region Of Interest	81
SAE	Stacked Autoencoder	11
SC	Sparse Coding	2
SEN	SENSitivity	96
sMRI	structural MRI	82
SpC	Sparseness Constraint	5
SPE	SPEcificity	96
SSCAE	Structured Sparse CAE	68
SVHN	Street View House Numbers	72
SVM	Support vector Machine	82
t-SNE	t-distributed Stochastic Neighbor Embedding	54
WM	White Matter	82

## CURRICULUM VITAE

NAME: Ehsan Hosseini-Asl

ADDRESS: Lutz Hall 408  
University of Louisville  
Louisville, KY 40292

### EDUCATION & TRAINING:

B.Sc. Electrical Engineering  
Sahand University of Technology  
2001-2005

M.Sc., Electrical Engineering  
Petroleum University of Technology  
2005-2008

Ph.D., Electrical and Computer Engineering  
University of Louisville  
2012-2016

Internship  
Captricity Inc., Oakland CA  
2015-2016 (8 months)

TEACHING: Digital Signal Processing Labs – GTA

AWARDS: Graduate Student Research Grant

## PUBLICATIONS

### REFEREED JOURNALS

E. Hosseini-Asl, J. M. Zurada, "Structured Sparse Convolutional Autoencoder", *In preparation for Pattern Analysis and Machine Intelligence, IEEE Transactions on.*

E. Hosseini-Asl, J. M. Zurada, and A. El-Baz, "A Deeply Supervised Adapted 3D Convolutional Neural Network for Alzheimer's Disease Classification, *In preparation for Pattern Analysis and Machine Intelligence, IEEE Transactions on.*

E. Hosseini-Asl, J. M. Zurada, O. Nasraoui, "Deep Learning of Part-Based Representation of Data Using Sparse Autoencoders With Nonnegativity Constraints," *in Neural Networks and Learning Systems, IEEE Transactions on*, vol.PP, no.99, pp.1-13, doi: 10.1109/TNNLS.2015.2479223.

E. Hosseini-Asl, J. M. Zurada, G. Gimel farb, A. El-Baz, "3D Lung Segmentation by Incremental Constrained Nonnegative Matrix Factorization," *in Biomedical Engineering, IEEE Transactions on*, vol.PP, no.99, pp.1-1, doi: 10.1109/TBME.2015.2482387.

E. Hosseini-Asl, J. M. Zurada, "Multiplicative Algorithm for Correntropy-Based Nonnegative Matrix Factorization", *Journal of Applied Computer Science Methods*, 5.2, pp.89-104, 2013.

### CONFERENCES

B., Ayinde, E. Hosseini-Asl, J. M. Zurada, Visualizing and Understanding Nonnegativity Constrained Sparse Autoencoder in Deep Architecture, *submitted to International Conference on Artificial Intelligence and Soft Computing*, 2016.

E. Hosseini-Asl, and J. M. Zurada, and A. El-baz, "Automatic Segmentation of Pathological Lung Using Incremental Nonnegative Matrix Factorization", *in Image Processing (ICIP)*,

2015 *IEEE Int. Conference on*, Quebec City, Canada, September 27-30, pp.3111-3115, 2015.

E. Hosseini-Asl, and J. M. Zurada, and A. El-baz, "Lung Segmentation Based on Nonnegative Matrix Factorization", in *Image Processing (ICIP), 2014 IEEE Int. Conference on*, Paris, France, Oct 2014, pp. 877-881.

E. Hosseini-Asl and J. M. Zurada, "Nonnegative Matrix Factorization for Document Clustering: A Survey", in *Artificial Intelligence and Soft Computing, Springer International Publishing*, 2014, vol. 8468, pp. 726-737.

V. D. Luca, E. Hosseini-Asl, S. Graziani, J. M. Zurada, "Neural Modeling of relative humidity on IP2C vibrating transducer", *Procedia Engineering, EUROSENSORS 2014, the 28th European Conference on Solid-State Transducers*, vol. 87, pp. 424-427, 2014.

J. M. Zurada, T. Ensari, E.H. Asl, J. Chorowski, "Nonnegative Matrix Factorization and its application to pattern analysis and text mining," in *Computer Science and Information Systems (FedCSIS)*, 2013 Federated Conference on, pp.11-16, 2013.

E. Hossaini-Asl, and M. Shahbazian. "Nonlinear dynamic system control using wavelet neural network based on sampling theory," *In Systems, Man and Cybernetics, SMC 2009. IEEE International Conference on*, pp. 4502-4507. IEEE, 2009.