SPARSE SIGNAL PROCESSING FOR MACHINE LEARNING AND COMPUTER VISION

by

Yin Zhou

A dissertation submitted to the Faculty of the University of Delaware in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Electrical and Computer Engineering

Fall 2014

ProQuest LLC.
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106 - 1346

SPARSE SIGNAL PROCESSING FOR MACHINE LEARNING AND COMPUTER VISION

by

Yin Zhou

Approved: _____
Kenneth E. Barner, Ph.D.
Chair of the Department of Electrical and Computer Engineering

Approved: _____
Babatunde Ogunnaike, Ph.D.
Dean of the College of Engineering

Approved: _____
James G. Richards, Ph.D.
Vice Provost for Graduate and Professional Education

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed: _____

Kenneth E. Barner, Ph.D.
Professor in charge of dissertation

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed: _____

Charles G. Boncelet, Jr., Ph.D.
Member of dissertation committee

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed: _____

Ryan Zurakowski, Ph.D.
Member of dissertation committee

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed: _____

Jingyi Yu, Ph.D.
Member of dissertation committee

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed: _____

Bahram Parvin, Ph.D.
Member of dissertation committee

ACKNOWLEDGEMENTS

First, I would like to express greatest gratitude to my advisor Prof. Kenneth E. Barner, for his patient guidance and persistent support throughout my Ph.D. study. During the past five years, I have been deeply inspired by his passion to explore new domains, his courage in overcoming difficulties, his wisdom in solving challenging problems, his concentration and rigorousness to details, his dedication and commitment to his work, and his respectful personality. I highly appreciate the trust and freedom he has given to me, allowing me to choose and pursue my own research project. More importantly, I want to thank him for consistently supporting me with his profound knowledge and generous encouragement along my Ph.D. journey. In addition, his enthusiasm about novel technology will always propel me towards exploring unknowns. In all, it is definitely a privilege to work with him and the invaluable experience as his student will benefit me for my whole life.

I also would like to extend my gratitude to my dissertation committee members: Prof. Charles G. Boncelet, Jr., Prof. Ryan Zurakowski, Prof. Jingyi Yu, and Prof. Bahram Parvin. It is a great honor for me to have them serve as the witness of my progress along the academic career path. It was a great pleasure to collaborate with Prof. Yu on several projects, during which I was deeply impressed by his commitment to efficiency and insights to research. I was fortunate to work with Prof. Parvin as summer intern at Lawrence Berkeley National Laboratory. I admire his passion to tackle cross-domain problems and his promptness in realizing new ideas. Moreover, I am also thankful to Prof. Boncelet and Prof. Zurakowski for giving me many insightful comments on my dissertation.

I am very grateful to all the previous and current group members, including Prof. Kai Liu, Dr. Rafael Carrillo, Dr. Jinglun Gao, Dr. Rui Hu, Luisa Polania, Xin

Guo, Sherin Mathews. I want to thank them for their selfless encouragement and help. During my Ph.D. study, I am very lucky to meet many good friends, including Dr. Hao Feng, Dr. Yao Xiao, Dr. Qi Wang, Bo Lu, Lu Li, Bin Zhu, Bohan Zhang, Guangyi Liu, Li Li, Xiaolong Wang, Dr. Jinwei Ye, Yu Ji, Shuhan Chen. I also would like to express gratitude to my colleagues and friends met during my internship, including Dr. Hang Chang, Gerald Fontenay, Kenneth Li and Christopher Phillippi.

Last but not least, I would like to give my deepest thanks to my parents for their unconditional support and love throughout my life. Without their support, this thesis would not have been possible. I want to give special thanks to my fiancée Dr. Wenqiong Tang. Meeting her on our first flight to UD is the luckiest coincidence in my life and without her companionship, my five-year Ph.D. study would not be so beautiful.

TABLE OF CONTENTS

LIST OF TABLES . . . . . **xi**
LIST OF FIGURES . . . . . **xiii**
ABSTRACT . . . . . **xvii**

Chapter

1 INTRODUCTION **1**

1.1 Sparse Signal Representation 1
1.2 Motivation 2
1.3 Related Work 4

1.3.1 Locality-Constrained Sparse Coding 4
1.3.2 Dictionary Learning 5
1.3.3 Unsupervised Feature Learning 7

1.4 Overview of Proposed Approaches 9
1.5 Summary of Contributions 10
1.6 Related Publications to The Described Contributions 11
1.7 Organization 12

2 LOCALITY-CONSTRAINED DICTIONARY LEARNING **13**

2.1 Introduction 13
2.2 LCDL Algorithm 15

2.2.1 Problem Formulation 15
2.2.2 Locality Constrained Dictionary Learning (LCDL) 16
2.2.3 Optimization 19

2.2.3.1 Solving for Local Reconstruction Codes 20
2.2.3.2 Dictionary Optimization 21

2.3 Experimental Results 23

2.4 Conclusion . . . 26

3 DISCRIMINATIVE DICTIONARY LEARNING FOR CLASSIFICATION . . . 27

3.1 3D Shape Recognition . . . 27

3.1.1 Introduction . . . 27
3.1.2 The DL-SLLR Algorithm . . . 29
3.1.3 Optimization . . . 31

3.1.3.1 Supervised Locally Linear Representation . . . 31
3.1.3.2 Updating the Dictionary and the Mapping . . . 32
3.1.3.3 Classification Strategy . . . 34

3.1.4 Experimental Results . . . 35

3.2 Image/Video and Data Classification . . . 38

3.2.1 Introduction . . . 38
3.2.2 The DL-SLC Algorithm . . . 40
3.2.3 Optimization . . . 42

3.2.3.1 Supervised Local Coding . . . 42
3.2.3.2 Locality-Preserving Dictionary Update . . . 44
3.2.3.3 Classification Strategy . . . 47

3.2.4 Experimental Results . . . 48

3.3 2013 IEEE GRSS Data Fusion Contest on Hyperspectral Image Classification . . . 55

3.3.1 Introduction . . . 56
3.3.2 Data Fusion and Classification Algorithm . . . 57
3.3.3 Results . . . 59

3.4 Conclusion . . . 59

4 AUTOMATIC FEATURE LEARNING FOR BIOMEDICAL IMAGE ANALYSIS . . . 61

4.1 Introduction . . . 61
4.2 Related Work . . . 65

4.3 The PSDSPM Algorithm for Tissue Classification 67

4.3.1 Unsupervised Feature Learning 67
4.3.2 Spatial Pyramid Matching (SPM) 69

4.4 The Multispectral CSC Algorithm for Tissue Classification 70

4.4.1 Convolutional Sparse Coding 70
4.4.2 Multispectral Feature Extraction 71

4.5 The SCCR Algorithm for Nuclei Segmentation 73

4.5.1 Training Algorithm 73
4.5.2 Decision Function 76

4.6 Experiments on Tissue Classification 76

4.6.1 The Datasets 77
4.6.2 Evaluating the PSD^{n}SPM Algorithm 78

4.6.2.1 Experimental Configurations 78
4.6.2.2 Discussion 81

4.6.3 Evaluating the MCSCSPM algorithm 82

4.6.3.1 Experimental Configurations 82
4.6.3.2 Discussion 85

4.7 Experiments on Nuclei Segmentation 86
4.8 Conclusion 88

5 KERNEL SPARSE CODING FOR GESTURE RECOGNITION 90

5.1 Introduction 90
5.2 Related Work and Problem Formulation 92

5.2.1 Related Work 92

5.2.2 Problem Formulation 93

5.3 Proposed method 93

5.3.1 Feature Extraction for MTS Data 93

5.3.1.1 SVD Properties of MTS Data 93
5.3.1.2 Simple features for sparse representation 94
5.3.1.3 Robust features for sparse representation 95

5.3.2 Kernelizing Sparse Representation for Classification 98
5.3.3 Algorithm Training Procedure 101
5.3.4 Classification Rule 102

5.4 Experiments on Classifying Real-World MTS Data 103
5.5 Experiments on Classifying Univariate Time Series Data 114
5.6 Conclusion 116

6 SUMMARY 118

6.1 Conclusions 118
6.2 Future Directions and Open Questions 120

BIBLIOGRAPHY 121

Appendix

COPYRIGHT PERMISSIONS 134

LIST OF TABLES

2.1 The overall time (seconds) includes dictionary learning and training data embedding. Note the time measurement may vary based on different implementations. 25

2.2 Comparison of computational complexity for all the methods, including the dictionary learning step and the training data encoding step. 26

3.1 Recognition results on SLI 3D Face Dataset. 36

3.2 Recognition results on SHREC'11 Contest Dataset. 37

3.3 Recognition results over the Extended YaleB Database. Note for D-KSVD and KSVD, recognition rates are cited from [1]. 51

3.4 Comparison of running time (ms) for classifying a test image. . . . 51

3.5 Error rates over the CMU PIE Database for various methods with different sizes training set. 52

3.6 Recognition results over the Weizmann Action Database. 54

3.7 Basic information about Iris, Satellite, Segmentation, Letter and Vehicle datasets from UCI Machine Learning Archive. 55

3.8 Classification accuracy over the UCI Machine Learning data sets. The 3rd column contains the results obtained by keeping only two dimensions of information, *i.e.*, pedal length and pedal width. . . . 55

4.1 Performance of different methods on the GBM dataset. 81

4.2 Performance of different methods on the KIRC dataset. 81

4.3 Performance of different methods on the GBM dataset. 84

4.4 Performance of different methods on the KIRC dataset. 84

4.5 Comparison of Segmentation Results. 88

5.1 Comparison among different kernel functions over the Georgia-Tech HG database. 107

5.2 Binary Classification comparison among various methods over the Auslan database. Recognition rates with $*$ are cited from [2]. . . . 108

5.3 Binary classification result over the Auslan database for various selection of attributes. 108

5.4 Multi-class Classification comparison among various methods over the Auslan database. Recognition rates with $*$ are cited from [3]. Proposed 1 is based on 10-fold cross-validation; For proposed 2, the data pool is divided into 2 folds, *i.e.*, one fold for training and the other fold for test, according to [3]. 110

5.5 Recognition rate on the HAuslan database. The dimension of random subspace is fixed at 40 for all the classification tasks. 111

5.6 Recognition performance on the HAuslan database. 112

5.7 Comparison among different kernel functions over the HAuslan database. 112

5.8 Comparison of recognition rate among various methods over the HAuslan database. Note that recognition rates with $*$ are cited from references. 112

5.9 Classification results on UCR Time-Series Repository. Note that DTW* [4] means 1NN-Best Warping Window DTW and TSBF* [5] represents Time Series based on a Bag-of-Features representation with the optimal parameter setting $z = 0.25$. Results for compared methods are cited from references. 117

LIST OF FIGURES

2.1 Overview of the proposed method. Given training data in high-dimensional observation space, a representational and locality-preserving dictionary is learned. Then, the low-dimensional embedding of the atoms is computed via some NLDR algorithm. Finally, using the geometric relationships among training data and the atoms in observation space, the low-dimensional embedding of training data is reconstructed as linear combinations of the low-dimensional embedding of the atoms. 14

2.2 Illustration of the learning objective. 15

2.3 Illustration of LCDL algorithm. 18

2.4 Low-dimensional embedding reconstruction comparison on Swiss roll (1st row), Punctured sphere (2nd row) and Gaussian (3rd row). Ground truth means the low-dimensional embedding obtained directly from all training samples. The nearest neighbor parameter k of NLDR algorithms is set to 6. The RMSE values are (c) 0.0299, (d) 0.7409, (e) 0.0666, (f) 0.0535, (i) 0.0705, (j) 0.8664, (k) 0.1060, (l) 0.1743, (o) 0.0104, (p) 0.2943, (q) 0.0419, (r) 0.1012. 22

2.5 Classification results over two face databases. The parameter k of LLE is set to 60 for both Extended YaleB and CMU PIE. 24

3.1 The proposed classification strategy. Given a query shape S, extract shape descriptors on it and then perform classification per descriptor. Finally the label of S is determined by majority voting over descriptor decisions. 28

3.2 Majority voting results after normalization on SHREC'11 Contest Dataset. The two objects are bird (a) and hand (b). The bird is associated to label 4 while the hand is associated to label 15. Since the number of extracted descriptors varies across different objects, we normalize the voting results for better visualization. 35

3.3 30 classes from SHREC'11 Contest Dataset. Image cited from SHREC'11 Contest website. 37

3.4 3D Nonrigid shapes from object class horse. 37

3.5 Comparison of performance for all methods on the robustness against partial occlusion. 38

3.6 (a) Classification performance with respect to s. (b) Objective function value versus iterations; (c) Classification error rate versus iterations. 49

3.7 Recognition results over the AR Face Database. 53

3.8 Example MHIs of 10 natural actions. 54

3.9 Illustration of the hyperspectral and LiDAR imaging over University of Houston. Image courtesy to IEEE GRSS Committee. 56

3.10 Contest legend. Image courtesy to IEEE GRSS Committee. 57

3.11 The proposed data fusion pipeline. Image courtesy to IEEE GRSS Committee. 57

3.12 Classification result. The label of each pixel is represented with different color. Image courtesy to IEEE GRSS Committee. 59

4.1 Computed basis functions from the Glioblastoma Multiforme (GBM) dataset. 62

4.2 27×27 multispectral filters learned from the GBM dataset. It can be seen that, learned from the nuclear channel, the filters (top figure) capture nuclear regions of distinct shapes; learned from the collagen channel, the filters (bottom figure) characterize the structural connectivity within various tissue sections. 63

4.3 Computational workflow of our approach (PSD^{n}SPM). 67

4.4 The proposed multispectral feature extraction framework. CoD means color decomposition; Abs means absolute value rectification; LCN means local contrast normalization; MP means max-pooling. The figure is best viewed in color at 150% zoom-in. 72

4.5 21×21 filters learned from the TCGA segmentation benchmark dataset. 75

4.6 GBM Examples. First column: Tumor; Second column: Transition to necrosis; Third column: Necrosis. 77

4.7 KIRC Examples. First column: Tumor; Second column: Normal; Third column: Stromal. 78

4.8 Comparison of PSD with linear and nonlinear regressors in terms of reconstruction. (a) Original image; (b) Reconstruction by PSD with linear regressor (SNR=14.9429); (c) Reconstruction by PSD with nonlinear regressor (SNR=19.3436). 85

4.9 GBM Examples. First row: original images. Second row: predictions by SCCR. Third row: final segmentation results. 87

5.1 Training samples and dictionary atoms of SRC. 99

5.2 Recognition rates for the Georgia-Tech HG database. (a) 15-class problem recognition rate versus selected features (markers) under various random projections. The horizontal axis represents the number of randomly chosen features, ranging from 2 to 22. The curves in different colors represent recognition rates over 5 different random subspaces. (b) 15-class problem recognition rate versus different dimensions of the random subspace; 22 features (markers) are employed. 105

5.3 Recognition rate for various methods over the Georgia-Tech HG database. 106

5.4 Recognition rate on the Georgia-Tech HG database. (a) PCA feature (b) LDA feature (c) CovSVDK feature (proposed method). All three feature extraction methods are fed to four classifiers, i.e., SVM, KNN, LS, the proposed Kernelized SRC. 106

5.5 3D trajectories for 8 signs. (a) Eat, (b) Exit, (c) Forget, (d) Give (e) Hello, (f) Know, (g) Love (h) No. 107

5.6 Illustrations of manifolds in multi-class classification tasks. Top row: the 3-label task; bottom row: the 4-label task. (a) 2D manifold with the kernel trick, (b) 2D manifold without the kernel trick, (c) 3D manifold with the kernel trick, (d) 3D manifold without the kernel trick. 109

5.7 Recognition rate for the HAuslan Database. 111

5.8 Recognition rate over the HAuslan database. (a) PCA feature (b) LDA feature (c) CovSVDK feature (proposed method). All three feature extraction methods are fed to four classifiers, i.e., SVM, KNN, LS, the proposed Kernelized SRC. 112

5.9 ROC curves for outlier detection over the Georgia-Tech HG and the HAuslan databases. (a) the Georgia-Tech HG database, (b) the HAuslan database. CovSVD means feature extraction following Definition. 1 and Definition. 2. 113

5.10 Accuracy scatter plot between Kernelized SRC and 1NN-Best Warping Window DTW [6]. Each dot represents a dataset. Dots above the diagonal mean that Kernelized SRC is better than 1NN-Best Warping Window DTW and vice versa. The farther away a dot is from the diagonal, the greater the accuracy improvement achieved [7]. 115

5.11 Comparison between Kernelized SRC and SRC. (a) accuracy scatter plot; (b) expected accuracy gain versus actual accuracy gain. Note that regions marked as TP/TN represent we correctly predict Kernelized SRC is better/worse than SRC; region FN means that we predict Kernelized SRC is worse than SRC but the fact is the opposite; region FP means that we predict Kernelized SRC is better than SRC but the fact is the opposite. Practically, only FP is the truly bad case [8]. 116

ABSTRACT

Signal sparse representation solves inverse problems to find succinct expressions of data samples as a linear combination of a few atoms in the dictionary or codebook. This model has proven effective in image restoration, denoising, inpainting, compression, pattern classification and automatic unsupervised feature learning.

Many classical sparse coding algorithms have exorbitant computational complexity in solving the sparse solution, which hinders their applicability in real-world large-scale machine learning and computer vision problems. In this dissertation, we will first present a family of locality-constrained dictionary learning algorithms, which can be seen as a special case of sparse coding. Compared to classical sparse coding, locality-constrained coding has closed-form solution and is much more computationally efficient. In addition, the locality-preserving property enables the newly proposed algorithms to better exploit the geometric structures of data manifold. Experimental results demonstrate that our algorithms are capable of achieving superior classification performance with substantially higher efficiency, compared to sparse-coding based dictionary algorithms.

Sparse coding is an effective building block of learning visual features. A good feature representation is critical for machine learning algorithms to achieve satisfactory results. In recent years, unsupervised feature learning has received increasing research interest in various computer vision and pattern recognition problems. Unlike human-engineered feature extractors that typically require domain knowledge and a large amount of labeled data, unsupervised learning algorithms are generic and designed to automatically discover the intrinsic patterns from the abundant unlabeled data that are usually readily available (from Internet) and require no laborious human labeling. In this dissertation, we will explore the capability of feature learning algorithms in

automated biomedical image analysis. Specifically, we will present two unsupervised feature learning models for histopathology image classification. We will also introduce a novel convolutional regression model for nuclei segmentation. Experiments on biomedical image classification and segmentation benchmarks demonstrate that the proposed feature learning systems can achieve very competitive results compared to dedicated systems incorporating biological prior knowledge.

Finally, we propose a sparse coding based framework for classifying complicated human gestures represented as multi-variate time series (MTS). Specifically, we will present a novel feature extraction strategy, which can overcome the problem of inconsistent lengths among MTS data and is robust to the large variability within human gestures. Moreover, we will introduce a generic approach to kernelize sparse representation, which leads to enhanced classification performance. Extensive experiments verify the effectiveness of the proposed framework.

Chapter 1

INTRODUCTION

1.1 Sparse Signal Representation

A sparse signal is a signal that can be succinctly expressed as a linear combination of a few signal templates (called atoms or bases) from an over-complete dictionary or codebook. Sparse representation of the signal aims to solve the following linear system requiring that there are only a few nonzeros in the coefficient vector,

$$\arg\min_{\mathbf{x}} \|\mathbf{x}\|_0 \text{ s.t. } \Phi\mathbf{x} = \mathbf{y} \tag{1.1}$$

where $\Phi \in \mathbb{R}^{m\times K}$ $(m \ll K)$ is an over-complete dictionary whose columns are bases with unit ℓ_2 norm; $\mathbf{x}$ is the sparse representation coefficient vector of signal $\mathbf{y}$ over dictionary Φ; the $\|\cdot\|_0$ is a pseudo-norm defined as the the number of nonzero entries in a vector. In practice, when the signal $\mathbf{y}$ is contaminated with noise, Eq. (1.1) is alternatively formulated by allowing some reconstruction error $\epsilon > 0$, as

$$\arg\min_{\mathbf{x}} \|\mathbf{x}\|_0 \text{ s.t. } \|\Phi\mathbf{x} - \mathbf{y}\|_2 < \epsilon \tag{1.2}$$

However, finding the sparest solution to the above ℓ_0 problem is combinatorially NP-hard [9, 10].

In recent years, the development in compressed sensing and sparse representation [9–11] revealed that if the exact solution $\mathbf{x}$ is sufficiently sparse, the solution to the ℓ_0-minimization problem can be equivalently obtained by solving the ℓ_1-minimization problem as

$$\arg\min_{\mathbf{x}} \|\mathbf{x}\|_1 \text{ s.t. } \|\Phi\alpha - \mathbf{y}\|_2 < \epsilon \tag{1.3}$$

In literature, there are many efficient solvers to Eq. (1.3) or its equivalent formulations, such as Basis Pursuit (BP) [12], Matching Pursuit (MP) [13], Orthogonal Matching Pursuit [14], Homotopy [15], Coordinate Descent algorihtm [16], Fast Iterative Shrinkage-Thresholding algorithm (FISTA) [17], Feature Signa algorithm [18].

1.2 Motivation

A pattern recognition system generally consists of two critical components, *i.e.,* the feature extractor and the classifier. Feature extraction is an operation that transforms the original data samples into some proper representations by exploiting the underlying data statistics, such that the characterization of samples from different classes is more discriminative for the next-step classification. A classifier is a function that maps input data to a class label. Many sophisticated classifiers require a large amount of training data to accurately establish the relationship between data input and class labels.

Nonlinear dimensionality reduction (NLDR) is an important feature extraction technique that discovers the most succinct and intrinsic forms of representation of the original high-dimensional data, allowing more effective learning and prediction. Unfortunately, many existing NLDR algorithms are of quadratic or even cubic complexity in the number of data, which diminishes the applicability of these algorithms to real-world large-scale tasks [19]. It is therefore very much needed to find a method which can compress the tremendous dataset into a small number of meaningful landmark points and preserve the geometric structure of the original data manifold. With such a technique, existing NLDR algorithms will be able to process large-scale datasets with substantially reduced computational cost.

Image-based classification of tissue histology plays an important role in predicting clinical outcomes. However this task is very challenging due to the presence of large technical variations (*e.g.,* fixation, staining) and biological heterogeneities (*e.g.,* cell type, cell state). Currently, many state-of-the-art recognition systems in computer

vision rely on human-engineered features. These techniques typically require domain-specific knowledge and the laborious human-engineering process, which greatly hinders their applicabilities to classifying massive amount of tumor types. Moreover, these manually crafted features can only characterize low-level image statistics [20], which does not satisfy the need of phenotypic concept learning. Finally, to achieve good performance, a system usually needs a large amount of training data, which in practice, are expensive to obtain, as labeling one image takes multiple rounds of discussion and analysis by several doctors. Due to the reasons above, existing human-engineered feature extractors cannot yield satisfactory performance in tissue image classification. It is therefore desirable to develop algorithms that can make use of abundant unlabeled biomedical images and automatically learn intrinsic high-level features.

Sparse coding was originally developed to explain the visual processing mechanism of brain [21] and has recently been proven to be an effective model for learning visual features [22–27] in the unsupervised manner. The primary strength of this technique lies in succinct representation, which essentially allows to abstract and capture the dominant information within the data. However, given that solving sparse coefficients requires time-consuming optimization, directly applying this model usually results in exorbitant computational cost when processing large-scale image classification problems, which therefore greatly hinders its practical usability. To this end, we are motivated to seek efficient methods to generate succinct and informative data representations.

Another problem considered in this dissertation is robust recognition of gestures captured as multivariate time series (MTS). Classifying MTS data is a challenging task in many areas, e.g., pattern recognition [28] and computer vision [29], due to the presence of inconsistent lengths among MTS data and the large inter-class variations. For conventional feature extraction methods, e.g., PCA and LDA, downsampling and interpolation are usually applied on each MTS in order to normalize the data length. However, downsampling may cause a loss of salient information [28], while interpolation may induce distortion to the original data [30]. Therefore, it is desirable to develop

an effective feature extractor for MTS data. Partially due to its robustness to noise and missing data, sparse coding has been successfully applied to many image and audio classification problems. However, little efforts have been made to MTS data classification using this model. It is thus also desirable to exploit the capability of sparse coding for robust MTS data classification.

1.3 Related Work

1.3.1 Locality-Constrained Sparse Coding

Recent studies [31, 32] indicate that by imposing locality constraint, we can exploit local geometry on the nonlinear data manifold and achieve enhanced performance compared to traditional sparse coding.

Specifically, Zhang *et al.* proposed Local Coordinate Coding (LCC) [31] based on ℓ_1-minimization by penalizing nonlocal dictionary atoms from being coded with large linear combination coefficients, as following

$$\min_{\gamma,\mathbf{C}} \sum_{\mathbf{x}} \|\mathbf{x} - \gamma(x)\|^2 + \mu \sum_{\mathbf{v}\in\mathbf{C}} |\gamma_{\mathbf{v}}(\mathbf{x})| \|\mathbf{v} - \mathbf{x}\|^2 + \lambda \|\mathbf{v}\|^2 \tag{1.4}$$

where $\gamma(x) = \sum_{\mathbf{v}\in\mathbf{C}} \gamma_{\mathbf{v}}(\mathbf{x})\mathbf{v}$; $\mathbf{x}$ is a data point in $\mathbf{R}^m$; $(\gamma, \mathbf{C})$ is a coordinate coding where $\mathbf{C}$ is a set of anchor points $\mathbf{v}$ in $\mathbf{R}^m$ and γ is a map of $\mathbf{x} \in \mathbf{R}^m$ to its codes $[\gamma_{\mathbf{v}}(\mathbf{x})]_{\mathbf{v}\in\mathbf{C}} \in \mathbf{R}^{|\mathbf{C}|}$ such that $\sum_{\mathbf{v}} \gamma_{\mathbf{v}}(\mathbf{x}) = 1$. Note that $\gamma_{\mathbf{v}}(\mathbf{x}) \in \mathbf{R}$ is the coefficient of an anchor point $\mathbf{v}$ for reconstructing $\mathbf{x}$ and that the requirement $\sum_{\mathbf{v}} \gamma_{\mathbf{v}}(\mathbf{x}) = 1$ allows the coding to be shift-invariant.

The work of LCC indicates that if a coordinate coding is sufficient localized a nonlinear function can be approximated by a linear function with respect to the coding [31] and that local coding is more effective than traditional sparse coding in terms of capturing the nonlinearity of a function. Nevertheless, the optimization procedure of LCC is computationally expensive.

Wang *et al.* further proposed Locality-constrained Linear Coding (LLC) [32] by introducing a weighting strategy to the coefficient, called locality adaptor. The LLC

is formulated as

$$\begin{aligned} \min_{\mathbf{C},\mathbf{B}} \quad & \sum_{i=1}^{N} \|\mathbf{x}_i - \mathbf{B}\mathbf{c}_i\|^2 + \lambda \|\mathbf{d}_i \odot \mathbf{c}_i\|^2 \\ \text{s.t.} \quad & \mathbf{1}^T \mathbf{c}_i = 1, \forall i \\ & \|\mathbf{b}_j\|^2 \leq 1, \forall j \end{aligned} \tag{1.5}$$

where $\mathbf{x}_i$ is a data point in $\mathbf{R}^m$; $\mathbf{B} \in \mathbf{R}^{m \times K}$ is the codebook; the sum-to-one constraint enables the coding to be shift-invariant; $\odot$ represents the element-wise multiplication and $\mathbf{d}_i \in mathbfR^m$ is the locality adaptor that allows different freedom for each codeword $\mathbf{b}_j$ proportional to its similarity to the input signal $\mathbf{x}_i$. Specifically,

$$\mathbf{d}_j = \exp(\frac{\text{dist}(\mathbf{x}_i, \mathbf{B})}{\sigma}) \tag{1.6}$$

where $\text{dist}(\mathbf{x}_i, \mathbf{B}) = [\text{dist}(\mathbf{x}_i, \mathbf{b}_1), \ldots, \text{dist}(\mathbf{x}_i, \mathbf{b}_K)]^T$, and $\text{dist}(\mathbf{x}_i, \mathbf{b}_j)$ is the Euclidean distance between $\mathbf{x}_i$ and $\mathbf{b}_j$ and σ is used for adjusting the weight decay speed for the locality adaptor [32]. Since the coding is essentially the least-square problem yielding a few significant coefficients, the authors used thresholding to generate the final sparse code.

The advantages of LLC is that it has closed-form solution and possesses local smooth sparsity. However, choosing appropriate parameters for the locality adaptor requires tremendous effort. Moreover, the energy constraint on codewords may cause the learned codebook to diverge from the the data manifold, which precludes its usability in capturing the nonlinearity and local geometry of the data manifold.

1.3.2 Dictionary Learning

While signal sparse representation seeks succinct linear combinations of atoms from a given dictionary, dictionary learning aims to adapt the dictionary to better fit the task-specific model [33]. In other words, given a large set of training signals, dictionary learning seeks a compact set of bases to best represent each signal in the

training set under some sparsity constraints. Specifically, given training set $\mathbf{Y} = \{\mathbf{y}_i\}_{i=1}^N$ containing N signals in $\in \mathbf{R}^m$, the best dictionary for sparse representation of $\mathbf{Y}$ is obtained by solving

$$\min_{\mathbf{D},\mathbf{X}} \|\mathbf{Y} - \mathbf{D}\mathbf{X}\|_F^2 \text{ s.t. } \forall i, \|\mathbf{x}_i\|_0 \leq T_0 \tag{1.7}$$

where $\mathbf{X}$ is the sparse code matrix for representing $\mathbf{Y}$ over the dictionary $\mathbf{D}$ and $\|\mathbf{x}_i\|_0 \leq T_0$ is the strict sparsity constraint allowing no more than T_0 nonzeros in $\mathbf{x}_i \in \mathbf{X}$.

The problem can also be formulated using ℓ_1 penalty as

$$\min_{\mathbf{D},\mathbf{x}\in\mathbf{X}} \frac{1}{N} \sum_{i=1}^{N} \left(\frac{1}{2} \|\mathbf{y}_i - \mathbf{D}\mathbf{x}_i\|_2^2 + \lambda \|\mathbf{x}_i\|_1 \right) \tag{1.8}$$

The problem (Eq. (1.7) and Eq. (1.8) is not jointly convex with respect to (w.r.t) $\mathbf{D}$ and $\mathbf{X}$ but is convex w.r.t one of them while keep the other fixed. A common approach to minimize the above objective is alternating between the two variables, minimizing w.r.t one while keep the other fixed. That is, iteratively solving for sparse representations based on the dictionary and updating the dictionary given the sparse codes, until the stopping criterion is met.

Method of Optimal Directions (MOD) [34] is an efficient dictionary learning algorithm. This method uses either Orthogonal Matching Pursuit (OMP) or FOCUSS to solve for sparse codes first and then performs one-step dictionary update by computing the derivative of the error function.

K-SVD [33] is one of state-of-the-art dictionary learning algorithms, which has achieved impressive results in many computer vision problems, *e.g.,* image inpainting, restoration, denoising and classification. The optimization is an iterative process alternating between solving sparse representations using Orthogonal Matching Pursuit (OMP) and dictionary update using singular value decomposition (SVD). The K-SVD algorithm generates dictionary atoms with unit energy and is guaranteed to converge

to local minimum.

Mairal *et al.* [35] recently proposed online dictionary learning by setting the minimization target as the expected approximation error rather than aiming at a perfect minimization of empirical cost. This algorithm uses stochastic approximations by processing one training signal at a time and minimizes a sequentially quadratic local approximations of the expected approximation error.

To scale to large image classification datasets, many dictionary learning (DL) algorithms have been developed to learn a compact dictionary while trading-off some discriminative terms, such as the Fisher discrimination term [36], the classifier prediction error [37], the incoherence promoting term [38], etc. By including label information and using KSVD [33], Zhang *et al.* [1] proposed Discriminative-KSVD (D-KSVD) for face recognition and Jiang *et al.* [39] further added a label consistent constraint into the objective function to enforce the correspondence between labels and atoms.

1.3.3 Unsupervised Feature Learning

Unsupervised feature learning is a large family of methods that are capable of learning meaningful features from abundant unlabeled data via a sequence of nonlinear processing and can be combined to build feature hierarchies. Representative algorithms are Auto-encoders [40], Restricted Boltzmann Machine (RBM) [41], Gaussian Mixture Model (GMM) [42], Sparse Coding [21], etc. In this dissertation, we mainly focus on sparse coding as the building block for unsupervised feature learning.

Sparse coding was originally developed to explain the visual processing mechanism of brain [21] and has recently been proven to be an effective model for learning visual features [22–27] in the unsupervised manner.

One drawback of sparse coding is that it typically require a time-consuming optimization process. Directly using this model cannot satisfy the need for high-speed signal/image recognition. Lecun *et al.* [22] proposed a highly efficient model, called Predictive Sparse Decomposition (PSD), by incorporating a feed-forward encoder into

the training objective, such that for any new sample, the encoder can predict an approximation to the optimal sparse code. Specifically, PSD is formulated as

$$\begin{aligned} \min_{\mathbf{B},\mathbf{Z},\mathbf{G},\mathbf{W}} \quad & \|\mathbf{Y}-\mathbf{B}\mathbf{X}\|_F^2+\lambda\|\mathbf{X}\|_1+\|\mathbf{X}-\mathbf{G}\sigma(\mathbf{W}\mathbf{Y})\|_F^2 \\ \text{s.t.} \quad & \|\mathbf{b}_i\|_2^2=1, \forall i=1,\ldots,h \end{aligned} \tag{1.9}$$

where $\mathbf{B} \in \mathbb{R}^{m\times h}$ is a set of the basis functions; $\mathbf{X}=[\mathbf{x}_1,...,\mathbf{x}_N] \in \mathbb{R}^{h\times N}$ is the sparse feature matrix; $\mathbf{W} \in \mathbb{R}^{h\times m}$ is the auto-encoder; $\mathbf{G}=\text{diag}(g_1,\ldots,g_h) \in \mathbb{R}^{h\times h}$ is a scaling matrix with diag being an operator aligning vector $[g_1,\ldots,g_h]$ along the diagonal, $\sigma(\cdot)$ is the element-wise sigmoid function and λ is a regularization constant. The goal of jointly minimizing Eq. (1.9) with respect to the quadruple $\langle \mathbf{B},\mathbf{Z},\mathbf{G},\mathbf{W}\rangle$ is to enforce the inference of the nonlinear regressor $\mathbf{G}\sigma(\mathbf{W}\mathbf{X})$ to be resemble to the optimal sparse codes $\mathbf{Z}$ that can reconstruct $\mathbf{X}$ over $\mathbf{B}$ [22]. By stacking this model into hierarchies, Lecun *et al.* [24] achieved state-of-the-art result on handwritten digit recognition.

Another drawback of traditional sparse coding is that the model is not shift-invariant. The model basically learns edge primitives and therefore results in highly redundant dictionary [25, 43]. In recent years, convolutional sparse coding has received increasing research interest in computer vision and machine learning communities [25–27, 43–45], mainly due to its capability of learning shift-invariant filters with complex patterns. The key concept of convolutional sparse coding is replacing dot product between the dictionary and code matrix with convolution operator. The dictionary thus becomes a 2D convolutional filter bank and the code matrix becomes 2D sparse feature maps. Specifically, convolutional sparse coding solves the following objective,

$$\begin{aligned} \min_{\mathbf{D},\mathbf{z}_k\in\mathbf{Z}} \quad & \mathcal{L}=\left\|\mathbf{x}-\sum_{k=1}^{K}\mathbf{d}_k*\mathbf{z}_k\right\|_{\text{F}}^2+\alpha\sum_{k=1}^{K}\|\mathbf{z}_k\|_1 \\ \text{s.t.} \quad & \|\mathbf{d}_k\|_2^2=1, \forall k=1,\ldots,K \end{aligned} \tag{1.10}$$

where the first and the second term represent the reconstruction error and the ℓ_1-norm

penalty respectively; $\mathbf{x}$ is a 2D training image; $\mathbf{D} = \{\mathbf{d}_k\}_{k=1}^{K}$ is the 2D convolutional filter bank having K filters, where each $\mathbf{d}_k$ is a 2D convolutional kernel; $\mathbf{Z} = \{\mathbf{Z}^i\}_{k=1}^{K}$ is the set of sparse feature maps for reconstructing $\mathbf{x}$; α is a regularization parameter; $*$ is the 2D discrete convolution operator; and filters are restricted to have unit energy to avoid trivial solutions. Convolutional sparse coding has achieved state-of-the-art performances in object recognition [27], pedestrian detection [44], retinal blood vessels segmentation [46], and image denoising [45], etc.

1.4 Overview of Proposed Approaches

The primary goal of this dissertation is to develop generic algorithms for both stages (*i.e.,* feature extraction and classification) in computer vision and pattern recognition models. To achieve this objective, we extend and improve existing sparse coding models.

This dissertation summarizes three projects. The first project develops a novel framework for nonlinear dimensionality reduction, for the purpose of algorithmically reducing computational and memory complexity when solving for low-dimensional embedding. We establish a theorem that the approximation to an unobservable intrinsic manifold by a few latent points residing on the manifold can be cast in a novel dictionary learning problem over the observation space. As a result, the proposed method achieves improved embedding quality and substantial efficiency gain. In addition, we explore the effectiveness of locality-preserving property and derive a family of discriminative dictionary learning algorithms for classification tasks and show that they can achieve very competitive performance compared to traditional sparse coding with much lower computational cost.

The second project addresses challenging problems in biomedical image analysis. We conduct pioneering work and develop feature learning models for tissue image classification and nuclei segmentation. Specifically, on tissue image classification, we will discuss two methods, *i.e.,* Stacked Predictive Sparse Decomposition and Multi-spectral Convolutional Sparse Coding. On nuclei segmentation, we will present an

approach called, sparsity constrained convolutional regression. The proposed models can achieve very competitive results compared to dedicated systems using biological prior knowledge.

The third project develops a unified framework based on sparse representation for classifying complicated human gestures captured as multivariate time series (MTS). The model consists of a novel feature extractor and a kernel sparse representation classifier. The proposed feature extractor is invariant to temporal disordering and overcomes the inconsistent lengths problem among MTS data. In addition, we propose a new approach to kernelize sparse representation. Through kernelization, realized dictionary atoms are more separable for sparse coding algorithms and nonlinear relationships among data are conveniently transformed into linear relationships in the kernel space, which leads to more effective classification. Extensive experiments show that our method yields superior results compared to many previously reported algorithms.

1.5 Summary of Contributions

The main contributions of this dissertation are as followings:

- We theoretically demonstrate that the approximation to an unobservable intrinsic manifold by a few latent landmark points can be cast as a novel dictionary learning problem in the observation space. The derived locality-constrained dictionary learning algorithm has analytic solution and can be extended to discriminative learning tasks.

- We apply locality-constrained discriminative dictionary learning algorithms to many computer vision and pattern recognition tasks, *e.g.,* face recognition, action recognition, hyperspectral image classification, etc, demonstrating the promising performance of the proposed methods.

- By developing unsupervised feature learning systems, we first introduce this emerging technology into biomedical image analysis. More importantly, in tissue

image classification and nuclei segmentation, we show that automatically learned features can achieve very competitive results compared to human-engineered features based on biological prior knowledge.

- We propose a generic approach to kernelizing sparse representation, which is readily applicable to many existing sparse coding algorithms. In addition, for multivariate time series classification we develop a feature extractor, which corresponds to a valid kernel. Combining the two components, we derive a kernel sparse representation classification algorithm.

1.6 Related Publications to The Described Contributions

The contributions described in this dissertation first appeared in a number of publications. The following lists some publication highlights that roughly correspond to different chapters in the dissertation:

- Chapter 2: Y. Zhou and K. E. Barner, "Locality Constrained Dictionary Learning for Nonlinear Dimensionality Reduction", *IEEE Signal Processing Letters*, Vol 20, No. 4, 2013.

- Chapter 3a: Y. Zhou, K. Liu, K. E. Barner, "Non-Rigid 3D Shape Recognition via Dictionary Learning", in *Proceedings, IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, Vancouver, Canada, May 2013.

- Chapter 3b: Y. Zhou, J. Gao, K. E. Barner, "Locality Preserving KSVD for Nonlinear Manifold Learning", in *Proceedings, IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, Vancouver, Canada, May 2013.

- Chapter 4a: Y. Zhou, H. Chang, K. E. Barner, P. Spellman and B. Parvin, "Classification of Histology Sections Using Multispectral Convolution Sparse Coding", to appear in *Proceedings, IEEE International Conference on Computer Vision and Pattern Recognition*, 2014.

- Chapter 4b: H. Chang, Y. Zhou, P. Spellman and B. Parvin, "Stacked Predictive Sparse Coding for Classification of Distinct Regions in Tumor Histopathology", in *Proceedings, IEEE International Conference on Computer Vision*, 2013.

- Chapter 4c: Y. Zhou, H. Chang, K. E. Barner and B. Parvin, "Nuclei Segmentation via Sparsity Constrained Convolutional Regression", submitted to *International Symposium on Biomedical Imaging*, 2015.

- Chapter 5: Y. Zhou and K. Liu and R. E. Carrillo and K. E. Barner and F. Kiamilev, "Kernel based Sparse Representation for Gesture Recognition", *Pattern Recognition*, Vol 46, Issue 12, December 2013.

1.7 Organization

The rest of this dissertation is organized as follows. Chapter 2 introduces the locality-constrained dictionary learning algorithm for nonlinear dimensionality reduction. Chapter 3 further explores the effectiveness of locality-constrained coding in a variety of real-world classification problems. Chapter 4 presents feature learning systems for histopathology image classification and nuclei segmentation. Chapter 5 proposes a unified framework consisting of a feature extraction strategy and kernelized sparse coding for MTS data classification. Chapter 6 summarizes this dissertation and points out future directions.

Chapter 2

LOCALITY-CONSTRAINED DICTIONARY LEARNING

In this chapter, we first propose an efficient locality-constrained dictionary learning (LCDL) algorithm to address the critical problem of exorbitant computational complexity for nonlinear dimensionality reduction [47–52].

2.1 Introduction

Many computer vision and pattern recognition problems involve high-dimensional large-scale datasets that are computationally expensive to process. Nonlinear dimensionality reduction (NLDR) is an important technique that discovers the most succinct and intrinsic forms of representation of the original high-dimensional data, allowing more effective learning and prediction. Unfortunately, many existing NLDR algorithms are of quadratic or even cubic complexity in the number of data, which diminishes the applicability of these algorithms to real-world large-scale tasks [19]. Efforts have been made on selecting a subset of training data as landmark points on the manifold to improve the efficiency of NLDR algorithms. Landmark points are meaningful points that preserve the local geometric structure of a manifold. In [53], the authors suggest using a subset of randomly selected data points, which, however, may yield a locally optimal solution with poor global performance. Alternatively, [19] proposes utilizing LASSO regression to select landmark points, an approach that has high computational cost due to the required ℓ_1 minimization. The effective learning of landmark points, thus, remains an open challenge.

Sparse representation–based dictionary learning has been proven to be effective in image restoration [54], image denoising [33,34] and image classification [1]. However, algorithms of this type generally do not ensure locality preservation and thus fails

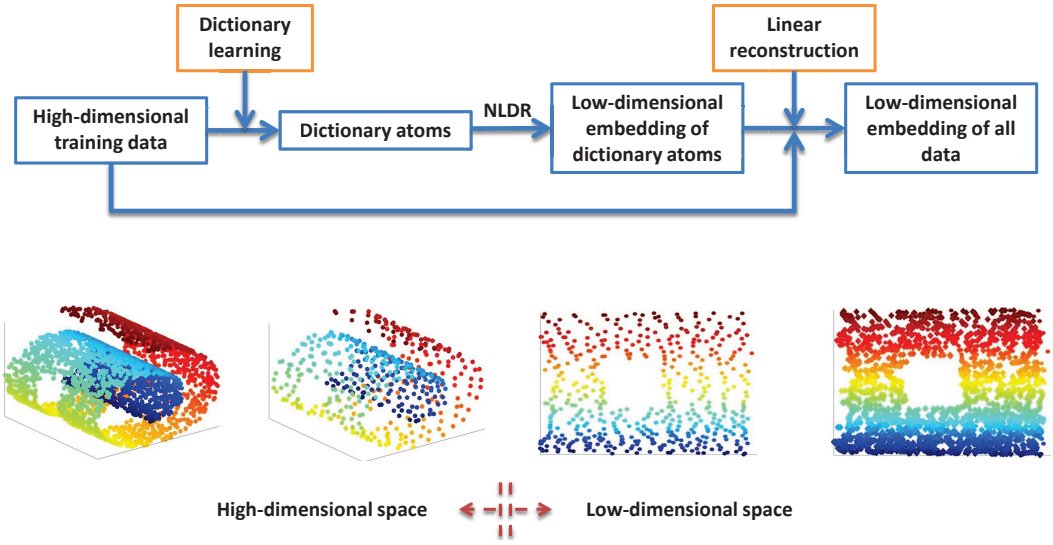


Figure 2.1: Overview of the proposed method. Given training data in high-dimensional observation space, a representational and locality-preserving dictionary is learned. Then, the low-dimensional embedding of the atoms is computed via some NLDR algorithm. Finally, using the geometric relationships among training data and the atoms in observation space, the low-dimensional embedding of training data is reconstructed as linear combinations of the low-dimensional embedding of the atoms.

to faithfully depict intrinsic manifold geometry. To address this issue, the approach in [31] approximates nonlinear functions via local coordinate coding. This method is based a modification to ℓ_1 minimization and, as such, has high computational cost. More recently proposed is a locality–constrained linear coding (LLC) approach that favors close samples and suppresses those distant [32]. Moreover, this approach has the advantage of an analytic solution.

In this work (Fig. 2.1), we show that reconstructing an unobservable intrinsic manifold via a few latent landmark points can be cast, under mild conditions, as a locality constrained dictionary learning problem in the observation space. Utilizing this approach, a novel locality constrained dictionary learning (LCDL) algorithm is introduced. The LCDL algorithm identifies a compact set of landmark points that are simultaneously representational and locality–preserving. Via the landmark points, LCDL naturally embeds training and unseen data onto the intrinsic manifold. Presented results demonstrate that LCDL can significantly improve the performance of

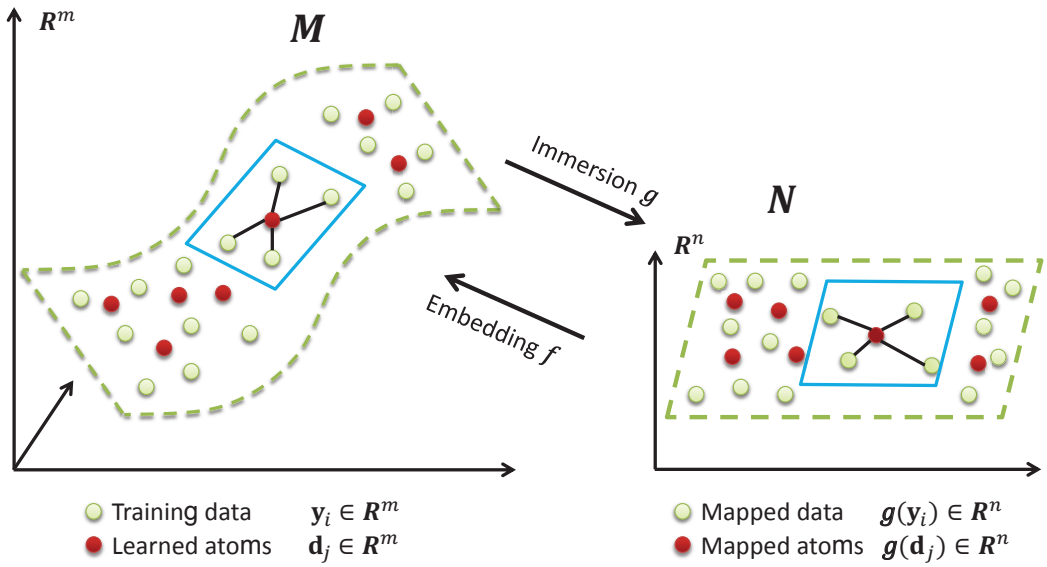


Figure 2.2: Illustration of the learning objective.

NLDR algorithms by yielding a more robust low–dimensional embedding at significantly reduced computational complexity.

2.2 LCDL Algorithm

2.2.1 Problem Formulation

Given an observation set $\{\mathbf{y}_i\}_{i=1}^N$ in $\mathbb{R}^m$, suppose all $\mathbf{y}_i$ reside on a smooth submanifold $\mathcal{M} \subset \mathbb{R}^m$, which is the image of a smooth n-manifold $\mathcal{N}$ under an embedding $f : \mathcal{N} \to \mathbb{R}^m$, where $n \ll m$. f is a diffeomorphism of $\mathcal{N}$ to $\mathcal{M}$ [55].

Let g denote the inverse mapping f^{-1} and let $g(\mathbf{y}_i) \in \mathbb{R}^n$ be the image of $\mathbf{y}_i$ via g located on $\mathcal{N}$. Define $\mathbf{x}_i \in \mathbb{R}^K$ as the local reconstruction code for representing $g(\mathbf{y}_i)$ as a linear combination of K landmarks. Our objective is to learn a codebook of landmark points on $\mathcal{M}$ in the observation space, *i.e.*, $\mathbf{D} = [\mathbf{d}_1, \ldots, \mathbf{d}_K] \in \mathbb{R}^{m \times K}$, $(K \ll N)$, such that $g(\mathbf{D})\mathbf{x}_i$ approximates $g(\mathbf{y}_i)$ in terms of ℓ_2 distance, for $i = 1, \ldots, N$. Here $g(\mathbf{D}) = [g(\mathbf{d}_1), \ldots, g(\mathbf{d}_K)] \in \mathbb{R}^{n \times K}$ is a matrix representing the image of the landmarks stored in $\mathbf{D}$ via the inverse mapping g, located on $\mathcal{N}$. An illustration of the proposed learning objective is shown in Fig. 2.2, where the green dots represent training data ($\mathbf{y}_i$ in the observation space and $g(\mathbf{y}_i)$ on the intrinsic manifold) and the red dots

represent the learned landmarks on the high-dimensional ($\mathbf{d}_j$) and low-dimensional manifold ($g(\mathbf{d}_j)$). Achieving this goal yields much more effective NLDR by learning only $K \ll N$ landmark points, making NLDR algorithms scalable to large dataset problems.

In practice, however, it is often infeasible to recover g due to the facts that: 1) the myriad of observed data causes intractable computation complexity and memory consumption; 2) the intrinsic manifold $\mathcal{N}$ is typically unknown. Without knowing g explicitly, even optimizing $\mathbf{D}$ on $\mathcal{N}$ becomes impractical. We therefore need to establish a relationship between the approximation problem among latent variables (*i.e.,* $g(\mathbf{y}_i)$ and $g(\mathbf{D})$) and the approximation problem among observation variables (*i.e.,* $\mathbf{y}_i$ and $\mathbf{D}$).

As noted by [47], $\mathbf{x}_i$ reflects intrinsic geometric properties of each neighborhood on $\mathcal{N}$ and these properties are expected to be equally valid for local patches on $\mathcal{M}$. We can therefore use the same set of local reconstruction codes to characterize the local geometric relationships between $g(\mathbf{y}_i)$ and $g(\mathbf{D})$ on $\mathcal{N}$ as to characterize those between $\mathbf{y}_i$ and $\mathbf{D}$ on $\mathcal{M}$.

2.2.2 Locality Constrained Dictionary Learning (LCDL)

By requiring $g(\mathbf{D})\mathbf{x}_i$ to approximate $g(\mathbf{y}_i)$ in terms of ℓ_2 distance, for $i = 1, \ldots, N$, we essentially obtain a representational $\mathbf{D}$ such that $\sum_{i=1}^{N} \|g(\mathbf{y}_i) - g(\mathbf{D})\mathbf{x}_i\|_2^2$ is minimized. For symmetry, we enforce $\mathbf{1}^{\mathrm{T}}\mathbf{x}_i = 1$ for all i such that the characterization of local geometry by $\mathbf{x}_i$ is invariant to scaling, rotation and shift of the coordinate system [47], where $\mathbf{1}$ is a column vector of all ones.

Lemma 1. *Let $\mathcal{M}$, $\mathcal{N}$ and g be as above. Let $\mathbf{p} \in \mathcal{U}_\mathbf{p}$ be an open subset of $\mathcal{M}$ with respect to $\mathbf{p}$, such that $\forall \mathbf{q} \in \mathcal{U}_\mathbf{p}$, the line segment $\overline{\mathbf{pq}}$ remains in $\mathcal{U}_\mathbf{p}$. If $|\partial g^s / \partial q^t| \leq c$, $1 \leq s \leq n$, $1 \leq t \leq m$, at every $\mathbf{q} \in \mathcal{U}_\mathbf{p}$, then we have $\forall \mathbf{q} \in \mathcal{U}_\mathbf{p}$ [55]:*

$$\|g(\mathbf{q}) - g(\mathbf{p})\|_2^2 \leq mnc^2 \|\mathbf{q} - \mathbf{p}\|_2^2 . \tag{2.1}$$

The proof can be derived as a generalization of the mean value theorem and as such we omit the steps for brevity (see [55] for details). Lemma 1 indicates that as $\mathcal{U}_{\mathbf{p}}$ shrinks to be a sufficiently small neighborhood of $\mathbf{p}$, $mnc^2 \left\|\mathbf{q}-\mathbf{p}\right\|_2^2 \longrightarrow \left\|g(\mathbf{q})-g(\mathbf{p})\right\|_2^2$. We use this observation below.

Theorem 1. *Let $g(\mathbf{y}_i)$, $\mathbf{y}_i$, $g(\mathbf{D})$, $\mathbf{D}$ and g be as above. Let $\mathbf{y}_i \in \mathcal{U}_{\mathbf{y}_i}$ and $\mathbf{D}\mathbf{x}_i \in \mathcal{U}_{\mathbf{D}\mathbf{x}_i}$ be open sets as in Lemma 1, that also satisfy $\mathbf{D}\mathbf{x}_i \in \mathcal{U}_{\mathbf{y}_i}$ and $\{\mathbf{d}_j | x_{ji} \neq 0, \forall j\} \subset \mathcal{U}_{\mathbf{D}\mathbf{x}_i}$ $\forall i$. If $\boldsymbol{1}^{\mathrm{T}}\mathbf{x}_i = 1$ and $\left\|\mathbf{x}_i\right\|_0 = \tau$ ($\tau \ll K$) for all i, then the following inequality holds:*

$$\sum_{i=1}^{N} \left\|g(\mathbf{y}_i) - g(\mathbf{D})\mathbf{x}_i\right\|_2^2 \leq \alpha \sum_{i=1}^{N} \left\|\mathbf{y}_i - \mathbf{D}\mathbf{x}_i\right\|_2^2 + \beta \sum_{i=1}^{N}\sum_{j=1}^{K} \left[x_{ji}^2 \left\|\mathbf{D}\mathbf{x}_i - \mathbf{d}_j\right\|_2^2\right] \tag{2.2}$$

where x_{ji} is the j-th element in vector $\mathbf{x}_i$, $\tau \in \mathbb{Z}^+$, and $\alpha = 2c_1$, $\beta = 2\tau c_2$, with $c_1 = \sup(\{|\partial g^s/\partial q^t| \mid \mathbf{q} \in \mathcal{U}_{\mathbf{y}_i}, \forall i,s,t\})$ and $c_2 = \sup(\{|\partial g^s/\partial q^t| \mid \mathbf{q} \in \mathcal{U}_{\mathbf{D}\mathbf{x}_i}, \forall i,s,t\})$. Note that i exclusively represents the indexes of $\mathbf{y}_i$ and its code $\mathbf{x}_i$ while j only denotes the j-th element in $\mathbf{x}_i$.

Proof. Denote by $\mathbf{Y} \in \mathbb{R}^{m\times N}$ the matrix containing all $\mathbf{y}_i$ and let $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_N] \in \mathbb{R}^{K\times N}$ be the matrix containing N local reconstruction codes. We have

$$\begin{aligned}
& \sum_{i=1}^{N} \left\|g(\mathbf{y}_i) - g(\mathbf{D})\mathbf{x}_i\right\|_2^2 \\
= \;& \left\|g(\mathbf{Y}) - g(\mathbf{D})\mathbf{X}\right\|_F^2 \\
\overset{(a)}{=} \;& \left\|g(\mathbf{Y}) - g(\mathbf{D}\mathbf{X}) + g(\mathbf{D}\mathbf{X}) - g(\mathbf{D})\mathbf{X}\right\|_F^2 \\
\overset{(b)}{\leq} \;& 2\left\|g(\mathbf{Y}) - g(\mathbf{D}\mathbf{X})\right\|_F^2 + 2\left\|g(\mathbf{D}\mathbf{X}) - g(\mathbf{D})\mathbf{X}\right\|_F^2 \\
\overset{(c)}{=} \;& 2\sum_{i=1}^{N} \left\|g(\mathbf{y}_i) - g(\mathbf{D}\mathbf{x}_i)\right\|_2^2 + 2\sum_{i=1}^{N} \left\|g(\mathbf{D}\mathbf{x}_i) - g(\mathbf{D})\mathbf{x}_i\right\|_2^2
\end{aligned} \tag{2.3}$$

where in (a) $g(\mathbf{D}\mathbf{X}) \in \mathbb{R}^{n\times N}$ is a matrix representing the image of the reconstructed signals $\mathbf{D}\mathbf{X}$ via g; (b) is from Cauchy-Schwarz inequality; in (c) $g(\mathbf{D}\mathbf{x}_i) \in \mathbb{R}^n$ is the i-th column in $g(\mathbf{D}\mathbf{X})$. Since $\mathbf{1}^{\mathrm{T}}\mathbf{x}_i = \sum_{j=1}^{K} x_{ji} = 1$ and $\left\|\mathbf{x}_i\right\|_0 = \tau$ for all i, Eq. (2.3)

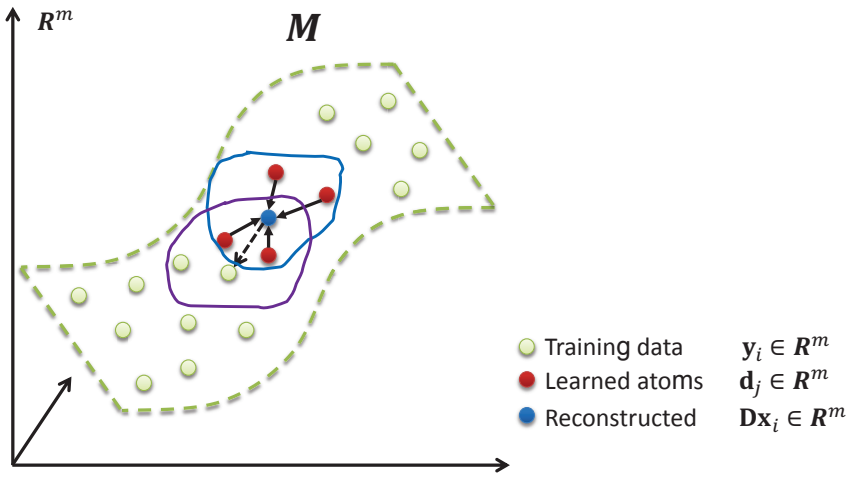


Figure 2.3: Illustration of LCDL algorithm.

can be written as:

$$\begin{aligned}
&\sum_{i=1}^{N} \|g(\mathbf{y}_i) - g(\mathbf{D})\mathbf{x}_i\|_2^2 \\
\leq\ & 2\sum_{i=1}^{N} \|g(\mathbf{y}_i) - g(\mathbf{D}\mathbf{x}_i)\|_2^2 + 2\sum_{i=1}^{N} \left\| \sum_{j=1}^{K} x_{ji} \Big[g(\mathbf{D}\mathbf{x}_i) - g(\mathbf{d}_j) \Big] \right\|_2^2 \\
\leq\ & 2\sum_{i=1}^{N} \|g(\mathbf{y}_i) - g(\mathbf{D}\mathbf{x}_i)\|_2^2 + 2\tau \sum_{i=1}^{N} \sum_{j=1}^{K} \left[x_{ji}^2 \, \|g(\mathbf{D}\mathbf{x}_i) - g(\mathbf{d}_j)\|_2^2 \right] \qquad (2.4)
\end{aligned}$$

Applying Lemma 1 to each $\|g(\mathbf{y}_i) - g(\mathbf{D}\mathbf{x}_i)\|_2^2$ and to each $\left[x_{ji}^2 \, \|g(\mathbf{D}\mathbf{x}_i) - g(\mathbf{d}_j)\|_2^2\right]$ in Eq. (2.4), $\exists\ c_1 = \sup\left(\{|\partial g^s / \partial q^t| \mid \mathbf{q} \in \mathcal{U}_{\mathbf{y}_i}, \forall i, s, t\}\right)$ and $c_2 = \sup(\{|\partial g^s / \partial q^t| \mid \mathbf{q} \in \mathcal{U}_{\mathbf{D}\mathbf{x}_i}, \forall i, s, t\})$ such that $2\|g(\mathbf{y}_i) - g(\mathbf{D}\mathbf{x}_i)\|_2^2 \leq 2c_1 \|\mathbf{y}_i - \mathbf{D}\mathbf{x}_i\|_2^2$, $\forall i$ and $2\tau[x_{ji}^2 \|g(\mathbf{D}\mathbf{x}_i) - g(\mathbf{d}_j)\|_2^2] \leq 2\tau c_2 [x_{ji}^2 \|\mathbf{D}\mathbf{x}_i - \mathbf{d}_j\|_2^2]$, $\forall i, j$. Letting $\alpha = 2c_1$ and $\beta = 2\tau c_2$ completes the result. □

Theorem 1 establishes a relationship between the latent variables and the observation variables by upper-bounding the approximation error on the intrinsic manifold $\mathcal{N}$ (LHS) in terms of the approximation error on $\mathcal{M}$ in the observation space (RHS). As indicated by Lemma 1, when $\mathbf{D}\mathbf{x}_i$ and all $\mathbf{d}_j \in \{\mathbf{d}_j | x_{ji} \neq 0, \forall j\}$ lie within a sufficiently small neighborhood of $\mathbf{y}_i$, the RHS of Eq. (2.2) $\longrightarrow$ the LHS of Eq. (2.2).

On the RHS, the 1-st term is the approximation error (reconstruction error term in dictionary learning literature) and the 2-nd term is the localization penalty term. An illustration of Theorem 1 can be found in Fig. 2.3, where for securing good approximation to each training sample ($\mathbf{y}_i$), we want the reconstructed signal $\mathbf{D}\mathbf{x}_i$ (blue dot) to be as close as possible to $\mathbf{y}_i$ (green dot); for preserving locality, we also want the neighborhood of $\mathbf{y}_i$ to be as small as possible, *i.e.,* the landmark points (red dots) contributing to reconstructing $\mathbf{y}_i$ should be close to the $\mathbf{D}\mathbf{x}_i$ (blue dot). By minimizing the RHS of Eq. (2.2) with respect to $\mathbf{D}$ and $\mathbf{x}_i$ for all i, we achieve faithful approximation (1-st term) and secure compact localization (2-nd term), *i.e.,* all $\mathbf{d}_j \in \{\mathbf{d}_j | x_{ji} \neq 0, \forall j\} \longrightarrow \mathbf{D}\mathbf{x}_i \longrightarrow \mathbf{y}_i$, indicating that $\beta \sum_{i=1}^{N} \sum_{j=1}^{K} [x_{ji}^2 \|\mathbf{D}\mathbf{x}_i - \mathbf{d}_j\|_2^2] \approx \beta \sum_{i=1}^{N} \sum_{j=1}^{K} [x_{ji}^2 \|\mathbf{y}_i - \mathbf{d}_j\|_2^2]$. We therefore formulate the practical LCDL optimization problem as:

$$\min_{\mathbf{D},\mathbf{X}} \quad \sum_{i=1}^{N} \|\mathbf{Y} - \mathbf{D}\mathbf{X}\|_F^2 + \lambda \sum_{i=1}^{N} \sum_{j=1}^{K} \left[x_{ji}^2 \|\mathbf{y}_i - \mathbf{d}_j\|_2^2\right] + \mu \|\mathbf{X}\|_F^2 \tag{2.5}$$

$$\text{s.t.} \quad \begin{cases} \mathbf{1}^{\mathrm{T}} \mathbf{x}_i = 1 \quad \forall i & (*) \\ x_{ji} = 0 \quad \text{if } \mathbf{d}_j \notin \Omega_\tau(\mathbf{y}_i) \quad \forall i, j & (**) \end{cases}$$

where $\Omega_\tau(\mathbf{y}_i)$ is defined as the τ-neighborhood containing τ nearest neighbors of $\mathbf{y}_i$, and λ, μ are positive regularization constants. $\mu \|\mathbf{X}\|_F^2$ is included for numerical stability of the least–squares solution. The sum-to-one constraint $(*)$ follows from the symmetry requirement, while the locality constraint $(**)$ ensures that $\mathbf{y}_i$ is reconstructed by atoms belonging to its τ-neighborhood, allowing $\mathbf{x}_i$ to characterize the intrinsic local geometry.

2.2.3 Optimization

An iterative process is employed for LCDL optimization. That is, the local reconstruction code $\mathbf{X}$ is optimized first, followed by $\mathbf{D}$. The iterations are repeated, with one aspect held fixed while the other is optimized. The repetition is terminated

once either objective function is below a preset threshold or a maximum number of iterations is reached.

2.2.3.1 Solving for Local Reconstruction Codes

Fixing $\mathbf{D}$, which is initialized or set from previous iteration, the i-th column $\mathbf{x}_i \in \mathbf{X}$ is obtained by solving:

$$\min_{\mathbf{x}_i} \quad \|\mathbf{y}_i - \mathbf{D}\mathbf{x}_i\|_2^2 + \lambda \sum_{j=1}^{K} \left[x_{ji}^2 \|\mathbf{y}_i - \mathbf{d}_j\|_2^2\right] + \mu \|\mathbf{x}_i\|_2^2 \tag{2.6}$$

$$\text{s.t.} \quad \begin{cases} \mathbf{1}^{\mathrm{T}}\mathbf{x}_i = 1 \\ x_{ji} = 0 \qquad \text{if } \mathbf{d}_j \notin \Omega_\tau(\mathbf{y}_i) \quad \forall j \end{cases}$$

Taking both of the constraints into consideration, and using Lagrange multiplier, we obtain

$$\mathcal{L}(\hat{\mathbf{x}}_i, \eta) = \|\mathbf{y}_i - \Omega_\tau \hat{\mathbf{x}}_i\|_2^2 + \lambda \sum_{\mathbf{d}_j \in \Omega_\tau} \left[x_{ji}^2 \|\mathbf{y}_i - \mathbf{d}_j\|_2^2\right] + \mu \|\hat{\mathbf{x}}_i\|_2^2 + \eta \left(\mathbf{1}^{\mathrm{T}}\hat{\mathbf{x}}_i - 1\right) \tag{2.7}$$

where for simplicity we express $\Omega_\tau(\mathbf{y}_i)$ as $\Omega_\tau \in \mathbb{R}^{m \times \tau}$ and $\hat{\mathbf{x}}_i$ as a succinct subvector containing only τ nonzero elements for those $\mathbf{d}_j \in \Omega_\tau$. Denote as $\mathbf{G} = (\Omega_\tau - \mathbf{y}_j\mathbf{1}^{\mathrm{T}})^{\mathrm{T}}(\Omega_\tau - \mathbf{y}_j\mathbf{1}^{\mathrm{T}})$ the local covariance matrix. Define $\delta(\cdot)$ as the operator that preserves only the diagonal of a square matrix and sets the remaining elements to zero. Thus $\delta(\mathbf{G})$ is a diagonal matrix of size τ-by-τ. For mathematical simplicity, we impose $\mathbf{1}^{\mathrm{T}}\hat{\mathbf{x}}_i = 1$ onto the 1-st term of Eq. (2.7) and get:

$$\mathcal{L}^*(\hat{\mathbf{x}}_i, \eta) = \hat{\mathbf{x}}_i^{\mathrm{T}}(\mathbf{G} + \lambda\delta(\mathbf{G}) + \mu\mathbf{I})\hat{\mathbf{x}}_i + \eta \left(\mathbf{1}^{\mathrm{T}}\hat{\mathbf{x}}_i - 1\right), \tag{2.8}$$

where $\mathbf{I}$ is the identity matrix. Setting $\nabla_{\hat{\mathbf{x}}_i}\mathcal{L}^*(\hat{\mathbf{x}}_i, \eta)$ and $\nabla_\eta\mathcal{L}^*(\hat{\mathbf{x}}_i, \eta)$ to zero, we obtain the solution as

$$\hat{\mathbf{x}}_i = \frac{(\mathbf{G} + \lambda\delta(\mathbf{G}) + \mu\mathbf{I})^{-1}\mathbf{1}}{\mathbf{1}^{\mathrm{T}}(\mathbf{G} + \lambda\delta(\mathbf{G}) + \mu\mathbf{I})^{-1}\mathbf{1}} \tag{2.9}$$

Although the formulations are different, we can still adopt the strategy in [47] to compute $\hat{\mathbf{x}}_i$ efficiently, *i.e.,* first solving the linear system of equations $(\mathbf{G} + \lambda\delta(\mathbf{G}) + \mu\mathbf{I})\hat{\mathbf{x}}_i = \mathbf{1}$ and then normalizing $\hat{\mathbf{x}}_i$ to satisfy the sum-to-one constraint. Note that in contrast with sparse coding algorithms, the proposed coding scheme has an analytic solution and thus is of substantially lower computational complexity.

2.2.3.2 Dictionary Optimization

Having obtained the optimal $\mathbf{X} \in \mathbb{R}^{K\times N}$, we now consider this term fixed and present a procedure for individually optimizing each atom of $\mathbf{D}$. Let $\mathbf{d}_j \in \mathbb{R}^m$ be the j-th atom in $\mathbf{D}$ and define row vector $\mathbf{x}_{j*} \in \mathbb{R}^{1\times N}$ as the j-th row of $\mathbf{X}$. With $\mathbf{X}$ and all other atoms fixed, we rewrite Eq. (4.5) and cast the optimization problem as

$$\begin{aligned}\min_{\mathbf{d}_j} \ \mathcal{H}(\mathbf{d}_j) \ &= \ \left\|\mathbf{Y} - \sum_{k\neq j}\mathbf{d}_k\mathbf{x}_{k*} - \mathbf{d}_j\mathbf{x}_{j*}\right\|_F^2 + \lambda\left\{\sum_{i=1}^{N}\left[x_{ji}^2\left\|\mathbf{y}_i - \mathbf{d}_j\right\|_2^2\right]\right. \\ &+ \ \left.\sum_{i=1}^{N}\sum_{k\neq j}^{K}\left[x_{ki}^2\left\|\mathbf{y}_i - \mathbf{d}_k\right\|_2^2\right]\right\}. \end{aligned} \tag{2.10}$$

Setting $\mathbf{E} = \mathbf{Y} - \sum_{k\neq j}\mathbf{d}_k\mathbf{x}_{k*}$ and eliminating irrelevant terms, Eq. (3.6) is simplified to

$$\min_{\mathbf{d}_j} \ \mathcal{H}(\mathbf{d}_j) = \mathrm{Tr}\left\{(\mathbf{E} - \mathbf{d}_j\mathbf{x}_{j*})(\mathbf{E} - \mathbf{d}_j\mathbf{x}_{j*})^{\mathrm{T}}\right\} + \lambda\sum_{i=1}^{N}\left[x_{ji}^2(\mathbf{y}_i - \mathbf{d}_j)^{\mathrm{T}}(\mathbf{y}_i - \mathbf{d}_j)\right] \tag{2.11}$$

Since $\mathcal{H}(\mathbf{d}_j)$ is convex, setting the gradient of $\mathcal{H}(\mathbf{d}_j)$ with respect to $\mathbf{d}_j$ to zero yields the optimal solution

$$\mathbf{d}_j = \frac{1}{(1+\lambda)(\mathbf{x}_{j*}\mathbf{x}_{j*}^{\mathrm{T}})}\left(\mathbf{E}\mathbf{x}_{j*}^{\mathrm{T}} + \lambda\mathbf{Y}\alpha\right) \tag{2.12}$$

where $\alpha = [x_{j1}^2, \ldots, x_{jN}^2]^{\mathrm{T}} \in \mathbb{R}^N$ is a column vector with terms the squared values of those in $\mathbf{x}_{j*}^{\mathrm{T}}$.

Discussion: In the framework of LCDL, the mapping g can be found by any NLDR algorithm. NLDR algorithms, however, require complexity $O(mN^2)$ or $O(mN^3)$ in time (depending on the specific formulation utilized) and $O(N^2)$ in space, when operating

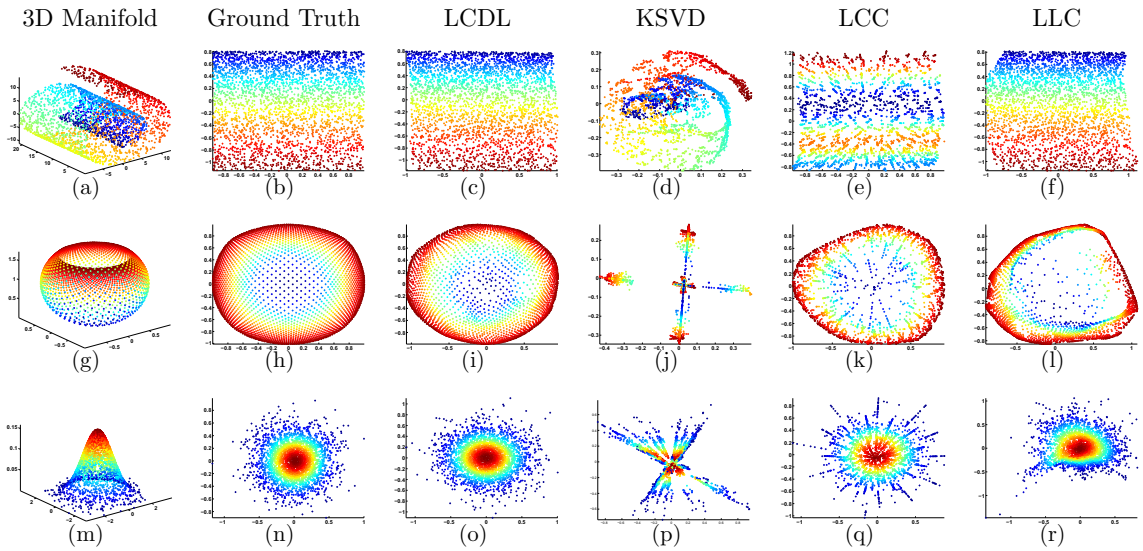


Figure 2.4: Low-dimensional embedding reconstruction comparison on Swiss roll (1st row), Punctured sphere (2nd row) and Gaussian (3rd row). Ground truth means the low-dimensional embedding obtained directly from all training samples. The nearest neighbor parameter k of NLDR algorithms is set to 6. The RMSE values are (c) 0.0299, (d) 0.7409, (e) 0.0666, (f) 0.0535, (i) 0.0705, (j) 0.8664, (k) 0.1060, (l) 0.1743, (o) 0.0104, (p) 0.2943, (q) 0.0419, (r) 0.1012.

on the full set of data. Utilizing a landmark points approach greatly reduces the NLDR complexity to $O(mK^2)$ or $O(mK^3)$ in time, again depending on the formulation utilized, and $O(K^2)$ in space, where $K \ll N$. The LCDL time complexity, for computing a single $\mathbf{x}_i$, is $O(mK) + O(m\tau^3)$, which is dominated by $O(mK)$ as $\tau^3 \ll K$. Additionally, the LCDL optimizing, for each $\mathbf{d}_j$, has time complexity $O(mN)$. The overall asymptotic complexity of LCDL is $O(mNK)$ per iteration. Though the convergence speed is task-dependent, the convergence to a local minimum is guaranteed and in our experiments 15 iterations are typically sufficient to achieve satisfactory results. When N is large, the LCDL complexity is negligible compared to that of NLDR. Thus LCDL, by efficiently establishing a faithful embedding and reconstruction representations, can significantly reduce the time and space complexity of NLDR algorithms, especially for large-scale datasets.

2.3 Experimental Results

Evaluation using Synthetic Manifolds

The proposed LCDL is evaluated by measuring the root mean square error (RMSE) introduced through the reconstruction of an intrinsic manifold $\mathcal{N}$, *i.e.*, $\|g(\mathbf{Y})-g(\mathbf{D})\mathbf{X}\|_F/\sqrt{N}$. LCDL is compared with three state-of-the-art DL algorithms, K-SVD [33], LCC [31] and the recently proposed LLC [32]. Note that $g(\mathbf{Y})$ and $g(\mathbf{D})$ are the low-dimensional embedding of training data and landmark points, respectively, computed via the NLDR algorithm, where $g(\mathbf{Y})$ is employed as the ground truth. Also, $\mathbf{X}$ is computed according to Eq. (3.5). For each synthetic dataset, $N = 3000$ training data are randomly generated, among which K samples are randomly selected for initialization. We set $K = 500, 200$, and 100 for the Swiss roll, Punctured sphere and Gaussian manifold, respectively. The NLDR algorithms are Hessian LLE [49], Laplacian Eigenmap [50], and LLE [47] for these three manifolds. We restrict training samples to be reconstructed by 2 atoms, as the intrinsic manifolds are 2D. The visualization and RMSE of the reconstructed low-dimensional manifolds are illustrated in Fig. 2.4. LCDL outperforms other competitive methods, yielding the closet approximation to the ground truth in all cases.

Evaluation using Face Recognition Datasets

Consider next the effectiveness in classification of the reconstructed low-dimensional manifolds produced by LCDL, with effectiveness determined through comparisons to the aforenoted methods. Though we only report classification results using LLE, drawn conclusions can be generalized to other NLDR algorithms.

The Extended YaleB Database [56] contains 2414 face images of 38 subjects. For each subject, we randomly select half of the images (about 32 per person) for training and the other half for testing. As in [57], we use a subset of the CMU PIE Database [58], *i.e.*, C05, C07, C09, C27, and C29, which yields a total 11554 images of 68 subjects. Following [57], a random selection of 130 images per person is employed to form the training set and the rest of the database is designated for testing. All images are normalized to 32×32 pixels and preprocessed by histogram equalization.

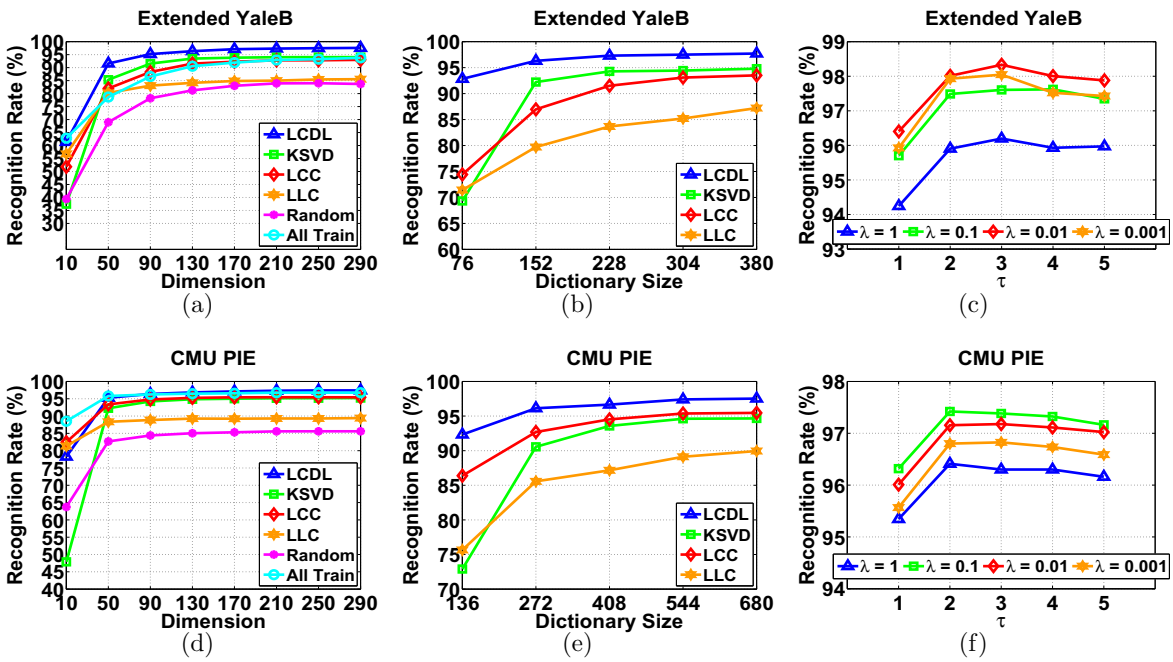


Figure 2.5: Classification results over two face databases. The parameter k of LLE is set to 60 for both Extended YaleB and CMU PIE.

The nearest neighbor classifier is employed and the results averaged over 10 repetitions are reported. The LCDL parameters are set as $\lambda = 0.1$, $\mu = 0.001$, and $\tau = 2$.

For all algorithms, a structured dictionary is learned as $\mathbf{D} = [\mathbf{D}_1|\mathbf{D}_2|\dots|\mathbf{D}_C]$, where $\mathbf{D}_i$ is the sub-dictionary for class i. As in [1], we fix the number of atoms per class to be 8, yielding a dictionary of 304 atoms for the Extended YaleB Database and a dictionary of 544 atoms for the CMU PIE Database. The sparsity factor is set, through exhaustive search optimization, to 24 and 32 for K-SVD over the two databases, respectively. All Train is selected as the baseline method, which represents the results obtained in performing LLE on the entire training set. Random means employing randomly selected training samples as the dictionary. The recognition rates versus dimension for all methods are illustrated in Fig. 2.5(a) and Fig. 2.5(d). The proposed LCDL achieves the highest accuracies, 97.5% on the Extended YaleB Database and 97.4% on the CMU PIE Database. LCDL outperforms All Train since the optimization ameliorates the noise and outlier effects within the training data, which leads

Table 2.1: The overall time (seconds) includes dictionary learning and training data embedding. Note the time measurement may vary based on different implementations.

	Extended YaleB		CMU PIE	
	Overall Time	Speedup	Overall Time	Speedup
All Train	22.1577 s	No	11807.3121 s	No
K-SVD [33]	71.2387 s	No	2751.2620 s	4.3x
LCC [31]	38.7172 s	No	1299.7146 s	9.1x
LLC [32]	11.6593 s	1.9x	69.2321 s	170.5x
LCDL	7.1001 s	3.1x	45.8025 s	257.8x

to more robust dimensionality reduction.

In addition, we evaluate the proposed approach by fixing the dimension and varying the number of atoms per class from 2 to 10, which yields the ratio $\frac{\sharp \text{ atoms}}{\sharp \text{ training samples}}$ between 6.25% to 31.25% and between 1.54% to 7.69% for the Extended YaleB Database and the CMU PIE Database respectively. As shown in Fig. 2.5(b) and Fig. 2.5(e), LCDL consistently produces higher accuracy than competing algorithms across a range of dictionary sizes. This results from the fact that LCDL establishes a dictionary that is both representational and locality preserving. Moreover, we examine the impact τ and λ have on LCDL performance. As shown in Fig. 2.5(c) and Fig. 2.5(f), LCDL maintains higher recognition rate than other methods over a wide range of τ and λ, indicating that performance is relatively robust to parameter value selections.

Finally, we evaluate the implementation efficiency of LCDL by measuring the speedup in terms of the overall training time compared to the All Train baseline, Table 2.1. The results show that LCDL is more efficient than comparison methods and significantly improves the learning efficiency of LLE by more than 2 orders of magnitude over the CMU PIE Database.

Discussion: For completeness, we summarize the computational complexity of these representative algorithms and compare with the proposed LCDL algorithm, as shown in Table 2.2. We list the computational complexity for both the dictionary learning step and the training data encoding step. For the purpose of keeping the notation consistent, τ represents the number of nearest neighbors for LLC and LCDL and denotes the sparsity level for K-SVD. Note that for iterative dictionary optimization algorithms,

Table 2.2: Comparison of computational complexity for all the methods, including the dictionary learning step and the training data encoding step.

Method	Dictionary Learning	Training Data Encoding
K-SVD [33, 59]	$O(\tau^2 NKt + 2mNKt)$	$O(2\tau mK + 2\tau^2 m + 2\tau(K+m) + \tau^3)$
LCC [31]	$O(mNKt) + O(mNKt\min\{m, K\})$	$O(mNKt\min\{m, K\})$
LLC [32]	$O(mNK^3) + O(mN\tau^3)$	$O(mN\tau^3)$
LCDL	$O(mNKt) + O(mN\tau^3 t)$	$O(mN\tau^3)$

i.e., K-SVD, LCC and LCDL, t represents the number of iterations.

2.4 Conclusion

We propose a novel algorithm, LCDL, that learns dictionary atoms as landmark points which are simultaneously representational and locality preserving. Experiments demonstrate that LCDL is superior to existing dictionary learning algorithms in terms of yielding more meaningful atoms for NLDR algorithms with greatly reduced computational complexity.

Chapter 3

DISCRIMINATIVE DICTIONARY LEARNING FOR CLASSIFICATION

3.1 3D Shape Recognition

In this section, we extend the previously introduced LCDL, by incorporating discriminative functional terms into the training objective. The proposed algorithm achieves very encouraging recognition results. Our work is the first attempt applying locality-constrained dictionary learning to 3D shape recognition.

3.1.1 Introduction

Accurately recognizing non-rigid 3D objects in real world has been a challenging topic in machine/computer-vision-based applications such as robotic control, surveillance, automatic navigation, assistive technology, etc [60]. To achieve this objective, effective feature extraction strategies and discriminative classification algorithms are much needed.

In order to extract robust features from 3D surface of non-rigid objects, many algorithms have been proposed. Typically, they can be categorized into global feature extraction, e.g., shape histograms [61], shape moments [62], spherical harmonics [63], etc, and local feature extraction, e.g., heat kernel signatures [64], meshSIFT [65], 3D SURF [60], etc. Experiments have demonstrated that the local feature based methods have obvious advantages for dealing with issues of noise and partial occlusion [60, 66]. In this work, we employ meshSIFT [65] algorithm to build 3D shape descriptors.

Once the features of an object are extracted, an effective classification algorithm is desired to identify the class label of an object. Among those proposed classification methods so far, we mainly investigate dictionary learning based approaches. Sparse

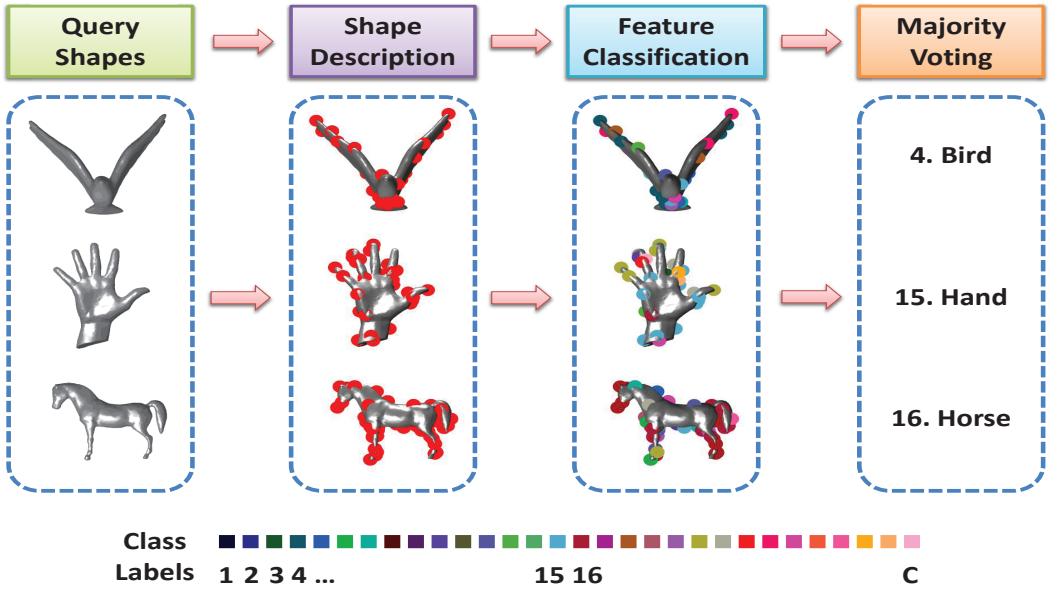


Figure 3.1: The proposed classification strategy. Given a query shape S, extract shape descriptors on it and then perform classification per descriptor. Finally the label of S is determined by majority voting over descriptor decisions.

coding solves for a compact and representational dictionary such that the large amount of training data can be expressed linearly by a few atoms in the dictionary. It has been proven that this model is effective in image restoration [54, 67], image denoising [33, 34], image classification [1, 11, 38], etc. In particular, Zhang *et al.* [1] proposed Discriminative-KSVD (D-KSVD) for face recognition by introducing label information into an objective function such that a representational dictionary and a linear classifier can be jointly optimized by using KSVD [33]. However, sparse coding based algorithms do not exploit the dependency information among local features and may therefore yield inconsistent representations of similar features [68, 69].

Locality-based coding was recently developed [31, 32] to address this issue. Particularly, Yu *et al.* [31] introduced Locally Coordinate Coding (LCC) to approximately express nonlinear functions as a linear combination of anchor points. Unfortunately, their coding strategy is based on a modification of LASSO (Least Absolute Shrinkage and Selection Operator) and hence suffers from high computation cost. Wang *et al.* [32] further proposed Locality-constrained Linear Coding (LLC) as a fast approximation to

LCC achieving impressive performance in image classification by using LLC codes as features and Support Vector Machine (SVM) as classifier. Nevertheless, little effort has been made to apply the aforementioned sparse or local coding techniques to non-rigid 3D shape recognition.

In this work, we extend the previously introduced LCDL algorithm and propose a novel algorithm, called dictionary learning based on supervised locally linear representation (DL-SLLR) for efficient 3D shape recognition. The main contribution is simultaneously incorporating a locality-preservation error term and the label approximation error term into the objective function. Unlike sparse coding based algorithms [1, 37–39, 70, 71], the proposed SLLR coding yields a closed-form solution. Moreover, the dictionary is optimized for both reconstruction and locality preservation, which therefore allows not only faithful reconstruction but also more consistent encoding of similar descriptors [68]. The proposed DL-SLLR is also different from recently proposed locality-based coding algorithms [31, 32, 72] in that 1) The SLLR coding is supervised such that training descriptors can only be coded by its same-class neighboring atoms, which thus yields a more discriminative dictionary; 2) A simple yet effective linear mapping is explicitly formulated into the unified objective function for classification.

To classify a query shape, we aggregate the predicted results of all descriptors using majority voting. Such a scheme requires negligible computational complexity and is invariant to rigid (rotation, scaling, and shift) and non-rigid (e.g., stretch, shrink and twist) transformations. Experiments over a newly generated SLI 3D Face Dataset and the SHREC'11 Contest Dataset validate the effectiveness of the proposed framework, *i.e.,* DL-SLLR in conjunction with majority voting.

3.1.2 The DL-SLLR Algorithm

Consider a C-label 3D shape classification problem. Let $\mathbf{Y}_i \in \mathbb{R}^{m \times n_i}$ be a set of m-dimensional n_i shape descriptors extracted from 3D objects with label i. Assign

label i to all descriptors in $\mathbf{Y}_i$. Set $\mathbf{Y} = [\mathbf{Y}_1, \mathbf{Y}_2, \ldots, \mathbf{Y}_C] \in \mathbb{R}^{m \times N}$ as the training set for all classes, where $N = \sum_{i=1}^{C} n_i$.

Let $\mathbf{D} = [\mathbf{D}_1, \mathbf{D}_2, \ldots, \mathbf{D}_C] \in \mathbb{R}^{m \times L}$ be a structured dictionary, where each $\mathbf{D}_i \in \mathbb{R}^{m \times K}$ is a class-specific sub-dictionary trained for $\mathbf{Y}_i$ and $L = KC$. Denote $\mathbf{x}_j \in \mathbb{R}^L$ as the sparse code of $\mathbf{y}_j$ over $\mathbf{D}$, where $\mathbf{y}_j \in \mathbf{Y}$ is the j-th descriptor in $\mathbf{Y}$, for $j = 1, \ldots, N$. We define $\mathbf{\Omega}_k(\mathbf{y}_j)$ as the same-class neighborhood with respect to $\mathbf{y}_j$ containing k-nearest-neighbor atoms from one particular sub-dictionary, which is pertaining to the same label as $\mathbf{y}_j$. Correspondingly, define $\mathbf{\Lambda}_{\mathbf{d}_i} \triangleq \{\mathbf{y}_j |\ \forall j,\ x_{ij} \neq 0,\ \mathbf{y}_j \in \mathbf{Y}\}$ as a neighborhood of $\mathbf{d}_i$, containing all $\mathbf{y}_j$ that are concurrently selecting $\mathbf{d}_i$ as one of their neighboring atoms, where x_{ij} is the i-th element in vector $\mathbf{x}_j$.

The goal at hand is achieving two objectives. The first is establishing a discriminative dictionary structured as $\mathbf{D} = [\mathbf{D}_1, \mathbf{D}_2, \ldots, \mathbf{D}_C]$ such that each $\mathbf{D}_i$ is independently trained for $\mathbf{Y}_i$ and every atom $\mathbf{d}_i$ preserves the locality of its neighborhood $\mathbf{\Lambda}_{\mathbf{d}_i}$. The second objective is realizing a linear mapping $\mathbf{W} \in \mathbb{R}^{C \times L}$ that transforms the sparse code $\mathbf{x}_j$ of every descriptor $\mathbf{y}_j$ to its label vector $\mathbf{h}_j = [0, \ldots, 1, \ldots, 0]^{\mathrm{T}}$, where the index of element 1 indicates the label of $\mathbf{y}_j$. Thus, the dictionary learning problem (DL-SLLR) is formalized as:

$$\begin{aligned} \min_{\mathbf{D},\mathbf{W},\mathbf{X}} \quad & \left(\begin{array}{l} \|\mathbf{Y} - \mathbf{D}\mathbf{X}\|_F^2 + \alpha \sum_{i=1}^{K} \sum_{\mathbf{y}_j \in \mathbf{\Lambda}_{\mathbf{d}_i}} \|\mathbf{y}_j - \mathbf{d}_i\|_2^2 \\ \\ +\beta \|\mathbf{H} - \mathbf{W}\mathbf{X}\|_F^2 + \gamma \|\mathbf{X}\|_F^2 + \mu \|\mathbf{W}\|_F^2 \end{array} \right) \\ \text{s.t.} \quad & x_{ij} = 0 \qquad \text{if} \quad \mathbf{d}_i \notin \mathbf{\Omega}_k(\mathbf{y}_j) \\ & \mathbf{1}^{\mathrm{T}} \mathbf{x}_j = 1 \quad \forall i, j \end{aligned} \tag{3.1}$$

where $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_N] \in \mathbb{R}^{L \times N}$ contains the SLLR codes (discussed in the next section) for descriptors in $\mathbf{Y} \in \mathbb{R}^{m \times N}$, and x_{ij} is the i-th element in column vector $\mathbf{x}_j \in \mathbf{X}$. Obeying the standard meaning, the first and the third terms represent the reconstruction and the classification errors respectively. The second term is the supervised locality-preservation error term, which ensures that every atom is close to those training samples that concurrently choose it as one of their neighboring atoms.

It therefore encourages atom consistency in local representations of similar descriptors [68]. Note that $\gamma \|\mathbf{X}\|_F^2$ and $\mu \|\mathbf{W}\|_F^2$ are regularization penalty terms, included for numerical stability, with γ and μ small positive constants. In addition, the first constraint requires each descriptor to be reconstructed only by its same-class neighboring atoms, ensuring that every sub-dictionary $\mathbf{D}_i$ is trained from the $\mathbf{Y}_i$ independently. The second constraint allows the coding to be shift-invariant, in which $\mathbf{1}$ is a column vector of all ones.

3.1.3 Optimization

The dictionary learning problem can be solved by iteratively repeating the following two steps to reduce the objective function, *i.e.,* first solving for the code matrix $\mathbf{X}$ with the other two variables fixed, and then updating $\mathbf{D}$ and $\mathbf{W}$, respectively. The iterations are terminated if either the objective function value is below some preset threshold or a maximum number of iterations has been reached.

3.1.3.1 Supervised Locally Linear Representation

Consider first solving for the SLLR code $\mathbf{x}_j \in \mathbf{X}$, for all $j = 1, \ldots, N$, with $\mathbf{D}$, $\mathbf{W}$ fixed. Define $\hat{\mathbf{y}}_j = \left[\mathbf{y}_j^{\mathrm{T}}, \sqrt{\beta}\mathbf{h}_j^{\mathrm{T}}\right]^{\mathrm{T}} \in \mathbb{R}^{m+C}$ as the j-th augmented training sample in the augmented training set $\hat{\mathbf{Y}} = \left[\mathbf{Y}^{\mathrm{T}}, \sqrt{\beta}\mathbf{H}^{\mathrm{T}}\right]^{\mathrm{T}} \in \mathbb{R}^{(m+C)\times N}$. Likewise denote $\hat{\mathbf{D}} = \left[\mathbf{D}^{\mathrm{T}}, \sqrt{\beta}\mathbf{W}^{\mathrm{T}}\right]^{\mathrm{T}} \in \mathbb{R}^{(m+C)\times L}$ as the augmented dictionary. Furthermore, set $\hat{\boldsymbol{\Omega}}_k(\mathbf{y}_j) = \{\hat{\mathbf{d}}_i |\ \forall i,\ \mathbf{d}_i \in \boldsymbol{\Omega}_k(\mathbf{y}_j),\ \hat{\mathbf{d}}_i \in \hat{\mathbf{D}}\}$ as the augmented neighborhood with respect to $\mathbf{y}_j$, with $\hat{\mathbf{d}}_i$ being the i-th column in $\hat{\mathbf{D}}$. Thus, minimizing Eq. (3.13) with respect to $\mathbf{x}_j$, is equivalent to solving the following locally linear representation (LLR) problem [47, 73] under the same-class neighborhood constraint.

$$\begin{aligned} \min_{\mathbf{x}_j} \quad & \|\hat{\mathbf{y}}_j - \hat{\mathbf{D}}\mathbf{x}_j\|_2^2 + \gamma \|\mathbf{x}_j\|_2^2 \\ \text{s.t.} \quad & x_{ij} = 0 \qquad \text{if} \quad \hat{\mathbf{d}}_i \notin \hat{\boldsymbol{\Omega}}_k(\mathbf{y}_j) \\ & \mathbf{1}^{\mathrm{T}}\mathbf{x}_j = 1 \quad \forall i \end{aligned} \tag{3.2}$$

Taking both of the constraints into consideration simultaneously, and using Lagrange multiplier, we get

$$\mathcal{J}(\tilde{\mathbf{x}}_j, \eta) = \|\hat{\mathbf{y}}_j - \hat{\mathbf{\Omega}}_k \tilde{\mathbf{x}}_j\|_2^2 + \gamma \|\tilde{\mathbf{x}}_j\|_2^2 + \eta \left(\mathbf{1}^{\mathrm{T}} \tilde{\mathbf{x}}_j - 1\right) \tag{3.3}$$

where for simplicity we express $\hat{\mathbf{\Omega}}_k(\mathbf{y}_j)$ as $\hat{\mathbf{\Omega}}_k \in \mathbb{R}^{(m+C)\times k}$, and $\tilde{\mathbf{x}}_j$ is a succinct vector containing only the nonzero coefficients for those $\hat{\mathbf{d}}_i \in \hat{\mathbf{\Omega}}_k(\mathbf{y}_j)$. Denote as $\mathbf{G} = (\hat{\mathbf{\Omega}}_k - \hat{\mathbf{y}}_j \mathbf{1}^{\mathrm{T}})^{\mathrm{T}}(\hat{\mathbf{\Omega}}_k - \hat{\mathbf{y}}_j \mathbf{1}^{\mathrm{T}})$ the local covariance matrix. Then Eq. (3.3) can be written as:

$$\mathcal{J}(\tilde{\mathbf{x}}_j, \eta) = \tilde{\mathbf{x}}_j^{\mathrm{T}} (\mathbf{G} + \gamma \mathbf{I}) \tilde{\mathbf{x}}_j + \eta \left(\mathbf{1}^{\mathrm{T}} \tilde{\mathbf{x}}_j - 1\right) \tag{3.4}$$

where $\mathbf{I}$ is the identity matrix. Setting $\nabla_{\tilde{\mathbf{x}}_j} \mathcal{J}(\tilde{\mathbf{x}}_j, \eta)$ and $\nabla_{\eta} \mathcal{J}(\tilde{\mathbf{x}}_j, \eta)$ to zero yields the desired closed-form solution, as

$$\tilde{\mathbf{x}}_j = \frac{(\mathbf{G} + \gamma \mathbf{I})^{-1} \mathbf{1}}{\mathbf{1}^{\mathrm{T}} (\mathbf{G} + \gamma \mathbf{I})^{-1} \mathbf{1}} \tag{3.5}$$

As suggested by [47], a more efficient way to compute $\tilde{\mathbf{x}}_j$ is by first solving the linear system of equations $(\mathbf{G} + \gamma \mathbf{I}) \tilde{\mathbf{x}}_j = \mathbf{1}$ and then normalizing $\tilde{\mathbf{x}}_j$ to satisfy the sum-to-one constraint. We adopt this for practical implementation.

Note that the proposed SLLR is different from LLR [47,73] in that 1) SLLR coding is supervised, which yields discriminative local reconstruction coefficients; 2) SLLR is performed over a compact dictionary and is combined with dictionary optimization, which in turn helps further reduce the reconstruction error.

3.1.3.2 Updating the Dictionary and the Mapping

Next, consider the update of $\mathbf{D}$ and $\mathbf{W}$, with $\mathbf{X}$ fixed. We individually optimize each atom of $\mathbf{D}$. Let $\mathbf{d}_i \in \mathbb{R}^m$ be the i-th atom in $\mathbf{D}$ and define $\mathbf{x}_{i*} \in \mathbb{R}^{1\times N}$ as the i-th row of $\mathbf{X}$. With $\mathbf{X}$ and the other atoms fixed, we rewrite Eq. (3.13) and cast the

optimization problem with respect to $\mathbf{d}_i$ as

$$\min_{\mathbf{d}_i} \ \mathcal{H}(\mathbf{d}_i) = \left\| \mathbf{Y} - \sum_{l \neq i} \mathbf{d}_l \mathbf{x}_{l*} - \mathbf{d}_i \mathbf{x}_{i*} \right\|_F^2 + \alpha \left\{ \sum_{\mathbf{y}_j \in \mathbf{\Lambda}_{\mathbf{d}_i}} \|\mathbf{y}_j - \mathbf{d}_i\|_2^2 + \sum_{l \neq i} \sum_{\mathbf{y}_j \in \mathbf{\Lambda}_{\mathbf{d}_l}} \|\mathbf{y}_j - \mathbf{d}_l\|_2^2 \right\} \tag{3.6}$$

Letting $\mathbf{E} = \mathbf{Y} - \sum_{l \neq i} \mathbf{d}_l \mathbf{x}_{l*}$ and rearranging Eq. (3.6), we have

$$\min_{\mathbf{d}_i} \ \mathcal{H}(\mathbf{d}_i) = \mathrm{Tr}\left\{ (\mathbf{E} - \mathbf{d}_i \mathbf{x}_{i*})(\mathbf{E} - \mathbf{d}_i \mathbf{x}_{i*})^{\mathrm{T}} \right\} + \alpha \sum_{\mathbf{y}_j \in \mathbf{\Lambda}_{\mathbf{d}_i}} \left[(\mathbf{y}_j - \mathbf{d}_i)^{\mathrm{T}} (\mathbf{y}_j - \mathbf{d}_i) \right] \tag{3.7}$$

Note that $\mathcal{H}(\mathbf{d}_i)$ is convex. Hence, setting the gradient of $\mathcal{H}(\mathbf{d}_j)$ with respect to $\mathbf{d}_i$ to zero yields the updated atom $\mathbf{d}_i^{new}$ as

$$\mathbf{d}_i^{new} = \frac{1}{(\mathbf{x}_{i*} \mathbf{x}_{i*}^{\mathrm{T}} + \alpha |\mathbf{\Lambda}_{\mathbf{d}_i}|)} \left(\mathbf{E} \mathbf{x}_{i*}^{\mathrm{T}} + \alpha \sum_{\mathbf{y}_j \in \mathbf{\Lambda}_{\mathbf{d}_i}} \mathbf{y}_j \right) \tag{3.8}$$

where $|\mathbf{\Lambda}_{\mathbf{d}_i}|$ denotes the cardinality of set $\mathbf{\Lambda}_{\mathbf{d}_i}$. Applying Eq. (3.8) to all $\mathbf{d}_i$, for $i = 1, \ldots, L$, completes the dictionary update in the current iteration.

In order to update $\mathbf{W}$, we solve the multivariate ridge regression [74] problem as

$$\mathbf{W}^{new} = \arg\min_{\mathbf{W}} \|\mathbf{H} - \mathbf{W}\mathbf{X}\|_F^2 + \mu \|\mathbf{W}\|_F^2 \tag{3.9}$$

where μ is a small positive constant for numerical stability. The solution is easily obtained as

$$\mathbf{W}^{new} = \mathbf{H}\mathbf{X}^{\mathrm{T}} (\mathbf{X}\mathbf{X}^{\mathrm{T}} + \mu \mathbf{I})^{-1} \tag{3.10}$$

Minimizing the objective function, we will obtain the optimal dictionary $\mathbf{D}$, which is representational for reconstructing training descriptors and capable of preserving locality of the data manifold, and we will also get the optimal mapping $\mathbf{W}$ in approximating the label matrix.

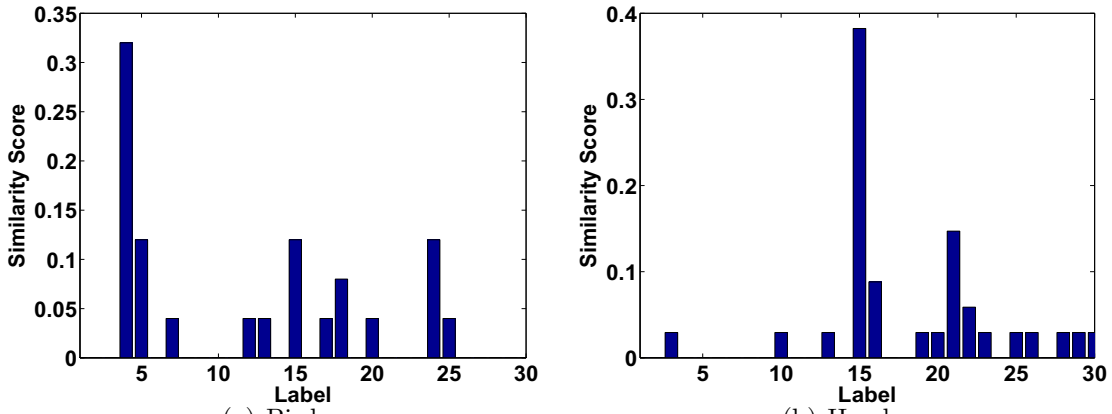
3.1.3.3 Classification Strategy

Human can distinguish different classes of objects (see Fig. 3.1) with mutual similarity in shape, even without using the clue of size, color and texture because we can make judgements based on seeking and comparing the most distinctive shape characteristics among those objects, despite the presence of a large portion of mutual similarity [75]. In other words, it is the most distinctive shape features of an object that plays the critical role in a successful recognition. We propose to emulate this process by employing majority voting and apply it to classifying 3D objects based on the newly proposed dictionary learning algorithm. Simplistically, we may assume that the votes from nondistinctive shape descriptors are approximately evenly spread across similar classes. Thus the outcome is the class that wins the most votes from the distinctive descriptors. Majority voting is an aggregation process (in which we need no explicit knowledge about which descriptors are distinctive or not) and its result is determined by the highest accumulated votes on a particular class. Visualizing the vote distribution of two objects from the SHREC'11 Contest Dataset [76], we can see in Fig. 3.2 that although a large portion of votes go to incorrect classes, the true class clearly receives the highest number of votes compared with any incorrect class.

Given a query object S, denote $\mathbf{Q}_S = [\mathbf{q}_1, \ldots, \mathbf{q}_n] \in \mathbb{R}^{m \times n}$ as the set of n extracted shape descriptors. The local reconstruction code $\mathbf{x}_j$ for each $\mathbf{q}_j$ is computed by solving

$$\begin{aligned} \min_{\mathbf{x}_j} \quad & \|\mathbf{q}_j - \mathbf{D}\mathbf{x}_j\|_2^2 + \gamma \|\mathbf{x}_j\|_2^2 \\ \text{s.t.} \quad & x_{ij} = 0 \qquad \text{if} \quad \mathbf{d}_i \notin \mathbf{\Gamma}_t(\mathbf{q}_j) \ \ \forall i \\ & \mathbf{1}^{\mathrm{T}}\mathbf{x}_j = 1 \end{aligned} \tag{3.11}$$

where x_{ij} is the i-th element in vector $\mathbf{x}_j \in \mathbb{R}^L$ and $\mathbf{\Gamma}_t(\mathbf{q}_j)$ is a neighborhood set consisting of t nearest-neighbor atoms of $\mathbf{q}_j$. The solution is given previously as Eq. (3.5).



(a) Bird (b) Hand

Figure 3.2: Majority voting results after normalization on SHREC'11 Contest Dataset. The two objects are bird (a) and hand (b). The bird is associated to label 4 while the hand is associated to label 15. Since the number of extracted descriptors varies across different objects, we normalize the voting results for better visualization.

Next, compute the projection $\mathbf{r} = \mathbf{W}\mathbf{x}_j \in \mathbb{R}^C$ and assign the label l_j to descriptor $\mathbf{q}_j$ according to

$$l_j = \arg\max_i (\mathbf{r} = [r_1, \ldots, r_i, \ldots, r_C]^{\mathrm{T}}) \tag{3.12}$$

Applying the same procedure to all $\mathbf{q}_j \in \mathbf{Q}$, a label vector is formed as $\mathbf{l} = [l_1, l_2, \ldots, l_n]$. Finally, we count the votes for each class label based on $\mathbf{l}$ and classify the query shape S according to the label receiving the most votes.

3.1.4 Experimental Results

The proposed DL-SLLR algorithm is evaluated using majority voting as classification scheme over two large datasets, the SLI 3D Face Dataset and the SHREC'11 Contest Dataset [76][1]. The proposed method is compared with D-KSVD [1] using majority voting and with a baseline SVM [77] method with Gaussian kernel using bag-of-words histogram (BoWH + SVM). The shape descriptors are extracted using meshSIFT [65]. Training parameters for DL-SLLR are $k \in \{2, 3, 4, 5\}$, $\alpha = \beta = 0.01$,

[1] Accessible at:http://www.itl.nist.gov/iad/vug/sharp/contest/2011/NonRigid/data.html

Table 3.1: Recognition results on SLI 3D Face Dataset.

Method	Proposed	D-KSVD [1]	BoWH + SVM [77]	Smeet's [65]
Accuracy	**96.00**%	95.78%	90.63%	91.67%

$\gamma = \mu = 0.001$ over both datasets. The neighborhood size t for classification is set to 10 and 6 for the face and the SHREC'11 datasets respectively.

Evaluation using SLI 3D Face Dataset

First presented are classification results over a newly generated Structured Light Illumination 3D Face Dataset (SLI 3D Face Dataset) [78]. This dataset is collected using the algorithm and hardware implementation developed in [79, 80]. It contains 576 high-quality dense 3D point clouds (approximately 5000 points per face) for 24 subjects with 4 static facial expressions under 3 different view angles. The population of 24 volunteers consists of 7 females and 17 males. Data for each individual is collected over two recording sessions in a dark room. During each session, an individual is required to face the camera at 3 different angles, *i.e.,* $\pm 45°$ (frontal right/left) and $0°$ (up-front), while at each angle performing 4 kinds of static facial expressions, *i.e.,* neutral, sad, happy, and anger.

Preprocessing the point clouds for classification, we use the depth information to segment subjects from the background and then manually crop the face area for each subject with a 3D bounding box. We employ the same subset of the database as [78] for evaluation. The total number of meshSIFT descriptors extracted from training faces is approximately 70,000. For DL-SLLR and D-KSVD, a dictionary of $L = 4800$ atoms is trained for classification, *i.e.,* $K = 200$ atoms per class. The results are reported based on 4-fold cross-validation over a repetitions. As shown in Table 3.1, the proposed approach outperforms other competitive methods yielding the highest recognition rate of 96.00%.

Evaluation using The SHREC'11 Contest Dataset

The SHREC'11 Contest Dataset [76] consists of 600 non-rigid 3D objects from 30 classes represented as watertight triangle meshes, including alien, horse, lamp, etc., as shown in Fig. 3.3 and Fig. 3.4. Each class equally has 20 objects. The total number of shape

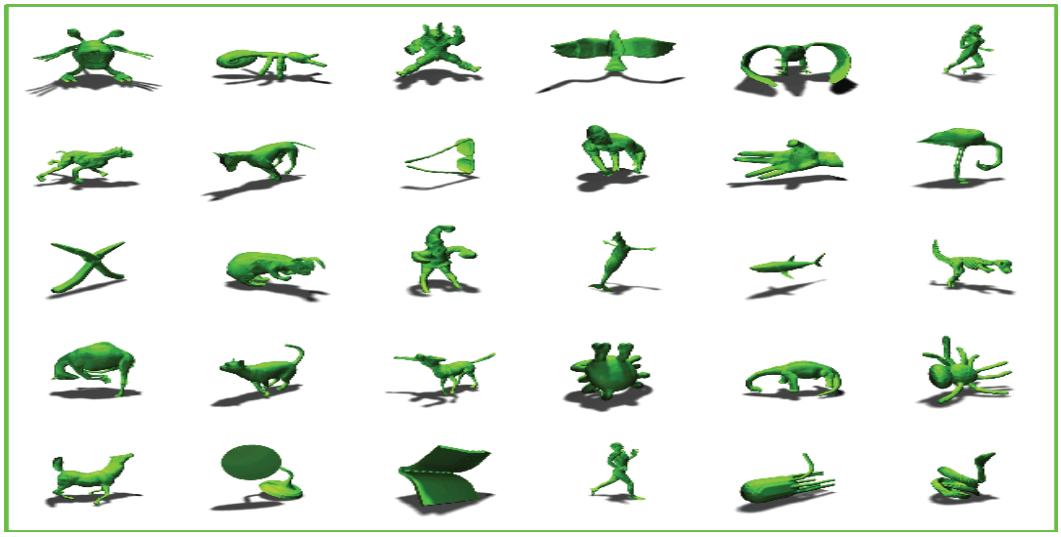


Figure 3.3: 30 classes from SHREC'11 Contest Dataset. Image cited from SHREC'11 Contest website.

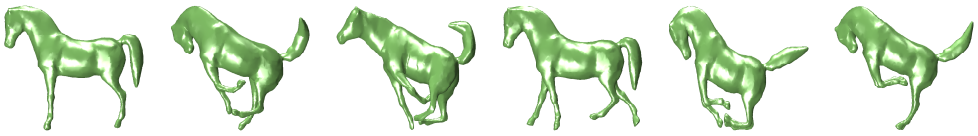


Figure 3.4: 3D Nonrigid shapes from object class horse.

Table 3.2: Recognition results on SHREC'11 Contest Dataset.

Method	Proposed	D-KSVD [1]	BoWH + SVM [77]	Smeet's [76]
Accuracy	**99.67**%	96.67%	98.00%	90.00%

descriptors extracted from training objects is approximately 380,000. For DL-SLLR and D-KSVD, a classification dictionary of $L = 6000$ atoms is trained, *i.e.,* $K = 200$ atoms per class. We conduct 10-fold cross-validation over the entire dataset and report averaged recognition results over 20 repetitions. As shown in Table 3.4, the proposed DL-SLLR with majority voting achieves the highest recognition rate of 99.67%. Finally, we study the robustness of aforementioned methods against to partial occlusions. Fig. 3.5 shows the performance of the methods under the conditions of varying percentage of occlusion. Clearly, the proposed approach (DL-SLLR in conjunction with majority) outperforms other methods.

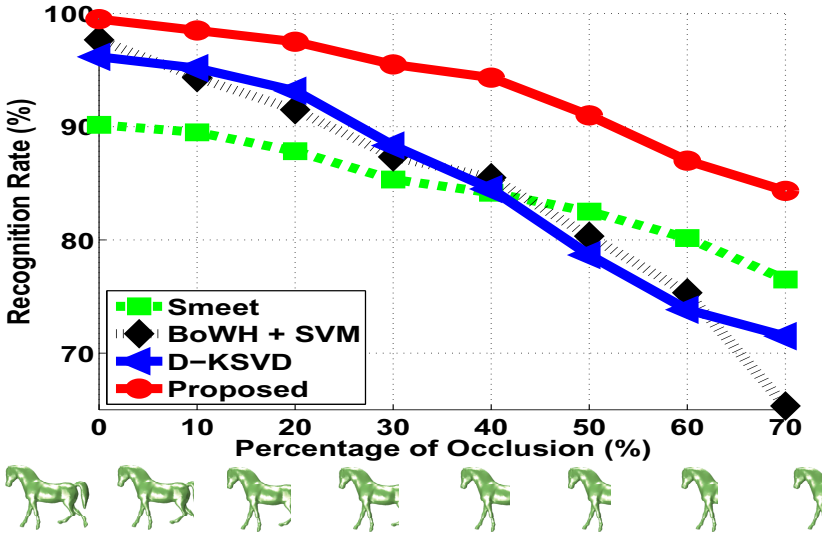


Figure 3.5: Comparison of performance for all methods on the robustness against partial occlusion.

3.2 Image/Video and Data Classification

In this section, we demonstrate the effectiveness of locality-constrained coding in traditional computer vision and pattern recognition tasks, *e.g.,* face recognition, action recognition, etc. Our work indicates that by imposing locality constraint, the proposed algorithm (DL-LPCP) can achieve superior classification performance compared to its sparse-coding based counterparts, *e.g.,* KSVD [33], D-KSVD [1] and SRC [11].

3.2.1 Introduction

Sparse coding solves inverse problems to find efficient expressions of data points as a linear combination of a few atoms in the dictionary or codebook. This model has proven effective in image restoration [54, 81], image denoising [33, 34] and image classification [1, 11, 38]. In [11], Sparse Representation based Classification (SRC) is proposed, which employs the totality of training data as the dictionary and achieves impressive results on face recognition.

To scale to large image classification datasets, many dictionary learning (DL) algorithms have been developed to learn a compact dictionary while trading-off some

discriminative terms, such as the Fisher discrimination term [36], the classifier prediction error [37], the incoherence promoting term [38], etc. By including label information and using KSVD [33], Zhang *et al.* [1] proposed Discriminative-KSVD (D-KSVD) for face recognition and Jiang *et al.* [39] further added a label consistent constraint into the objective function to enforce the correspondence between labels and atoms. These DL algorithms encode signals based on a global coordinate system and thus fail to exploit the locality in feature space, which could degrade the effectiveness of these methods in modeling data residing on nonlinear manifold [31]. Moreover, these DL algorithms require memory and computation intensive re-training when new classes are included, which would diminish their applicability in real-world user-centric recognition systems, for example when upgrading such a system with the inclusion of new user face data or user-customized gestures.

Some recent works have been proposed to exploit nonlinear structure of feature space utilizing locality constraints [31, 32]. In particular, Yu *et al.* [31] theoretically proved that nonlinear functions can be linearly approximated by a set of anchor points if certain locality requirements are satisfied. Their work suggests that locality can be more essential than sparsity in representing data distributed on nonlinear manifold. However, their coding strategy is based on ℓ_1 minimization and hence is of high computational complexity. Wang *et al.* [32] further proposed Locality-constrained Linear Coding (LLC) as a fast approximation to LCC and achieved impressive performance in image classification by using LLC codes as features for SVM. Although these methods are effective in learning a codebook for local representation, they cannot be directly employed for classification, as they do not include a discriminative penalty term into the objective function.

In this work, we present a novel and highly-efficient dictionary learning algorithm (DL-LPCP) by introducing the Locality-Preserving Constraint Pair (LPCP). DL-LPCP is a unified optimization scheme consisting of Supervised Local Coding (SLC) and Locality-Preserving Dictionary Update. Under the proposed LPCP, each labeled data point is encoded by its nearest same-class dictionary atoms based on

SLC codes, a representational dictionary together with a discriminative scaling matrix. Locality-preservation and linear mapping are jointly obtained through the proposed dictionary optimization approach. Additionally, a new classification strategy is proposed, exploiting both the representational dictionary and the locality-preserving dictionary.

Compared to existing methods [1, 31, 32, 36–39], our approach imposes explicit correspondence between labeled data and atoms via discriminatively exploiting the locality of feature space, which effectively encourages more consistent encoding of similar features. Moreover, DL-LPCP possesses the advantage of class-independent training, and thus is potentially more suitable for real-world applications where timely system upgrade is of great necessity.

3.2.2 The DL-SLC Algorithm

We consider a C-label classification problem. Let $\mathbf{Y}_i \in \mathbb{R}^{m\times n_i}$ be a set of m-dimensional n_i features extracted from image or video samples with label i. Collecting all $\mathbf{Y}_i$ together, the training set is formed as $\mathbf{Y} = [\mathbf{Y}_1|\mathbf{Y}_2|\dots|\mathbf{Y}_C] \in \mathbb{R}^{m\times N}$, where $N = \sum_{i=1}^{C} n_i$.

The goal here is the joint achievement of three objectives. The first objective is establishing a representational dictionary with structured as $\mathbf{D} = [\mathbf{D}_1|\mathbf{D}_2|\dots|\mathbf{D}_C] \in \mathbb{R}^{m\times L}$, $(L = KC)$, where atoms have unit ℓ_2-norm and each $\mathbf{D}_i \in \mathbb{R}^{m\times K}$ is a class-specific sub-dictionary independently trained for $\mathbf{Y}_i$. Second, we seek a discriminative scaling matrix $\mathbf{\Lambda} \triangleq \mathrm{diag}([\lambda_1, \dots, \lambda_L]) \in \mathbb{R}^{L\times L}$ such that the i-th scaled atom $\lambda_i\mathbf{d}_i$ in $\mathbf{D\Lambda} \in \mathbb{R}^{m\times L}$ preserves the locality of a neighborhood $\mathbf{\Gamma}_{\lambda_i\mathbf{d_i}} \subset \mathbf{Y}$, which consists of same-class neighboring training samples with respect to atom $\lambda_i\mathbf{d_i}$. We say that $\mathbf{\Lambda}$ is discriminative because there is explicit correspondence between its diagonal elements and the labeled data. And the third objective is realizing a linear mapping $\mathbf{W} \in \mathbb{R}^{C\times L}$ that transforms the reconstruction code $\mathbf{x}_j$ of every feature $\mathbf{y}_j$ to its label vector $\mathbf{h}_j$.

The dictionary learning problem via imposing locality-preserving constraint pair

(DL-LPCP) is thus formalized as:

$$\min_{\mathbf{D},\mathbf{\Lambda},\mathbf{W},\mathbf{X}} \quad \|\mathbf{Y}-\mathbf{D}\mathbf{\Lambda}\mathbf{X}\|_F^2+\alpha\,\|\mathbf{H}-\mathbf{W}\mathbf{X}\|_F^2 \tag{3.13}$$

$$\text{s.t.} \quad \forall i,j$$

$$\mathbf{1}^{\mathrm{T}}\mathbf{x}_j=1,\ \ \|\mathbf{d}_i\|_2=1$$

$$x_{ij}=0 \quad \text{if} \quad \lambda_i\mathbf{d}_i\notin\Omega_{\mathbf{y}_j} \tag{3.13a}$$

$$\lambda_i=\arg\min_{\lambda_i}\sum_{\mathbf{y}_j\in\mathbf{\Gamma}_{\lambda_i\mathbf{d}_i}}|x_{ij}|\,\|\mathbf{y}_j-\lambda_i\mathbf{d}_i\|_2^2 \tag{3.13b}$$

where the 1^{st} and 2^{nd} terms denote the reconstruction error and the label vector approximation error respectively; $\mathbf{X}=[\mathbf{x}_1,\ldots,\mathbf{x}_N]\in\mathbb{R}^{L\times N}$ contains sparse codes for reconstructing $\mathbf{Y}$ with x_{ij} being the i-th element in column $\mathbf{x}_j$; $\mathbf{H}\in\mathbb{R}^{C\times N}$ stores the label information of $\mathbf{Y}$ and its j-th column $\mathbf{h}_j$, is of the form $[0,\ldots,1,\ldots,0]^{\mathrm{T}}\in\mathbb{R}^C$ indicating the label of $\mathbf{y}_j$ by the index of element 1; we define $\Omega_{\mathbf{y}_j}$ as a same-class neighborhood with respect to $\mathbf{y}_j$, which consists of neighboring scaled atoms $\lambda_i\mathbf{d}_i$ of $\mathbf{y}_j$; and let $\mathbf{\Gamma}_{\lambda_i\mathbf{d}_i}=\{\mathbf{y}_j|\ \forall j,\ \ x_{ij}\neq 0,\mathbf{y}_j\in\mathbf{Y}\}$ be the neighborhood set with respect to $\lambda_i\mathbf{d}_i$ containing all the $\mathbf{y}_j$ that are concurrently selecting $\lambda_i\mathbf{d}_i$ as one of their contributing neighbors; the constraint $\mathbf{1}^{\mathrm{T}}\mathbf{x}_j=1$ allows coding to be translation-invariance and $\mathbf{1}$ is a column vector of all ones. Note that $\Omega_{\mathbf{y}_j}$ can be determined by either using ϵ-ball or selecting τ nearest neighbors [47]. In this work, we use the latter method.

We call Eq. (3.13a) and Eq. (3.13b) locality-preserving constraint pair (LPCP), as Eq. (3.13a) enforces supervised local coding (SLC) - each training sample $\mathbf{y}_j$ can only be reconstructed by its same-class neighbors in $\mathbf{D}\mathbf{\Lambda}$, while Eq. (3.13b) correspondingly requires every scaled atom $\lambda_i\mathbf{d}_i$ to be local with respect to those $\mathbf{y}_j$ that are simultaneously choosing it as a contributing neighbor in SLC. Note that the localization errors in Eq. (3.13b) are weighted by the absolute value of coefficients, which adaptively encourages the scaled atom to be more resemble to the training samples that are coded by larger coefficients. This is because when encoding a signal, neighboring atoms are more likely to be coded by larger coefficients than are atoms lying far away [31, 32].

In addition, if we use double indices to express the diagonal elements in $\mathbf{\Lambda}$ and let λ_j^i stand for the element $\lambda_{K(i-1)+j}$ corresponding to the j-th atom in $\mathbf{D}_i$ ($1 \leq j \leq K$ and $1 \leq i \leq C$), then each scaled sub-dictionary can be represented as $\mathbf{D}_i\mathbf{\Lambda}_i$, where $\mathbf{\Lambda}_i = \mathrm{diag}([\lambda_1^i, \ldots, \lambda_K^i]) \in \mathbb{R}^{K\times K}$. Since the local coding is supervised, each sub-dictionary $\mathbf{D}_i$ is representational for the corresponding $\mathbf{Y}_i$ and each discriminatively scaled sub-dictionary $\mathbf{D}_i\mathbf{\Lambda}_i$ is capable of preserving locality with respect to the feature space occupied by $\mathbf{Y}_i$. Therefore, the proposed LPCP enables explicit correspondence between labeled data and the same-class atoms and enhances the coding consistency of similar features.

3.2.3 Optimization

Eq. (3.13) is essentially a bilevel optimization problem [81], where the lower level optimization of the scaling matrix $\mathbf{\Lambda}$ for discriminative locality preservation is embedded into the upper level objective function as a constraint. We minimize Eq. (3.13) by iteratively repeating the following two steps to reduce the objective function, *i.e.*, first solving for $\mathbf{X}$ with all the other variables fixed and then updating $\mathbf{D}$, $\mathbf{\Lambda}$, $\mathbf{W}$ as well as $\mathbf{X}$ jointly while satisfying Eq. (3.13b). Iterations are terminated when stopping criterion met.

3.2.3.1 Supervised Local Coding

In this step, we keep $\mathbf{D}$, $\mathbf{\Lambda}$ and $\mathbf{W}$ fixed and minimize the objective function in Eq. (3.13) with respect to $\mathbf{x}_j \in \mathbf{X}$, for $j = 1, \ldots, N$. We solve the constrained multivariate ridge regression problem [74] as:

$$\begin{aligned} \min_{\mathbf{x}_j} \quad & \left\| \begin{bmatrix} \mathbf{y}_j \\ \sqrt{\alpha}\mathbf{h}_j \end{bmatrix} - \begin{bmatrix} \mathbf{D}\mathbf{\Lambda} \\ \sqrt{\alpha}\mathbf{W} \end{bmatrix} \mathbf{x}_j \right\|_2^2 + \beta \left\| \mathbf{x}_j \right\|_2^2 \\ \text{s.t.} \quad & \mathbf{1}^{\mathrm{T}}\mathbf{x}_j = 1 \\ & x_{ij} = 0 \quad \text{if} \quad \lambda_i \mathbf{d}_i \notin \Omega_{\mathbf{y}_j}, \ \forall i \end{aligned} \tag{3.14}$$

where $\beta \|\mathbf{x}_j\|_2^2$ is the regularization term included for numerical stability with β a small positive constant.

Eq. (3.14) is essentially the locally linear representation problem [47, 73] under the same-class constraint and it can be solved using the approach described in Chapter 2. Specifically, define $\hat{\mathbf{y}}_j = [\mathbf{y}_j^{\mathrm{T}}, \sqrt{\alpha}\mathbf{h}_j^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{m+C}$ as the the j-th column in the augmented training set $\hat{\mathbf{Y}} = [\mathbf{Y}^{\mathrm{T}}, \sqrt{\alpha}\mathbf{H}^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{(m+C)\times N}$ and denote $\hat{\mathbf{D}} = [(\mathbf{D}\mathbf{\Lambda})^{\mathrm{T}}, \sqrt{\alpha}\mathbf{W}^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{(m+C)\times L}$ as the augmented dictionary matrix. Eq. (3.14) is simplified as

$$\min_{\mathbf{x}_j} \|\hat{\mathbf{y}}_j - \hat{\mathbf{D}}\mathbf{x}_j\|_2^2 + \beta \|\mathbf{x}_j\|_2^2 \quad \text{s.t. constraints} \tag{3.15}$$

Define $\hat{\Omega}_{\mathbf{y}_j} = \{\hat{\mathbf{d}}_i | \ \forall i, \ \lambda_i \mathbf{d}_i \in \Omega_{\mathbf{y}_j}, \hat{\mathbf{d}}_i \in \hat{\mathbf{D}}\}$ as the augmented neighborhood with respect to $\mathbf{y}_j$, where $\hat{\mathbf{d}}_i$ is the i-th column in $\hat{\mathbf{D}}$. Let $\tilde{\mathbf{x}}_j$ be the sub-vector containing the τ nonzero elements corresponding to $\hat{\mathbf{d}}_i \in \hat{\Omega}_{\mathbf{y}_j}$. The solution can be efficiently derived by first solving the linear system of equations $(\mathbf{G} + \beta\mathbf{I})\tilde{\mathbf{x}}_j = \mathbf{1}$ and then normalizing $\tilde{\mathbf{x}}_j$ to satisfy the sum-to-one constraint as $\tilde{\mathbf{x}}_j = \tilde{\mathbf{x}}_j / \mathbf{1}^{\mathrm{T}}\tilde{\mathbf{x}}_j$ where $\mathbf{G} = (\hat{\Omega}_{\mathbf{y}_j} - \hat{\mathbf{y}}_j\mathbf{1}^{\mathrm{T}})^{\mathrm{T}}(\hat{\Omega}_{\mathbf{y}_j} - \hat{\mathbf{y}}_j\mathbf{1}^{\mathrm{T}})$ is the local covariance matrix [47] and $\mathbf{I}$ is the identity matrix.

Remarks: The proposed objective function (Eq. (3.13)) has an analytic solution in computing reconstruction coefficients, which is much more efficient than sparse coding (SC). For instance, for encoding one feature, Orthogonal Matching Pursuit (OMP) [14] requires $O(T_0 mL)$ time complexity [14], where T_0 denotes the sparsity priori. The time complexity of SLC is $O(\tau mK) + O(m\tau^3)$, where the first and second terms come from the τ-nearest-neighbor search and the least square solution respectively. In our case, $\tau \leq 3$ and $\tau^2 \ll K$, yielding that the time complexity is dominated by $O(\tau mK)$. Given that $L = KC \gg K$ and that for satisfactory signal recovery, T_0 needs to be sufficiently large (*i.e.*, $T_0 > \tau$), the computational cost of SLC therefore is substantially lower than OMP.

3.2.3.2 Locality-Preserving Dictionary Update

Upon obtaining the reconstruction coefficient matrix $\mathbf{X}$, we continue to minimize Eq. (3.13) by updating $\mathbf{D}$, $\mathbf{\Lambda}$, $\mathbf{W}$ and $\mathbf{X}$ jointly.

Let $\hat{\mathbf{d}}_k \in \mathbb{R}^{(m+C)}$ be the k-th column in the augmented dictionary $\hat{\mathbf{D}}$ and define $\mathbf{x}_{k*} \in \mathbb{R}^{1\times N}$ be the k-th row of the coefficient matrix $\mathbf{X}$. Note that $\hat{\mathbf{d}}_k$ involves three variables, *i.e.,* $\mathbf{d}_k$, λ_k and $\mathbf{w}_k$. We update $\hat{\mathbf{d}}_k$ together with the the nonzero coefficients in $\mathbf{x}_{k*}$, sequentially for $k = 1, \ldots, L$. Specifically, we solve Eq. (3.13) as follows. First, isolate the product $\hat{\mathbf{d}}_k\mathbf{x}_{k*}$ as $\hat{\mathbf{Y}} - \sum_{i\neq k} \hat{\mathbf{d}}_i\mathbf{x}_{i*} - \hat{\mathbf{d}}_k\mathbf{x}_{k*}$ and let $\hat{\mathbf{E}}_k = \hat{\mathbf{Y}} - \sum_{i\neq k} \hat{\mathbf{d}}_i\mathbf{x}_{i*}$. Next, revisiting Eq. (3.13), we have $\mathbf{\Gamma}_{\lambda_k\mathbf{d_k}} = \{\mathbf{y}_j |\ \forall j,\ \ x_{kj} \neq 0, \mathbf{y}_j \in \mathbf{Y}\}$, which keeps track of which $\mathbf{y}_j$ are concurrently selecting $\lambda_k\mathbf{d}_k$ as a contributing neighbor. In other words, $\mathbf{\Gamma}_{\lambda_k\mathbf{d_k}}$ preserves the indices information of the nonzero coefficients in $\mathbf{x}_{k*}$. Denote $[\mathbf{x}_{k*}]_{\mathbf{\Gamma}_{\lambda_k\mathbf{d_k}}} \in \mathbb{R}^{1\times|\mathbf{\Gamma}_{\lambda_k\mathbf{d_k}}|}$ as a subvector of $\mathbf{x}_{k*}$ and $[\hat{\mathbf{E}}_k]_{\mathbf{\Gamma}_{\lambda_k\mathbf{d_k}}} \in \mathbb{R}^{(m+C)\times|\mathbf{\Gamma}_{\lambda_k\mathbf{d_k}}|}$ as a submatrix of $\hat{\mathbf{E}}_k$, consisting of the nonzero coefficients in $\mathbf{x}_{k*}$ and the relevant columns in $\hat{\mathbf{E}}_k$ respectively, all being associated to $\mathbf{\Gamma}_{\lambda_k\mathbf{d_k}}$.

Now we convert Eq. (3.13) to the optimization problem with respect to $\mathbf{d}_k$, λ_k, $\mathbf{w}_k$ and $[\mathbf{x}_{k*}]_{\mathbf{\Gamma}_{\lambda_k\mathbf{d_k}}}$ as

$$\begin{aligned} \min_{\substack{\mathbf{d}_k,\lambda_k,\mathbf{w}_k,\\ [\mathbf{x}_{k*}]_{\mathbf{\Gamma}_{\lambda_k\mathbf{d_k}}}}} & \left\| [\hat{\mathbf{E}}_k]_{\mathbf{\Gamma}_{\lambda_k\mathbf{d_k}}} - \begin{bmatrix} \lambda_k\mathbf{d}_k \\ \sqrt{\alpha}\mathbf{w}_k \end{bmatrix} [\mathbf{x}_{k*}]_{\mathbf{\Gamma}_{\lambda_k\mathbf{d_k}}} \right\|_F^2 \\ \text{s.t.} \quad & \|\mathbf{d}_k\|_2 = 1 \\ & \lambda_k = \arg\min_{\lambda_k} \sum_{\mathbf{y}_j\in\mathbf{\Gamma}_{\lambda_k\mathbf{d_k}}} |x_{kj}| \, \|\mathbf{y}_j - \lambda_k\mathbf{d}_k\|_2^2 \end{aligned} \tag{3.16}$$

Note that we have disregarded constraints $x_{ij} = 0$ if $\lambda_i\mathbf{d}_i \notin \Omega_{\mathbf{y}_j}$ and $\mathbf{1}^{\mathrm{T}}\mathbf{x}_j = 1$, $\forall i, j$. The former one requires supervised local sparsity, which is already satisfied by imposing $\mathbf{\Gamma}_{\lambda_k\mathbf{d_k}}$ onto $\hat{\mathbf{E}}_k$ and $\mathbf{x}_{k*}$. The latter constraint can be easily realized after the updating step by column-wise normalization on each $\mathbf{x}_j \in \mathbf{X}$. In practice, however, there is no need to take extra computation for performing normalization, as on classifying query signals, only $\mathbf{D}$, $\mathbf{\Lambda}$ and $\mathbf{W}$ are of our interest. The solution to Eq. (3.16) is given by

the following proposition.

Proposition 1 (Locality-Preserving Dictionary Update)**.** *Let* $\mathbf{d}_k \in \mathbb{R}^m$, $\lambda_k \in \mathbb{R}$, $\mathbf{w}_k \in \mathbb{R}^C$, $\mathbf{\Gamma}_{\lambda_k \mathbf{d_k}}$, $[\mathbf{x}_{k*}]_{\mathbf{\Gamma}_{\lambda_k \mathbf{d_k}}} \in \mathbb{R}^{1 \times |\mathbf{\Gamma}_{\lambda_k \mathbf{d_k}}|}$ *and* $[\hat{\mathbf{E}}_k]_{\mathbf{\Gamma}_{\lambda_k \mathbf{d_k}}} \in \mathbb{R}^{(m+C) \times |\mathbf{\Gamma}_{\lambda_k \mathbf{d_k}}|}$ *be defined as above. Let* $\delta_t = [\mathbf{I}_{m \times m} \ \mathbf{0}_{m \times C}] \in \mathbb{R}^{m \times (m+C)}$ *and* $\delta_b = [\mathbf{0}_{C \times m} \ \mathbf{I}_{C \times C}] \in \mathbb{R}^{C \times (m+C)}$ *be operators keeping the top* m *and bottom* C *elements of a column vector respectively, where* $\mathbf{0}$ *is a zero matrix. Then the solution, i.e.,* $\mathbf{d}_k^{new}$, λ_k^{new}, $\mathbf{w}_k^{new}$, $[\mathbf{x}_{k*}]^{new}_{\mathbf{\Gamma}_{\lambda_k \mathbf{d_k}}}$ *that minimizes Eq. (3.16) is given as*

$$
\begin{aligned}
\hat{\mathbf{U}} \mathbf{\Delta} \hat{\mathbf{V}}^{\mathrm{T}} &= [\hat{\mathbf{E}}_k]_{\mathbf{\Gamma}_{\lambda_k \mathbf{d_k}}} \\
\mathbf{d}_k^{new} &= \frac{\delta_t \hat{\mathbf{u}}}{\|\delta_t \hat{\mathbf{u}}\|_2} && (3.17) \\
\lambda_k^{new} &= \sum_{\mathbf{y}_j \in \mathbf{\Gamma}_{\lambda_k \mathbf{d_k}}} \frac{|x_{kj}| (\delta_t \hat{\mathbf{u}})^{\mathrm{T}} \mathbf{y}_j}{\|\delta_t \hat{\mathbf{u}}\|_2} \Big/ \sum_{\mathbf{y}_j \in \mathbf{\Gamma}_{\lambda_k \mathbf{d_k}}} |x_{kj}| && (3.18) \\
\mathbf{w}_k^{new} &= \frac{\lambda_k^{new} \delta_b \hat{\mathbf{u}}}{\|\delta_t \hat{\mathbf{u}}\|_2} && (3.19) \\
[\mathbf{x}_{k*}]^{new}_{\mathbf{\Gamma}_{\lambda_k \mathbf{d_k}}} &= \frac{\Delta(1,1) \|\delta_t \hat{\mathbf{u}}\|_2 \hat{\mathbf{v}}^{\mathrm{T}}}{\lambda_k^{new}} && (3.20)
\end{aligned}
$$

where $\hat{\mathbf{u}} \in \mathbb{R}^{m+C}$ *and* $\hat{\mathbf{v}} \in \mathbb{R}^{|\mathbf{\Gamma}_{\lambda_k \mathbf{d_k}}|}$ *are the first columns of* $\hat{\mathbf{U}}$ *and* $\hat{\mathbf{V}}$ *respectively, and* $\Delta(1,1)$ *is the largest singular value in* $\mathbf{\Delta}$*.*

Proof. Applying singular value decomposition (SVD) to $[\hat{\mathbf{E}}_k]_{\mathbf{\Gamma}_{\lambda_k \mathbf{d_k}}} = \hat{\mathbf{U}} \mathbf{\Delta} \hat{\mathbf{V}}^{\mathrm{T}}$ yields the best rank-1 matrix approximation as $\Delta(1,1) \hat{\mathbf{u}} \hat{\mathbf{v}}^{\mathrm{T}}$ [33], where $\hat{\mathbf{u}}$ and $\hat{\mathbf{v}}$ are the first columns of $\hat{\mathbf{U}}$ and $\hat{\mathbf{V}}$, and $\Delta(1,1)$ is the largest singular value in $\mathbf{\Delta}$. Then, use operator $\delta_t = [\mathbf{I}_{m \times m} \ \mathbf{0}_{m \times C}] \in \mathbb{R}^{m \times (m+C)}$ to extract the top m elements ($\delta_t \hat{\mathbf{u}} \in \mathbb{R}^m$) from $\hat{\mathbf{u}}$ for $\mathbf{d}_k$. In order to satisfy the unit ℓ_2-norm constraint, dividing $\mathbf{d}_k$ by $\|\mathbf{d}_k\|_2$ generates $\mathbf{d}_k^{new} = \frac{\delta_t \hat{\mathbf{u}}}{\|\delta_t \hat{\mathbf{u}}\|_2}$.

Once $\mathbf{d}_k$ is updated, λ_k^{new} can be easily obtained by solving $\min_{\lambda_k} \sum_{\mathbf{y}_j \in \mathbf{\Gamma}_{\lambda_k \mathbf{d_k}}} |x_{kj}| \|\mathbf{y}_j - \lambda_k \mathbf{d}_k^{new}\|_2^2$. Note that $\mathbf{\Gamma}_{\lambda_k \mathbf{d_k}}$ is already determined in the last SLC step and does not change through dictionary update. Finally, use $\delta_b = [\mathbf{0}_{C \times m} \ \mathbf{I}_{C \times C}] \in \mathbb{R}^{C \times (m+C)}$ to extract the bottom C elements ($\delta_b \hat{\mathbf{u}} \in \mathbb{R}^C$) from $\hat{\mathbf{u}}$ for $\mathbf{w}_k$. Without breaking the best

rank-1 matrix approximation to $[\hat{\mathbf{E}}_k]_{\mathbf{\Gamma}_{\lambda_k \mathbf{d_k}}}$, simultaneously multiplying $\mathbf{w}_k$ and dividing $\Delta(1,1)\hat{\mathbf{v}}^{\mathrm{T}}$ by scalar $\frac{\lambda_k^{new}}{\|\delta_t \hat{\mathbf{u}}\|_2}$ yields the desired updates $\mathbf{w}_k^{new}$ and $[\mathbf{x}_{k*}]^{new}_{\mathbf{\Gamma}_{\lambda_k \mathbf{d_k}}}$. □

Sequentially applying Proposition 1 to all $\mathbf{d}_i$ for $i = 1, \ldots, L$ finishes the locality-preserving dictionary update in the current step.

Optimizing DL-LPCP iteratively by repeating the above two steps (Supervised Local Coding and Locality-Preserving Dictionary Update), we can jointly obtain dictionary $\mathbf{D}$ together with the discriminative scaling matrix $\mathbf{\Lambda}$ and mapping $\mathbf{W}$. $\mathbf{D}$ is representational with respect to training data $\mathbf{Y}$, because each atom is a unit basis, optimized by achieving the best rank-1 approximation. The scaled dictionary $\mathbf{D\Lambda}$ is capable of preserving locality in the feature space, as λ_k is optimized such that $\lambda_k \mathbf{d}_k$ best represents the neighborhood $\mathbf{\Gamma}_{\lambda_k \mathbf{d_k}}$.

Algorithm 1 DL-LPCP Algorithm

Input: Training set $\mathbf{Y} \in \mathbb{R}^{m \times N}$

Output: Dictionary $\mathbf{D} \in \mathbb{R}^{m \times L}$, discriminative scaling matrix $\mathbf{\Lambda} \in \mathbb{R}^{L \times L}$ and linear mapping $\mathbf{W} \in \mathbb{R}^{C \times L}$

1: Class-specific initialization for each scaled sub-dictionary $\mathbf{D}_i\mathbf{\Lambda}_i$ via K-Means over $\mathbf{Y}_i$, or randomly picking K samples per $\mathbf{Y}_i$.
2: **repeat**
3: **for** $j = 1$ to N **do**
4: Fixing $\mathbf{D}$, $\mathbf{\Lambda}$ and $\mathbf{W}$, computing SLC code $\mathbf{x}_j$ for $\mathbf{y}_j$ according to Eq. (3.14)
5: **end for**
6: **for** $k = 1$ to L **do**
7: Updating $\mathbf{d}_k$, λ_k, $\mathbf{w}_k$ and $[\mathbf{x}_{k*}]_{\mathbf{\Gamma}_{\lambda_k \mathbf{d_k}}}$ following Proposition 1
8: **end for**
9: **until** Convergence (maximum iterations reached or objective function ≤ threshold)

By assuming that the SLC step can always find the most accurate approximations to all training samples, the convergence of DL-LPCP to a local minimum is ensured, as when updating the k-th quadruple $\left\langle \mathbf{d}_k, \lambda_k, \mathbf{w}_k, [\mathbf{x}_{k*}]_{\mathbf{\Gamma}_{\lambda_k \mathbf{d_k}}} \right\rangle$ for all k, DL-LPCP is guaranteed to reduce the objective function value to the extent of best rank-1 matrix approximation [33] to the current residual error matrix. The overall algorithm for training DL-LPCP is summarized in Algorithm 4.

Remarks: The proposed DL-LPCP algorithm has a major advantage of class-independent training, allowing new classes to be easily defined through the inclusion of sub-dictionaries, which is in contrast to many existing algorithms [1, 37–39, 77] requiring memory and computation intensive re-training. Our approach is, therefore, suitable for leave-the-rest-unchanged upgrades in humanity-centric computing systems. This is necessary, for instance, when user style-preferred training samples (e.g., gestures) better fit a user's need or when including user-customized classes (e.g., multiuser face images) are necessary to adapt the system to more complex usages.

3.2.3.3 Classification Strategy

Upon obtaining $\mathbf{D}$, $\mathbf{\Lambda}$ and $\mathbf{W}$, we propose a new classification strategy, harnessing effectively both the representational capability of $\mathbf{D}$ and the locality-preserving characteristic of $\mathbf{D\Lambda}$. Given a query signal $\mathbf{y} \in \mathbb{R}^m$, we solve for the coefficient vector $\mathbf{x} = [x_1, \ldots, x_i, \ldots, x_L]^{\mathrm{T}}$ that minimizes the cost function as

$$\begin{aligned} \min_{\mathbf{x}} \quad & \left\| \frac{\mathbf{y}}{\|\mathbf{y}\|_2} - \sum_{i=1}^{L} x_i \mathbf{d}_i \right\|_2^2 + \gamma \sum_{i=1}^{L} \exp\left(\frac{\|\mathbf{y} - \lambda_i \mathbf{d}_i\|_2}{\sigma} \right) x_i^2 \\ \text{s.t.} \quad & \mathbf{1}^{\mathrm{T}} \mathbf{x} = 1 \end{aligned} \tag{3.21}$$

where the 1^{st} term requires the coding to find the linear combination of unit atoms ($\|\mathbf{d}_i\|_2 = 1$) best representing the direction of $\mathbf{y}$, while the 2^{nd} term encourages the coding to be localized in terms of Euclidean distance by penalizing the coefficients with large weights for atoms far away; γ balances the relative importance between the two terms; σ is included for controlling the decay speed of weight function $\exp\left(\frac{\|\mathbf{y} - \lambda_i \mathbf{d}_i\|_2}{\sigma} \right)$ [32].

Minimizing Eq. (3.21) is essentially a discriminative basis-selection process in which x_i is encoded with large value only if the following two conditions are simultaneously satisfied, *i.e.,* $\mathbf{d}_i$ is representational for $\mathbf{y}$ and $\lambda_i \mathbf{d}_i$ are sufficiently close to $\mathbf{y}$. Note that the proposed classification coding strategy of Eq. (3.21) is different from sparse coding (SC) [1,36–39] and local coding (LC) [31,32] in the fact that we compute

$\mathbf{x}$ exploiting not only the representational dictionary $\mathbf{D}$ but also the locality-preserving dictionary $\mathbf{D\Lambda}$ while SC and LC solve for a reconstruction code using a single representational dictionary. Eq. (3.21) also has a closed-form solution which can be derived in a similar way as solving Eq. (3.14).

To enforce sparsity in $\mathbf{x}$, define index set $\mathbf{S} = \{i \mid \forall i, |x_i| > s \max(\mathbf{x})\}$, where s is the cutting-off ratio and $\max(\mathbf{x})$ denotes the largest element in $\mathbf{x}$. Then we set $x_i = 0$ if $i \notin \mathbf{S}, \forall i$ and update $\mathbf{x}$ by solving a local linear system $[\mathbf{D\Lambda}]_{\mathrm{S}}\tilde{\mathbf{x}} = \mathbf{y}$, s.t. $\mathbf{1}^{\mathrm{T}}\tilde{\mathbf{x}} = 1$ [32], where $\tilde{\mathbf{x}}$ and $[\mathbf{D\Lambda}]_{\mathbf{S}}$ are elements in $\mathbf{x}$ and columns in $\mathbf{D\Lambda}$, corresponding to $\mathbf{S}$. To this end, $\|\mathbf{x}\|_0 = |\mathbf{S}|$.

Finally, we employ $\mathbf{W}$ and compute projection $\mathbf{l} = \mathbf{W}\mathbf{x} \in \mathbb{R}^C$. The label of $\mathbf{y}$ is determined by $i = \arg\max_i(\mathbf{l} = [l_1, \ldots, l_i, \ldots, l_C]^{\mathrm{T}})$.

3.2.4 Experimental Results

In this section, we evaluate DL-LPCP using several benchmark datasets, including the Extended Yale B Database [56], the CMU PIE Database [58], the AR Face Database [82], the Weizmann Action Database [83] and five benchmarks from UCI Machine Learning Archive. The performance of the proposed method is compared with SRC [11], KSVD [33], D-KSVD [1], k-Nearest Neighbor (kNN), and multi-class SVM [77]. Without specific instructions, all results reported are based on our own implementation.

Parameter selection is a challenging task for most sophisticated models. We now explain some simple rules for setting parameters in DL-LPCP. We set $\tau = 2$ consistently for experiments over all datasets since a smaller τ can better preserve locality and reduce computational complexity. We also set $\alpha = 1$ uniformly for all datasets to place equal emphasis on both faithful reconstruction and accurate label approximation. For the Extended Yale B Database and the AR Face Dataset, K is set in accordance with literature [1] while for other datasets we set K empirically without searching for the optimum. In classification coding strategy Eq.(3.21), s is of dominant importance compared to γ and σ, as it directly manipulates the sparsity

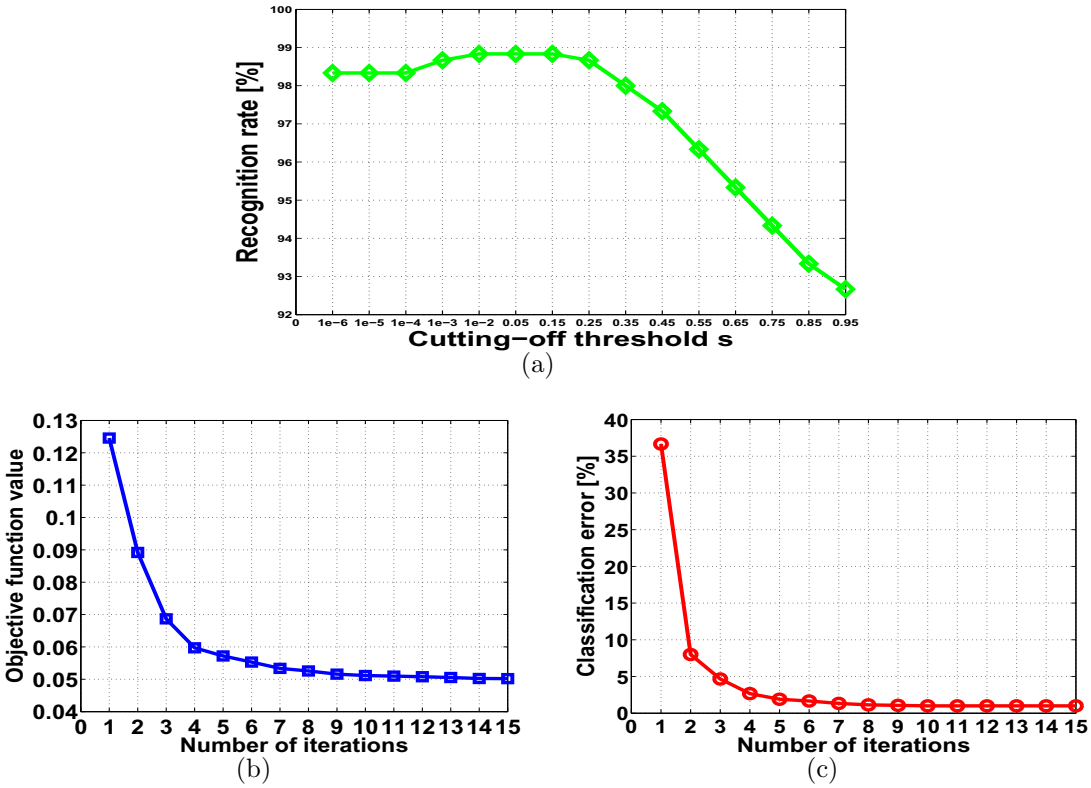


Figure 3.6: (a) Classification performance with respect to s. (b) Objective function value versus iterations; (c) Classification error rate versus iterations.

in the resultant coefficient vector. To reduce searching dimensionality, we manually select $\gamma = 10$ and $\sigma = 1$ and perform 1D grid search, seeking the optimal s based on cross-validation [84]. For example, as shown in Fig. 3.6(a), the proposed DL-LPCP reaches its peak performance on the AR Face Database when $0.01 \leq s \leq 0.15$.

In addition, initialization strategy is sometimes also an important factor affecting the performance of a method. Unlike many recently proposed algorithms [1, 32, 38, 39] that require deliberately conducted initialization process to prevent underfitting/overfitting, the proposed method handles well with random initialization. For proof-of-concept, the learning curves of DL-LPCP with random initialization over the AR Face Database are illustrated in Fig. 3.6(b) and Fig. 3.6(c), from which we can see that DL-LPCP converges quickly and achieves impressive classification performance within only a few iterations.

Evaluation using Extended YaleB Database

We evaluate DL-LPCP over the Extended Yale B Database which contains 2414 frontal face images of 38 subjects [56], *i.e.*, about 64 image per person. This database is captured under varying illumination conditions and expressions. The images are normalized to 32×32 to form 1024D feature vectors for classification. As in [1, 11], we randomly select half of the images (about 32 per person) for training and the other half for testing. In pre-process stage, histogram equalization is performed for DL-LPCP while ℓ_2 normalization is carried out for sparse-coding-based approaches. We evaluate various methods over two random subspaces (with dimension 300D and 504D) and the raw feature space (1024D). For fair comparison, we adopt the optimal parameter settings for D-KSVD and KSVD [1], and set $K = 8$ for all dictionary learning algorithms. For the proposed DL-LPCP, the parameter $s = 0.01$. For D-KSVD and KSVD, the sparsity prior T_0 is set to 16. The experiment is repeated 30 times.

The learned dictionary contains 304 atoms (8 atoms per subject). From Table 3.3, we see that DL-LPCP achieves the best classification performance in all three scenarios. More specifically, as the feature dimension increasing, DL-LPCP yields 98.9%, 99.1% and 99.6% recognition rates, compared to the second best SRC achieving 96.7% and 98.2% on 300D and 504D random feature space respectively. Note that to keep consistent experimental setup as [11], we end up evaluating SRC at 504D and for fair comparison, recognition rates of D-KSVD and KSVD are cited from [1]. We also test the performance of SRC with 8 images per person and the corresponding recognition rate is obviously lower than dictionary learning algorithms. This fact confirms that the learnt dictionary atoms are of much better discriminative ability than image prototypes.

In addition, using 504D feature vectors, we compare the average running time of DL-LPCP with SRC and D-KSVD for classifying one image. Table 3.4 indicates that our method is approximately 12.9 times faster than D-KSVD and 155.9 times faster than SRC.

Evaluation using CMU PIE Database

The CMU PIE database contains 41368 face images of 68 subjects, and for each person there are 13 different poses, 43 different illumination conditions and 4 different expressions. As in [57], we use a subset of the database, *i.e.*, C05, C07, C09, C27, and C29, in which images are nearly frontal poses and are taken under varying conditions of illumination and expression. The subset yields a total number of 11554 images with about 170 images per subject. Following [57], a random selection of $p = (30, 50, 70, 90, 130)$ images per person are employed to form the training set, and the rest of the database are for testing. Classification is performed by using 32×32 cropped images. In preprocess stage, histogram equalization is performed for DL-LPCP while ℓ_2 normalization is carried out for sparse-coding-based approaches. We set the number of atoms per class to $K = 20$ for all dictionary learning methods. The parameter s is set to 0.1 for DL-LPCP. To achieve satisfactory results, sparsity prior T_0 is set to 40 for D-KSVD and KSVD. The experiment is repeated 20 times.

Following [57], we evaluate all methods using recognition error rates. The results are presented in Table 5. Note that the error rates of S-LDA (7-th row) are cited directly from [57] as the state-of-the-art results. The proposed DL-LPCP significantly outperforms the competing approaches in all cases $p = 30, 50, 70, 90, 130$.

Evaluation using AR Face Database

Table 3.3: Recognition results over the Extended YaleB Database. Note for D-KSVD and KSVD, recognition rates are cited from [1].

Dimension	300D	504D	1024D
SRC	96.7%	98.2%	N/A
$SRC^{\star}$ (8)	79.0%	80.2%	84.5%
D-KSVD [1]	N/A	95.6%	N/A
KSVD [1]	N/A	93.2%	N/A
SVM	92.3%	95.6%	97.4%
kNN	77.8%	88.1%	88.6%
DL-LPCP	**98.9**%	**99.1**%	**99.6**%

Table 3.4: Comparison of running time (ms) for classifying a test image.

	DL-LPCP	D-KSVD [1]	SRC
Running time	6.52ms	84.00ms	1016.61ms

Table 3.5: Error rates over the CMU PIE Database for various methods with different sizes training set.

Training samples	30	50	70	90	130
SRC	5.8%	4.2%	3.2%	3.0%	2.3%
D-KSVD	7.9%	6.1%	4.5%	4.3%	3.5%
KSVD	6.7%	5.9%	4.6%	4.5%	4.1%
SVM	9.2%	5.3%	3.9%	3.2%	2.4%
kNN	17.5%	15.0%	10.8%	8.5%	7.6%
S-LDA [57]	3.6%	2.5%	2.1%	1.8%	1.6%
DL-LPCP	**2.4**%	**1.7**%	**1.2**%	**1.0**%	**0.9**%

The AR Face Database consists of over 4, 000 color images of 126 persons. Each individual has 26 face images taken during two separate sessions. As in [11], we also choose a subset consisting of 50 male individuals and 50 female individuals. For each person, 14 images with only variations in illumination conditions and expressions are collected, with 7 images from session 1 and the other 7 images from session 2. This yields a total number of 1400 images. Each face image of size 165-by-120 pixels, is projected onto a 540D random feature space via a randomly generated matrix. As previous two databases, image pre-processing techniques are applied to various methods. We employ different number of training samples per class as $n = 7, 8, 9, 10, 11$ and train correspondingly $K = 5, 6, 7, 8, 9$ atoms per class for all dictionary learning algorithms. For DL-LPCP, the parameter $s = 0.05$. For D-KSVD and KSVD, T_0 is set to 10 [1]. Experiment is repeated 20 times for each case.

Fig. 3.7 shows that our method maintains a high recognition rate and outperforms other approaches consistently in all different settings. More specifically, DL-LPCP achieves accuracies between 96.1% and 98.7%, which correspond to the case $n = 7$ (K = 5) and the case $n = 11$ (K = 9) respectively, compared to SRC yielding 94.6% and 98.3% in such two cases. Note that in face recognition, our classification strategy is related to the collaborative representation method [85] and therefore to ensure faithful reconstruction, parameter s for face datasets is set relatively smaller so as to include more representational bases, compared to the one for human-action recognition dataset (see next section).

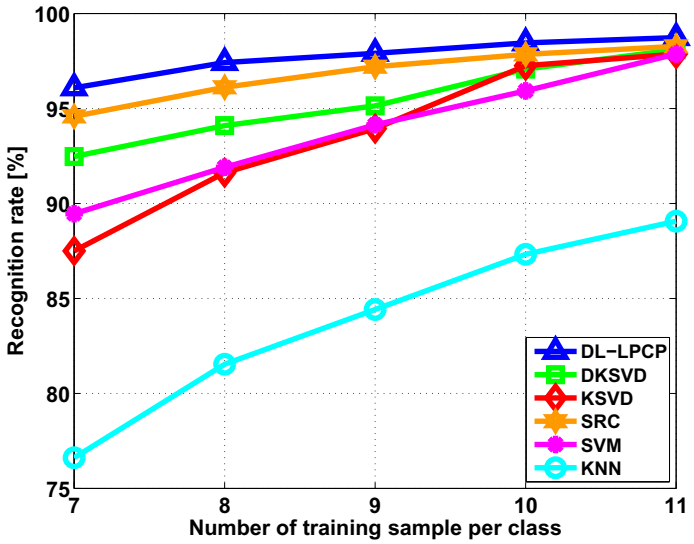
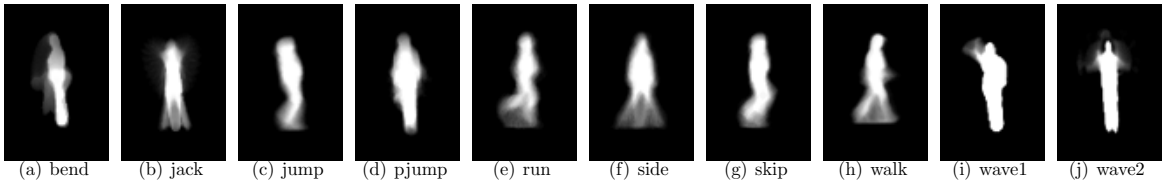


Figure 3.7: Recognition results over the AR Face Database.

Evaluation using Weizmann Action Database

The Weizmann Action Database [86] contains 90 videos of 9 different individuals. Each person performed 10 natural actions, *i.e.*, bend, jumping jack (jack), jump forward (jump), jump in place (pjump), run, gallop sideways (side), skip, walk, wave one hand (wave1) and wave both hands (wave2). As this database is captured by a fixed camera under static background, a simple background subtraction and normalized cross-correlation based registration strategy could align human figures very well.

Obeying the same evaluation protocol as in [83], we perform leave-one-person-out experiments to compare various methods. We utilize Motion History Image (MHI) [87] to transform each aligned training sequence into two silhouette images by averaging the odd-numbered and even-numbered frames respectively, which thus yields a total number of 160 training images. The test set is generated by computing one MHI for each query sequence. Example MHIs of actions are illustrated in Fig.3.8. Finally, all the samples are mapped onto a subspace with dimension $m = 38$ by PCA for classification. We set $K = 10$ for all dictionary learning algorithms. For DL-LPCP, $s = 0.3$. For D-KSVD and KSVD, $T_0 = 12$. As listed in Table 3.6, DL-LPCP achieves the highest recognition rate 100.0% among all the competing algorithms. The basic reason for the



(a) bend (b) jack (c) jump (d) pjump (e) run (f) side (g) skip (h) walk (i) wave1 (j) wave2

Figure 3.8: Example MHIs of 10 natural actions.

Table 3.6: Recognition results over the Weizmann Action Database.

Methods	DL-LPCP	SRC	D-KSVD	KSVD	SVM	kNN
Accuracy	**100.0**%	94.5%	93.3%	87.8%	96.7%	93.3%

good performance of DL-LPCP over these datasets is two-fold: 1) the novel locality-preserving constraint pair encourages same-class features to share similar atoms by ensuring training samples to be locally best represented by same-class atoms; 2) the new classification strategy poses a rigorous bases selection criterion allowing only atoms that are simultaneously representational and local with respect to the query sample to be selected.

Evaluation using UCI Machine Learning Datasets

Finally, we evaluate DL-LPCP using five datasets from UCI Machine Learning Archive, namely, Iris, Satellite, Segmentation, Letter and Vehicle datasets. Their basic information are listed in Table 3.7. In the previously reported literature, various combinations of feature extraction, dimensionality reduction and classifier have been applied over the selected datasets. For fair comparison, we directly employ raw data samples for classification without performing any preprocess or feature extraction. Note that the dictionary learning algorithms, KSVD and D-KSVD, have the same settings as the proposed method for the parameters α and K, while SRC uses all the available training samples for recognition. The classification results are based on 10-fold cross-validation with 30 repetitions, and, as shown in Table 3.8, DL-LPCP leads in performance throughout the five UCI datasets.

Remark: We compare the methods by only extracting information of pedal length and pedal width. Such 2D features are sufficiently informative for kNN and SVM, and are

Table 3.7: Basic information about Iris, Satellite, Segmentation, Letter and Vehicle datasets from UCI Machine Learning Archive.

Dataset	Total samples	Dimensions	Classes
Iris	150	4	3
Satellite	6435	36	6
Segmentation	2310	19	7
Letter	20000	16	26
Vehicle	946	18	4

Table 3.8: Classification accuracy over the UCI Machine Learning data sets. The 3rd column contains the results obtained by keeping only two dimensions of information, *i.e.*, pedal length and pedal width.

	Iris	Iris (l vs. w)	Satellite	Letter	Segmentation	Vehicle
SRC	94.2%	77.3%	70.8%	95.6%	93.6%	72.1%
D-KSVD	83.1%	57.3%	74.2%	92.7%	91.8%	64.8%
KSVD	95.7%	78.2%	73.0%	90.0%	87.8%	73.3%
SVM	96.0%	93.1%	88.7%	92.9%	94.8%	70.8%
kNN	95.1%	95.3%	88.6%	94.9%	92.4%	72.0%
DL-LPCP	**97.8**%	**97.1**%	**90.7**%	**95.8**%	**96.9**%	**82.6**%

usually employed for visualization[2]. We find that sparse-coding based classification schemes suffer a significant drop in accuracy (3rd column in Table 3.8). This is due to the fact that three classes of data are approximately distributed along the same radius direction and so are the learned dictionary atoms. Without proper constraints, sparse coding algorithms *i.e.,* SRC, KSVD, D-KSVD, would lose their discrimination ability over such type of data. Our observation coincides with [88].

3.3 2013 IEEE GRSS Data Fusion Contest on Hyperspectral Image Classification

In this section, we present some exciting outcomes from the participation in 2013 IEEE GRSS Data Fusion Contest on Hyperspectral Image Classification. Under the guidance of Prof. Arce and Prof. Barner, I performed actively as team leader and developed a highly efficient algorithm for data fusion and classification, based on the

[2] For details, see http://en.wikipedia.org/wiki/Iris-flower-data-set

Figure 3.9: Illustration of the hyperspectral and LiDAR imaging over University of Houston. Image courtesy to IEEE GRSS Committee.

previously introduced LCDL algorithm. Our team members are Ana Remirez, Luisa Polania and Sherin Mathews.

3.3.1 Introduction

The Data Fusion Contest is annually organized by the Data Fusion Technical Committee of the Geoscience and Remote Sensing Society (GRSS). The 2013 Contest aims at exploring the synergetic use of hyperspectral and LiDAR data. The hyperspectral image cube contains 144 spectral bands from 380 to 1050 nm. A co-registered LiDAR is also provided, which is derived using Digital Surface Model (DSM) to characterize elevation information. Both datasets have the same spatial resolution (2.5 m) [89]. As shown in Fig. 3.9 the data is captured during the summer of 2012 over the University of Houston (UH) and the neighboring urban area. The data pre-processing is conducted by student volunteers at UHs Hyperspectral Image Analysis group, and NCALM staff. A ground truth is created by the contest organizing committee via photo-interpretation [89].

In the classification challenge, participants were asked to categorize each image pixel into one of the 14 classes of interest, including distinct types of vegetation, soil, water, but also less common objects, such as commercial buildings, highways, railway, and vehicles (see Fig. 3.10 for the 14 classes information). Among a total number of $349 \times 1905 = 664845$ pixels, only 2832 labeled pixels were available for training and all

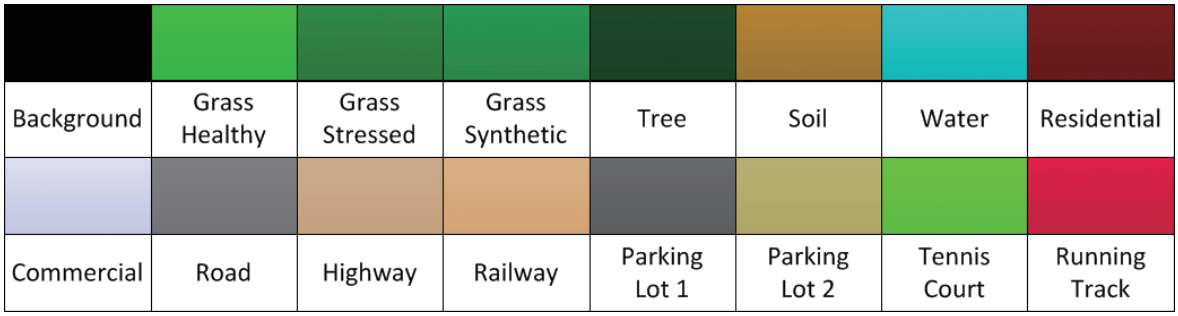


Figure 3.10: Contest legend. Image courtesy to IEEE GRSS Committee.

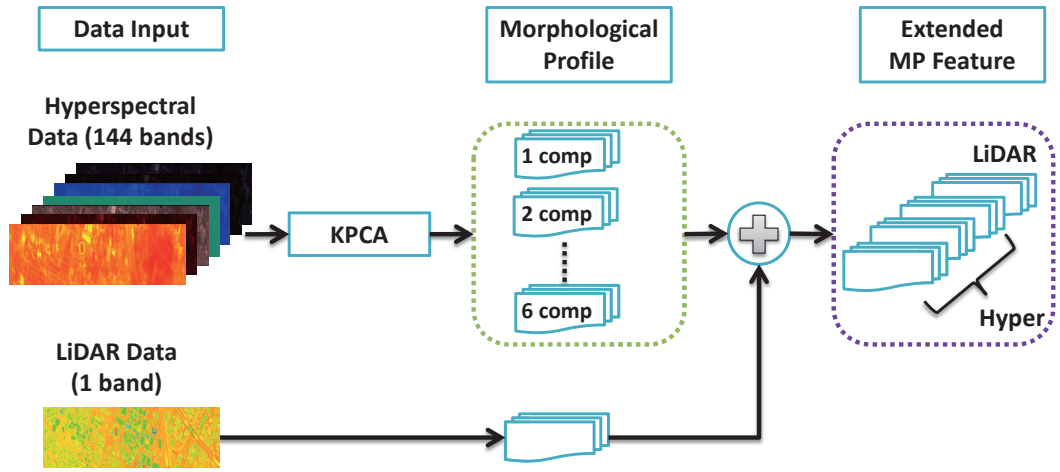


Figure 3.11: The proposed data fusion pipeline. Image courtesy to IEEE GRSS Committee.

the remaining 662013 pixels were for testing. The validation samples that the Contest organizers used to evaluate the submissions were not disclosed.

3.3.2 Data Fusion and Classification Algorithm

As shown in Fig. 3.11, the hyperspectral data cube is first compressed into 6 components via Kernel Principle Component Analysis (KPCA) and then a series of morphological operations are performed on each component and the LiDAR data, yielding 7 Morphological Profiles (MP). Next, the Extended Morphological Profile (EMP) is obtained by concatenating all the MPs [90], which finishes the data fusion step. The final discriminative features are obtained by projecting the EMP into some low-dimensional feature space via Linear Discriminant Analysis (LDA).

Given that observable objects from the outer space are approximately homogeneously textured in the Hyperspectral image cube and consistently characterized in terms of elevation by LiDAR image. Nearby pixels are more likely to be associated to the same class label as the center target pixel. In other words, exploiting the local neighborhood of each pixel may potentially produce more discriminative information for classification. Therefore, we propose a novel discriminative dictionary learning algorithm by extending our previous work the Locality-Constrained Dictionary Learning. Specifically, we suppose the feature of every pixel resides on some latent intrinsic nonlinear manifold, which typically is of much lower dimensionality than the actually captured data in observation space. We have shown in theory that the approximation to an unobservable intrinsic manifold by a few latent landmark points residing on the manifold can be cast in a dictionary learning problem over the observation space. By incorporating the classification error penalty term to form a unified object function, the dictionary and the classifier are jointly learned using the training features provided. The algorithm converges quickly, typically within 15 iterations. The testing data is classified by first computing the locally linear reconstruction code over the dictionary and then the code is mapped to a label vector via the classifier.

Classification of high-resolution satellite image is a challenging task in remote sensing, since the image cube usually contains $\sim 10^6$ pixels, each with tens to hundreds of feature dimensions. Despite their great success, training traditional classification algorithms, *e.g.,* nonlinear SVM, is of quadratic complexity in the number of training samples, which requires exorbitant computational complexity and memory usage. In contrast, training our newly proposed algorithm has only linear complexity with respect to the number of samples, which allows orders of magnitude speedup when dealing with large-scale dataset. In our pilot study, classifying 662013 pixels takes no more than 3 minutes with MATLAB implementation.

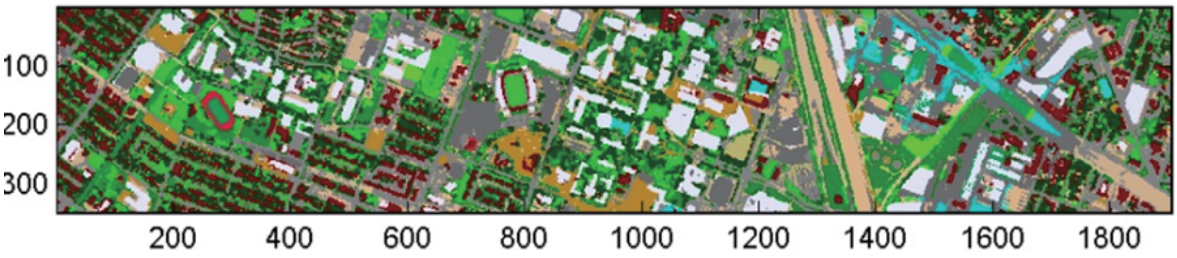


Figure 3.12: Classification result. The label of each pixel is represented with different color. Image courtesy to IEEE GRSS Committee.

3.3.3 Results

This year's contest received more than 900 registrations from different academic institutions and corporations across 69 countries. Among the 50+ teams which are mostly composed of researchers specialized in satellite imaging or remote sensing, our UD SPC team was able to achieve the global 9th place. It is also worthy to point out that our team is the only US team within top 10. The resulting classification label map is shown in Fig. 3.12. The encouraging result confirms the effectiveness of locality-constrained dictionary learning as a useful machine learning algorithm in hyperspectral imaging classification.

After comparing the performances between ours and other better ranked teams, the major drawback of our approach lies in the fact that we did not include an appropriate preprocessing technique to recover shadow-covered areas. From this fact, we also understand that in order to achieve the best performance in real-world problems, task-specific domain knowledge sometimes is equally important if not more as a sophisticated machine learning algorithm. Therefore, integration an automatic shadow/fog detection and removing algorithm as processing operation will be and is our on-going work.

3.4 Conclusion

In this chapter, we apply the concept of locality-constrained coding and dictionary learning to various computer vision and pattern recognition tasks, including 3D

shape recognition, face recognition, action recognition, data classification, hyperspectral image classification, etc. Specifically, we present three discriminative dictionary learning algorithms by imposing properly designed locality constraints. Experimental results demonstrate that compared to existing sparse-coding based dictionary learning algorithms, our approaches possess three merits: yielding more meaning dictionary atoms with locality-preserving property; highly efficient in training and testing; achieving very competitive performance. We believe that our algorithms hold promise in learning discriminative dictionaries for real-world classification problems.

Chapter 4

AUTOMATIC FEATURE LEARNING FOR BIOMEDICAL IMAGE ANALYSIS

4.1 Introduction

Histology sections contain significant information about the tissue architecture. Automated analysis of tissue histology sections can potentially help predict the clinical outcomes. Hematoxylin and eosin (H&E) are two commonly used histological stains, which respectively label DNA (*e.g.,* nuclei) and protein contents, with various color shades. Abberation in the histology architecture is often seen as an indicator of disease progression. The abberation indices enable the prediction of clinical outcomes *e.g.,* survival, response to therapy. Therefore an effective quantization of these indices is very much desired. However, as an essential ground on which outcome-based analysis is established, large cohorts usually contain large technical variations and biological heterogeneities, which greatly undermines the performance of existing techniques [91, 92].

To solve such problems, several researchers [91, 93–95] have proposed to design and fine tune the human engineered features. These approaches are usually task-specific, which limits their cross-domain applicability. Not until recently has the potential of unsupervised feature learning been exploited in tissue classification [92, 96]. Inspired by previous efforts in unsupervised feature learning, we first present a model called stacked predictive sparse decomposition (PSD^{n}SPM), which is based on traditional sparse coding. Then, we analyze its drawbacks and further introduce a more advanced model called, multispectral convolutional sparse coding (MCSCSPM).

The building block of the first model, *i.e.,* PSD^{n}SPM, is the predictive sparse decomposition (PSD) [22], which incorporates a nonlinear predictor into the traditional

Figure 4.1: Computed basis functions from the Glioblastoma Multiforme (GBM) dataset.

sparse coding objective, for the purpose of efficiently predicting the sparse coefficient vector and avoiding the time consuming optimization process. Stacking multiple layers of PSD, the model (PSD^{n}SPM) can capture higher-level sparse tissue morphometric features. The work of PSD^{n}SPM is a pioneering work in applying deep learning to tissue image classification with encouraging results achieved. For example, the dictionary trained over GBM dataset is shown in Figure 4.1. Yet, the underlying feature learning module of PSD^{n}SPM is sparse coding, which suffers two major drawbacks, *i.e.,* 1) yielding only Gabor-like low-level feature detectors (filters), and 2) having high redundance in the feature representation.

For the reasons listed above, we further propose a multispectral unsupervised feature learning model for tissue classification (MCSCSPM), based on convolutional

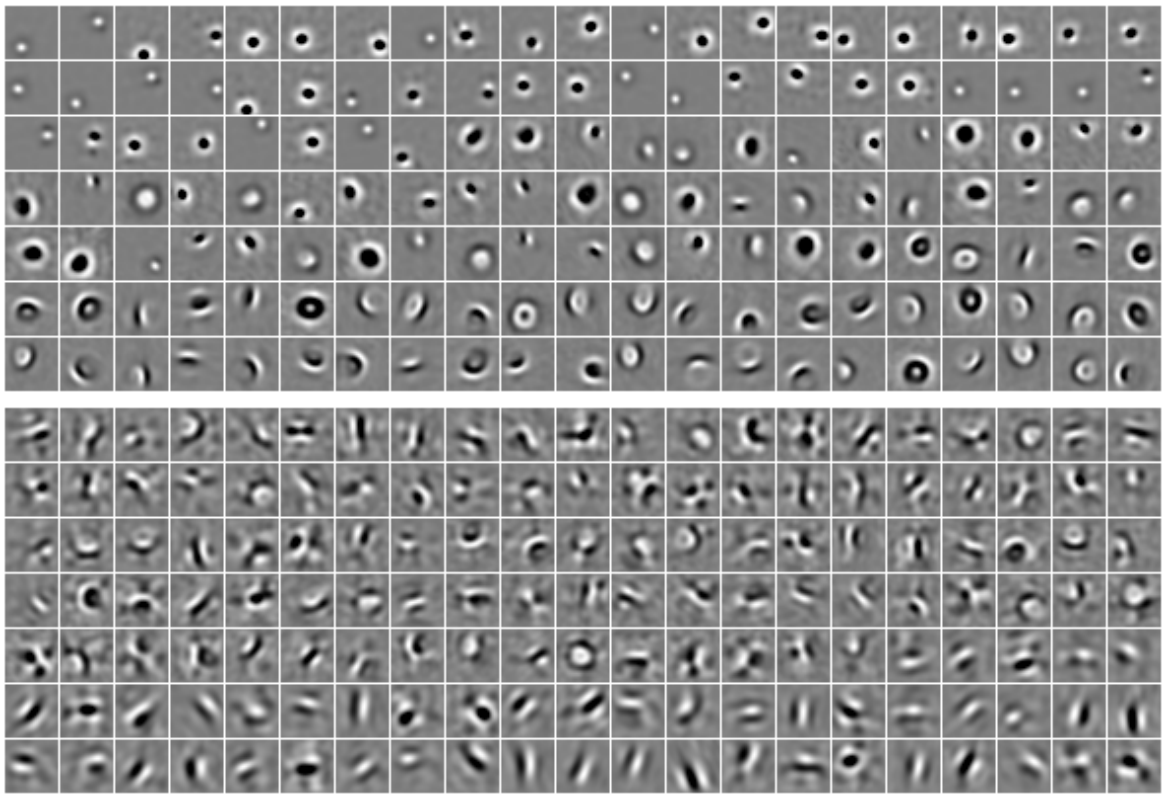


Figure 4.2: 27×27 multispectral filters learned from the GBM dataset. It can be seen that, learned from the nuclear channel, the filters (top figure) capture nuclear regions of distinct shapes; learned from the collagen channel, the filters (bottom figure) characterize the structural connectivity within various tissue sections.

sparse coding (CSC) [25] and spatial pyramid matching (SPM) [97]. The multispectral features are learned in an unsupervised manner through CSC, followed by the summarization through SPM at various scales and locations. Eventually, the image-level tissue representation is fed into linear SVM for efficient classification [98]. Compared with sparse coding, CSC possesses two merits 1) invariance to translation; and 2) producing more complex filters, which contribute to more succinct feature representations. Meanwhile, the proposed approach also benefits from 1) the biomedical intuitions that different color spectrums typically characterize distinct structures; and 2) the utilization of context, provided by SPM, which is important in diagnosis. In short, our work (MCSCSPM) is the first attempt using convolutional sparse coding for tissue classification and achieves superior performance compared to patch-based sparse feature

learning algorithms, *e.g.,* PSDSPM [92]. Moreover, MCSCSPM is capable of generating very competitive results compared to systems built upon biological prior knowledge, *e.g.,* SMLSPM [91]. Finally, our study further indicates that learning features over multiple spectrums can potentially generate biological-component-specific filters. For example, the filters learned from the nuclear channel and collagen channel respectively capture various nuclear regions and the structural connectivity within tissue sections (Figure 4.2).

Tumor histopathology reflects the interaction of underlying molecular defects and environmental factors. The quantification of morphological features and organization, from cell-by-cell analysis of histology sections, can potentially provide a new approach for characterizing and identifying molecular markers of heterogeneity. Large-scale quantitative characterization of tumor morphology from standard hematoxylin and eosin (H&E) stained tissue sections can offer alternative views, as opposed to genome-wide array data, for subtyping and survival analysis. A particular endpoint is that the computed morphometric indices can be tested against outcome. Simultaneously, derived representations (*e.g.,* meta-features), from cell-by-cell analysis, can also be leveraged to probe for heterogeneity and its underlying molecular basis. Tumor heterogeneity can reveal tumor plasticity (*e.g.,* adaptation to environmental factors), potential peripheral molecular drivers, and drug resistivity.

Nuclei segmentation allows accurate delineation of cellular properties and provides insights on characterizing tumor histopathology. Most existing methods are based on human-designed features and their effectiveness can be largely affected by the aforementioned variabilities among data samples. In this dissertation, we propose a novel approach, called sparsity constrained convolutional regression (SCCR), for accurate nuclei segmentation, in the hope of overcoming the difficulties suffered by traditional methods [99–104]. Given raw image patches and the corresponding binary masks, SCCR jointly learns a convolutional filter bank and a linear mapping with sparsity constraint. The filter bank is a set of specialized feature detectors and is employed

to extract pixel-wise feature vectors. The convolutional regression prediction is computed as the inner product between the feature vector and the linear mapping. By feeding the prediction score into a simple decision function, the pixel label can be determined. Comapred to traditional CNN-based models [27,105,106] for image labeling, our method seeks to accurately classify each pixel into nuclear region or background.

4.2 Related Work

In literature, there are several excellent reviews for the analysis of H&E stained sections [107, 108]. Generally speaking, efforts in histology section analysis can be divided into four different directions: 1) Some researchers [101, 109–111] advocated nuclear segmentation and organization for tumor grading and/or the prediction of tumor recurrence; 2) Some groups [93,94,112] focused on patch level analysis (*e.g.,* small regions), using color and texture features, for tumor representation; 3) Some other studies [113, 114] had been conducted on block-level analysis to distinguish different states of tissue development using cell-graph representation; 4) There was also a research branch [103] suggesting detection and representation of the auto-immune response as a prognostic tool for cancer.

Automated biomedical image analysis is a challenging task due to the presence of significant technical variations and biological heterogeneities in the data [91,95], which typically results in techniques that are tumor type specific. In tissue classification, recent studies have focused on either fine tuning human engineered features [93–95], or applying automatic feature learning [92, 96], for robust representation.

In nuclei segmentation, researchers have made a significant amount of effort by introducing techniques from image processing, computer vision and machine learning. Some representative approaches are fuzzy clustering [99], adaptive thresholding followed by morphological filtering [100], hybrid color and texture analysis followed by learning and unsupervised clustering [101], color separation followed by optimum thresholding and learning [102], level set method combining gradient information [103], graph cut method based on seed detection [104]. Color decomposition is a common

preprocessing technique to accentuate the nuclear dye. Thresholding and clustering are based on the assumption that all nuclear regions in the image have consistent chromatin content, which in practice, however, does not hold, due to the following reasons: 1) different cell type and cell state may cause significant variations in chromatin content; 2) the overlapping and clumping of cells may cause distortion to the underlying chromatin content. In addition, aforementioned methods are usually applied to a small dataset collected from a single laboratory and therefore their capability of overcoming technical variations is limited.

In the context of computer vision research on image categorization, the traditional bag of features (BoF) model has been widely studied and improved through different variations [97,115–118], among which SPM [97] has clearly become the major component of the state-of-art systems [119] for its effectiveness in practice.

The evolution of patch-based histology analysis has been SIFT-like feature extraction followed by a evaluation of several kernel-based classification policies [112]; independent subspace analysis that utilizes unsupervised learning without the constraint of being able to reconstruct the original signal [120]; a single layer predictive sparse coding with SVM classifier [121]; and more recently, coupling of either prior knowledge [91] or predictive sparse coding [92] with with spatial pyramid matching. Nevertheless, sparse coding based models, suffer two major drawbacks, *i.e.,* 1) yielding only Gabor-like low-level feature detectors (filters), and 2) having high redundance in the feature representation.

In recent years, convolutional sparse coding has received increasing research interest in computer vision and machine learning communities [25–27,43–45], mainly due to its capability of learning shift-invariant filters with complex patterns. Kavukcuoglu *et al.* [25] proposed to improve the feature extraction efficiency by jointly learning a feed-forward encoder with the convolutional filter bank, and applied the algorithm to Convolutional Networks (ConvNets) achieving impressive results on object recognition. Zeiler *et al.* [26] developed an approach, called Deconvolutional Networks, learning top-bottom feature hierarchies to reconstruct the original image. [27] further extended

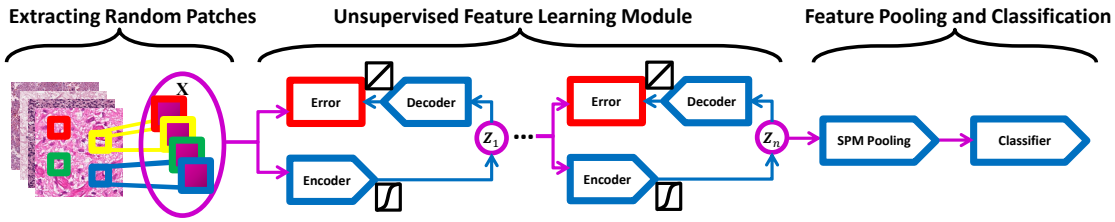


Figure 4.3: Computational workflow of our approach (PSD^{n}SPM).

the work [26] by incorporating a set of latent switch variables and max-pooling, which allows unified training of multiple layers. Bristow *et al.* [43] came up with an efficient method for convolutional sparse coding in Fourier domain, using the Alternating Direction Method of Multipliers approach. In addition to object recognition, convolutional sparse coding has also achieved state-of-the-art performances in pedestrian detection [44], retinal blood vessels segmentation [46], and image denoising [45], etc.

4.3 The PSDSPM Algorithm for Tissue Classification

In this work (PSD^{n}SPM), we employ predictive sparse decomposition (PSD) [22] as a building block for the purpose of constructing hierarchical learning framework, which can capture higher-level sparse tissue morphometric features [122]. Unlike many unsupervised feature learning algorithms [123–126], the feed-forward feature inference of PSD is very efficient, as it involves only element-wise nonlinearity and matrix multiplication. For classification, the predicted sparse features are used in a similar fashion as SIFT features in the traditional framework of SPM, as shown in Figure 4.4.

4.3.1 Unsupervised Feature Learning

Given $\mathbf{X} = [\mathbf{x}_1, ..., \mathbf{x}_N] \in \mathbb{R}^{m \times N}$ as a set of vectorized image patches, we formulate the PSD optimization problem as:

$$\begin{aligned} \min_{\mathbf{B},\mathbf{Z},\mathbf{G},\mathbf{W}} \quad & \|\mathbf{X} - \mathbf{B}\mathbf{Z}\|_F^2 + \lambda\|\mathbf{Z}\|_1 + \|\mathbf{Z} - \mathbf{G}\sigma(\mathbf{W}\mathbf{X})\|_F^2 \\ \text{s.t.} \quad & \|\mathbf{b}_i\|_2^2 = 1, \forall i = 1, \ldots, h \end{aligned} \tag{4.1}$$

where $\mathbf{B} = [\mathbf{b}_1, ..., \mathbf{b}_h] \in \mathbb{R}^{m \times h}$ is a set of the basis functions; $\mathbf{Z} = [\mathbf{z}_1, ..., \mathbf{z}_N] \in \mathbb{R}^{h \times N}$ is the sparse feature matrix; $\mathbf{W} \in \mathbb{R}^{h \times m}$ is the auto-encoder; $\mathbf{G} = \text{diag}(g_1, \ldots, g_h) \in \mathbb{R}^{h \times h}$ is a scaling matrix with diag being an operator aligning vector, $[g_1, \ldots, g_h]$, along the diagonal; $\sigma(\cdot)$ is the element-wise sigmoid function; and λ is a regularization constant. Joint minimization of Eq. (4.1) with respect to the quadruple $\langle \mathbf{B}, \mathbf{Z}, \mathbf{G}, \mathbf{W} \rangle$, enforces the inference of the nonlinear regressor $\mathbf{G}\sigma(\mathbf{WX})$ to be similar to the optimal sparse codes, $\mathbf{Z}$, which can reconstruct $\mathbf{X}$ over $\mathbf{B}$ [22].

As shown below, optimization of Eq. (4.1) is iterative, where the algorithm terminates when either the objective function is below a preset threshold or the maximum number of iterations has been reached.

1. Randomly initialize $\mathbf{B}$, $\mathbf{W}$, and $\mathbf{G}$.

2. Fixing $\mathbf{B}$, $\mathbf{W}$ and $\mathbf{G}$, minimize Eq. (4.1) with respect to $\mathbf{Z}$, where $\mathbf{Z}$ can be either solved as a ℓ_1-minimization problem [123] or equivalently solved by greedy algorithms, e.g., Orthogonal Matching Pursuit (OMP) [127].

3. Fixing $\mathbf{B}$, $\mathbf{W}$ and $\mathbf{Z}$, solve for $\mathbf{G}$, which is a simple least-square problem with analytic solution.

4. Fixing $\mathbf{Z}$ and $\mathbf{G}$, update $\mathbf{B}$ and $\mathbf{W}$, respectively, using the stochastic gradient descent algorithm.

5. Repeat [2]-[4] until stopping condition is satisfied.

In large-scale feature learning problems, involving $\sim 10^5$ image patches, it is computationally intensive to evaluate the sum-gradient over the entire training set. However, both stochastic gradient descent algorithm and GPU parallel computing can provide a significant increase in speed. The former approximates the true gradient of the objective function by the gradient evaluated over mini-batches, and the latter further accelerates the process (up to 5X) with our Matlab implementation based on an Nvidia GTX 580 graphics card. Figure 4.1 illustrates 1024 basis functions computed from the

GBM dataset, which capture both color and texture information from the data and is generally difficult to realize using hand-engineered features.

4.3.2 Spatial Pyramid Matching (SPM)

Having computed the sparse features, $\mathbf{Z} \in \mathbb{R}^{h \times N}$ (e.g., predictions by the non-linear regressor $\mathbf{G}\sigma(\mathbf{WX})$), we then construct a code book and proceed with SPM pooling.

The codebook, $\mathbf{D} = [\mathbf{d}_1, ..., \mathbf{d}_K] \in \mathbb{R}^{h \times K}$, consisting of K sparse tissue morphometric types, is constructed by solving the following optimization problem:

$$\begin{aligned} \min_{\mathbf{D},\mathbf{C}} \quad & \sum_{i=1}^{N} \|\mathbf{z}_i - \mathbf{D}\mathbf{c}_i\|^2 \\ \text{s.t.} \quad & card(\mathbf{c}_i) = 1, \|\mathbf{c}_i\|_1 = 1, \mathbf{c}_i \succeq 0, \forall i \end{aligned} \tag{4.2}$$

where $\mathbf{C} = [\mathbf{c}_1, ..., \mathbf{c}_N] \in \mathbb{R}^{K \times N}$ is the code matrix assigning each $\mathbf{z}_i$ to its closest sparse tissue morphometric type in $\mathbf{D}$, $card(\mathbf{c}_i)$ is a cardinality constraint enforcing only one nonzero element in $\mathbf{c}_i$, and $\mathbf{c}_i \succeq 0$ is a non-negative constraint on all vector elements. Eq. (4.2) is optimized by alternating between the two variables, *i.e.,* minimizing one while keeping the other fixed. After training, $\mathbf{D}$ is fixed and the query signal set, $\mathbf{Z}$, is encoded by solving Eq. (4.2) with respect to $\mathbf{C}$ only.

The next step is to construct a spatial histogram for SPM [97]. By repeatedly subdividing an image, histograms of different sparse tissue morphometric types over the resulting subregions are computed. The spatial histogram, H, is then formed by concatenating the appropriately weighted histograms of sparse tissue morphometric types at all resolutions, *i.e.,*

$$\begin{aligned} H_0 &= H_0^0 \\ H_l &= (H_l^1, ..., H_l^{4^l}), 1 \leq l \leq L \\ H &= (\frac{1}{2^L}H_0, \frac{1}{2^L}H_1, ..., \frac{1}{2^{L-l+1}}H_l, ..., \frac{1}{2}H_L) \end{aligned} \tag{4.3}$$

where $(\cdot)$ denotes the vector concatenation operator, $l \in \{0, ..., L\}$ is the resolution level of the image pyramid, and H_l represents the concatenation of histograms for all image subregions at pyramid level l. Instead of using kernel SVM, we employ the homogeneous kernel map [128] and linear SVM [98] for improved efficiency.

4.4 The Multispectral CSC Algorithm for Tissue Classification

In this work, we adopt convolutional sparse coding (CSC) [25] as the fundamental module for learning filter banks, based on which the proposed multispectral unsupervised feature learning system (MCSCSPM) is constructed. As noted by several researchers [25, 43], sparse coding typically assumes that training image patches are independent from each other, and thus neglects the spatial correlation among them. In practice, however, this assumption typically leads to filters that are simply translated versions of each other, and, as a result, generates highly redundant feature representation. While, CSC generates more compact features due to its intrinsic shift-invariant property. Moreover, CSC is capable of generating more complex filters capturing higher-older image statistics, compared to sparse coding that basically learns edge primitives [25].

In the proposed multispectral feature learning framework, CSC is applied to each separate spectral channel, yielding target-specific filter banks. For instance, some biologically meaningful filters are learned from the nuclear channel and the collagen channel respectively, as illustrated in Figure 4.2. Features extracted from multiple spectrums are summarized by SPM [97] at various scales and locations, and ultimate tissue representations are fed into linear SVM [98] for classification.

4.4.1 Convolutional Sparse Coding

Let $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^N$ be a training set containing N 2D images with dimension $m \times n$. Let $\mathbf{D} = \{\mathbf{d}_k\}_{k=1}^K$ be the 2D convolutional filter bank having K filters, where each $\mathbf{d}_k$ is an $h \times h$ convolutional kernel. Define $\mathbf{Z} = \{\mathbf{Z}^i\}_{i=1}^N$ be the set of sparse feature maps such that subset $\mathbf{Z}^i = \{\mathbf{z}_k^i\}_{k=1}^K$ consists of K feature maps for reconstructing image $\mathbf{x}_i$,

where $\mathbf{z}_k^i$ has dimension $(m+h-1)\times(n+h-1)$. Convolutional sparse coding aims to decompose each training image $\mathbf{x}_i$ as the sum of a series of sparse feature maps $\mathbf{z}_k^i \in \mathbf{Z}^i$ convolved with kernels $\mathbf{d}_k$ from the filter bank $\mathbf{D}$, by solving the following objective function:

$$\begin{aligned} \min_{\mathbf{D},\mathbf{Z}} \quad & \mathcal{L} = \sum_{i=1}^{N} \left\{ \left\| \mathbf{x}_i - \sum_{k=1}^{K} \mathbf{d}_k * \mathbf{z}_k^i \right\|_{\mathrm{F}}^2 + \alpha \sum_{k=1}^{K} \left\| \mathbf{z}_k^i \right\|_1 \right\} \\ \text{s.t.} \quad & \|\mathbf{d}_k\|_2^2 = 1, \forall k = 1, \ldots, K \end{aligned} \tag{4.4}$$

where the first and the second term represent the reconstruction error and the ℓ_1-norm penalty respectively; α is a regularization parameter; $*$ is the 2D discrete convolution operator; and filters are restricted to have unit energy to avoid trivial solutions. Note that here $\|\mathbf{z}\|_1$ represents the entry-wise matrix norm, *i.e.,* $\|\mathbf{z}\|_1 = \sum_{i,j} |z_{ij}|$, where z_{ij} is the entry at location (i,j) of a feature map $\mathbf{z} \in \mathbf{Z}$. The construction of $\mathbf{D}$ is realized by balancing the reconstruction error and the ℓ_1-norm penalty.

Note that the objective of Eq. (4.4) is not jointly convex with respect to (w.r.t.) $\mathbf{D}$ and $\mathbf{Z}$ but is convex w.r.t. one of the variables with the other keeping fixed [35]. Thus, we solve Eq. (4.4) by alternatively optimize the two variables, *i.e.,* iteratively performing the two steps that first compute $\mathbf{Z}$ and then update $\mathbf{D}$. We use the Iterative Shrinkage Thresholding Algorithm (ISTA) to solve for the sparse feature maps $\mathbf{Z}$. On updating the convolutional dictionary $\mathbf{D}$, we use the stochastic gradient descent for efficient estimation of the gradient by considering one training sample at a time [25]. The optimization procedure is sketched in Algorithm 4. Alternative methods for updating the dictionary can be found in [26, 27, 43].

4.4.2 Multispectral Feature Extraction

In the field of biomedical imaging, different spectrums usually capture distinct targets of interest. Specifically, in our case, color decomposition [129] produces two separate spectrums (channels) which characterize the nuclear chromatin and the collagen,

Algorithm 2 CSC Algorithm

Input: Training set $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^{N}$, K, α
Output: Convolutional filter bank $\mathbf{D} = \{\mathbf{d}_k\}_{k=1}^{K}$

1: **Initialize:** $\mathbf{D} \sim \mathcal{N}(0, 1)$, $\mathbf{Z} \leftarrow \mathbf{0}$
2: **repeat**
3: **for** $i = 1$ to N **do**
4: Normalize each kernel in $\mathbf{D}$ to unit energy
5: Fixing $\mathbf{D}$, compute sparse feature maps $\mathbf{Z}^i$ by solving
$$\mathbf{Z}^i \leftarrow \arg\min_{\mathbf{z}_k^i \in \mathbf{Z}^i} \|\mathbf{x}_i - \sum_{k=1}^{K} \mathbf{d}_k * \mathbf{z}_k^i\|_{\mathrm{F}}^2 + \alpha \sum_{k=1}^{K} \|\mathbf{z}_k^i\|_1$$
6: Fixing $\mathbf{Z}$, update $\mathbf{D}$ as
$$\mathbf{D} \leftarrow \mathbf{D} - \mu \nabla_{\mathbf{D}} \mathcal{L}(\mathbf{D}, \mathbf{Z})$$
7: **end for**
8: **until** Convergence (maximum iterations reached or objective function $\leq$ threshold)

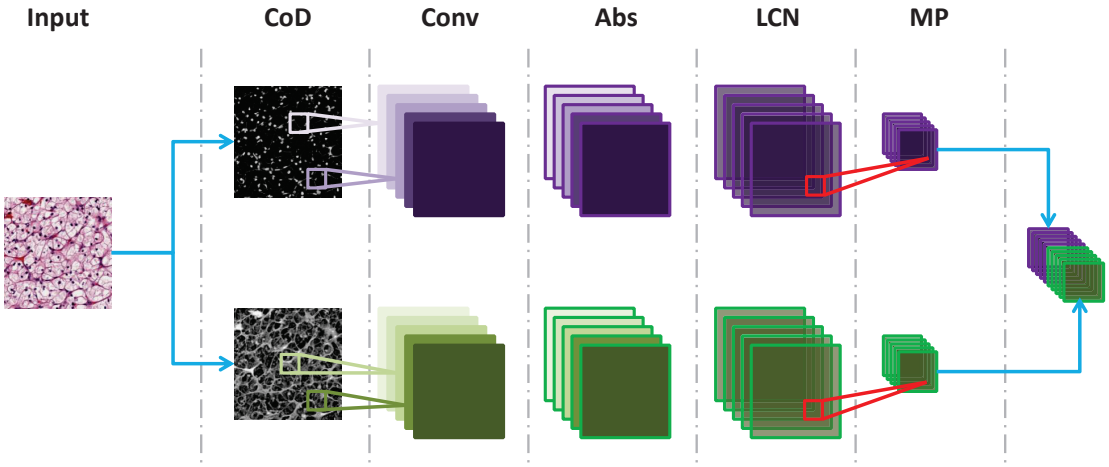


Figure 4.4: The proposed multispectral feature extraction framework. CoD means color decomposition; Abs means absolute value rectification; LCN means local contrast normalization; MP means max-pooling. The figure is best viewed in color at 150% zoom-in.

respectively (as shown in Figure 4.4). Therefore, in the filter learning phase, we propose to apply convolutional sparse coding to each spectrum, separately, for the purpose of learning biological-component-specific feature detectors. Without loss of generality, we assume that the number of filters for each spectrum (channel) is K and there are S spectrums (channels) after decomposition. Define 2D feature map $\mathbf{y}_k^s = \mathbf{d}_k^s * \hat{\mathbf{x}}^s$, for $1 \leq k \leq K$ and $1 \leq s \leq S$, where $\hat{\mathbf{x}}^s$ is the s-th spectrum component of input image $\mathbf{x}$

and $\mathbf{d}_k^s \in \mathbf{D}^s$ is the k-th convolutional kernel in filter bank $\mathbf{D}^s$ learned over spectrum with index s.

Upon finishing filter bank learning, we extract multispectral tissue features using the proposed framework illustrated in Figure 4.4, where an input image is first decomposed and divided into several spectral channels and then each decomposed component is convolved with the channel-specific filter bank followed by three cascaded layers, namely, element-wise absolute value rectification (Abs), local contrast normalization (LCN) and max-pooling (MP) [24]. Note that for specificity, the model in Figure 4.4 shows only two spectrums, but it is straightforward to generalize to multiple spectrums for different biomedical applications. The Abs layer computes absolute value elementwisely in each feature map, $\mathbf{y}_k^s$, to avoid the cancellation effect in sequential operations. The LCN layer aims to enhance the stronger feature responses and suppress weaker ones across feature maps, $\{\mathbf{y}_k^s\}_{k=1}^K$, in each spectrum, by performing local subtractive and divisive operations[1]. The MP layer partitions each feature map into non-overlapping windows and extracts the maximum response from each of the pooling window. The MP operation allows local invariance to translation [24]. Finally, the multispectral tissue features are formed by aggregating feature responses from all spectrums.

We further denote the multispectral tissue features of image, $\mathbf{x}$, as a 3D array, $\mathbf{U} \in \mathbb{R}^{a \times b \times KS}$, where the first two dimensions indicate the horizontal and vertical locations of a feature vector in the image plane and the third dimension represents the length of feature vectors. The multispectral tissue features are then fed into the SPM framework for classification as detailed in Section 4.3.

4.5 The SCCR Algorithm for Nuclei Segmentation

4.5.1 Training Algorithm

We consider the nuclei segmentation as a binary classification problem at pixel level. Let $\mathbf{X} = \{\mathbf{x}^i\}_{i=1}^N$ be a training set containing N 2D images with dimension

[1] Limited by space, we refer readers to [24,44] for detailed discussions on local contrast normalization.

$m \times n$. Let $\mathbf{Y} = \{\mathbf{y}^i\}_{i=1}^N$ be the set of N binary masks, where $\mathbf{y}^i$ is an $m \times n$ binary matrix corresponding to image $\mathbf{x}^i$, and each pixel $y_{j,k}^i \in \{0, 1\}$ in $\mathbf{y}^i$ indicates the label of the pixel $x_{j,k}^i$ in image $\mathbf{x}^i$. Here, we use $y_{j,k}^i = 1$ to denote the nuclear region and use $y_{j,k}^i = 0$ to represent the background. Let $\mathbf{D} = \{\mathbf{d}_k\}_{k=1}^K$ be the 2D convolutional filter bank consisting of K filters, where each $\mathbf{d}_k$ is an $h \times h$ convolutional kernel. Let $\mathbf{w} = [w_1, \ldots, w_K]^{\mathrm{T}} \in \mathbb{R}^K$ be the vector containing K linear combination coefficients, where the k^{th} coefficient w_k is related to the k^{th} filter $\mathbf{d}_k \in \mathbf{D}$. Our goal therefore is simultaneously achieving two objectives. The first is to learn a set of nuclei feature detectors $\mathbf{D}$ that can capture intrinsic cellular morphometric patterns. The second objective is to realize a sparse representation $\mathbf{w}$ which maps the feature vector extracted at each pixel to its label. The optimization problem is formulated as

$$\min_{\mathbf{D},\mathbf{w}} \quad \mathcal{L} = \sum_{i=1}^{N} \left\| \mathbf{y}^i - \sum_{k=1}^{K} w_k \sigma\left(\mathbf{d}_k * \mathbf{x}^i\right) \right\|_{\mathrm{F}}^2 + \quad \alpha \sum_{k=1}^{K} \|\mathbf{d}_k\|_{\mathrm{F}}^2 + \beta \|\mathbf{w}\|_1 \tag{4.5}$$

where the first term represents the segmentation error, the second term is a regularization term for penalizing the model complexity and the third term is ℓ_1 regularization term included for enforcing the linear representation vector $\mathbf{w}$ to be sparse; α, β are positive regularization constants; σ denotes the sigmoid function; $*$ is the 2D convolution operator. The sparsity constraint enables feature selection [130] and thus allows the filters to capture diversified nuclear patterns.

We solve Eq. (4.5) by alternatively optimizing the two variables, *i.e.,* iteratively performing the two steps, that is, first compute $\mathbf{w}$ and then update $\mathbf{D}$. For the purpose of handling large-scale dataset, we follow the mini-batch based training protocol [131], *i.e.,* in each iteration, computing the gradient based on a small subset of the dataset. Specifically, we use the conjugate gradient method [132] to solve for the sparse representation vector $\mathbf{w}$. On updating the convolutional filter bank $\mathbf{D}$, we use the Limited memory BFGS (L-BFGS) for efficient estimation of the gradient. The optimization procedure is sketched in Algorithm 4. Alternative methods for updating the dictionary can be found in [25–27, 43]. Note that the objective of Eq. (4.5) is

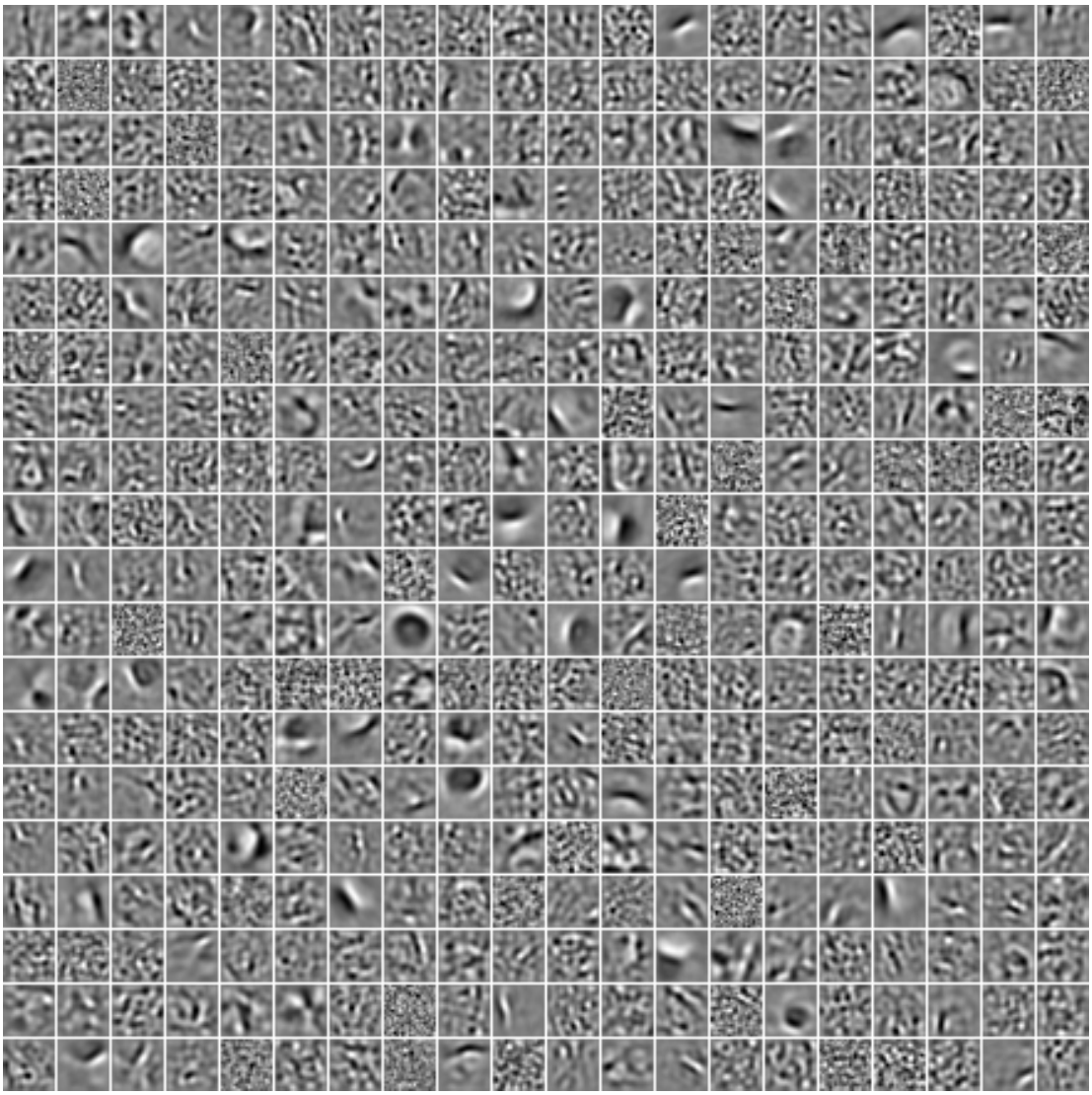


Figure 4.5: 21×21 filters learned from the TCGA segmentation benchmark dataset.

convex with respect to $\mathbf{w}$ but it is not convex with respect to $\mathbf{D}$ due to the nonlinear sigmoid function, and therefore the optimization can only guarantee the convergence to a local minima. However, in practice, achieving local minima is sufficient to generate satisfactory performance. Figure 4.5 illustrates some of the filters learned from the TCGA segmentation benchmark dataset.

Algorithm 3 Training Algorithm

Input: Training image set $\mathbf{X}$, training binary mask set $\mathbf{Y}$, filter bank size K, mini-batch size T, regularization constants α and β

Output: Convolutional filter bank $\mathbf{D}$, coefficient vector $\mathbf{w}$

1: **Initialize:** $\mathbf{D} \sim \mathcal{N}(0,1)$, $\mathbf{w} \leftarrow \mathbf{0}$

2: **repeat**

3: Generate a random index set $\Omega \subset \{1, 2, \ldots, N\}$ containing $|\Omega| = T$ indices

4: Fixing $\mathbf{D}$, compute $\mathbf{w}$ by solving

$$\mathbf{w} \leftarrow \arg\min_{\mathbf{w}} \sum_{i \in \Omega} \left\| \mathbf{y}^i - \sum_{k=1}^{K} w_k \sigma\left(\mathbf{d}_k * \mathbf{x}^i\right) \right\|_{\mathrm{F}}^2 + \beta \left\|\mathbf{w}\right\|_1$$

5: Fixing $\mathbf{w}$, update $\mathbf{D}$ over the same training subset as

$$\mathbf{D} \leftarrow \mathbf{D} - \mu \nabla_{\mathbf{D}} \mathcal{L}(\mathbf{D}, \mathbf{w})$$

6: **until** Convergence (maximum iterations reached or objective function $\leq$ threshold)

4.5.2 Decision Function

Now, we suppose the training of the proposed SCCR is completed. Given a test image $\mathbf{x}$ of dimension $p \times q$, the segmentation process consists of three steps. First, compute the convolutional regression prediction as

$$\mathbf{z} = \sum_{k=1}^{K} w_k \sigma\left(\mathbf{d}_k * \mathbf{x}\right) \tag{4.6}$$

Second, feed the prediction $\mathbf{z}$ into sigmoid function to squash the value of every pixel within the range of $(0, 1)$. Finally, the label of pixel at location (i, j) for all $i = 1, \ldots, p$ and $j = 1, \ldots, q$ is predicted according to the following decision rule

$$\text{Label}(x_{i,j}) = \begin{cases} 1 & \text{if } \sigma(z_{ij}) \geq 0.5 \\ 0 & otherwise \end{cases} \tag{4.7}$$

where $x_{i,j}$ and $z_{i,j}$ represent the pixel at location (i, j) in $\mathbf{x}$ and $\mathbf{z}$ respectively.

4.6 Experiments on Tissue Classification

In this section, we discuss the performance of PSD^{n}SPM and MCSCSPM in tissue histopathology classification respectively, by presenting detailed experiment setup and evaluation results. The two distinct tumor datasets, for evaluation, are curated

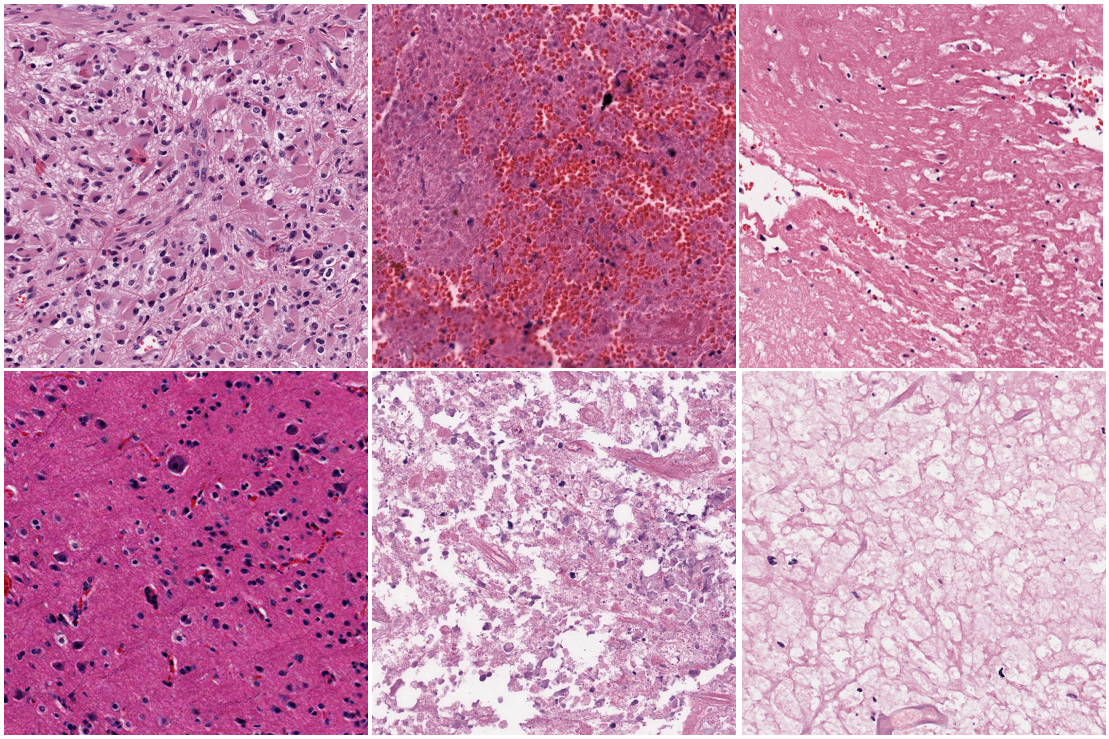


Figure 4.6: GBM Examples. First column: Tumor; Second column: Transition to necrosis; Third column: Necrosis.

from The Cancer Genome Atlas (TCGA), namely (i) Glioblastoma Multiforme (GBM) and (ii) Kidney Renal Clear Cell Carcinoma (KIRC), which are publicly available from the NIH (National Institute of Health) repository.

4.6.1 The Datasets

1. GBM Dataset. The GBM dataset contains 3 classes: Tumor, Necrosis, and Transition to Necrosis, which were curated from whole slide images (WSI) scanned with a 20X objective (0.502 micron/pixel). Examples can be found in Figure 4.6. The number of images per category are 628, 428 and 324, respectively. Most images are 1000 × 1000 pixels. In this experiment, we train on 40, 80 and 160 images per category and tested on the rest, with three different dictionary sizes: 256, 512 and 1024. Detailed comparisons are shown in Table 4.3.

2. KIRC Dataset. The KIRC dataset contains 3 classes: Tumor, Normal, and

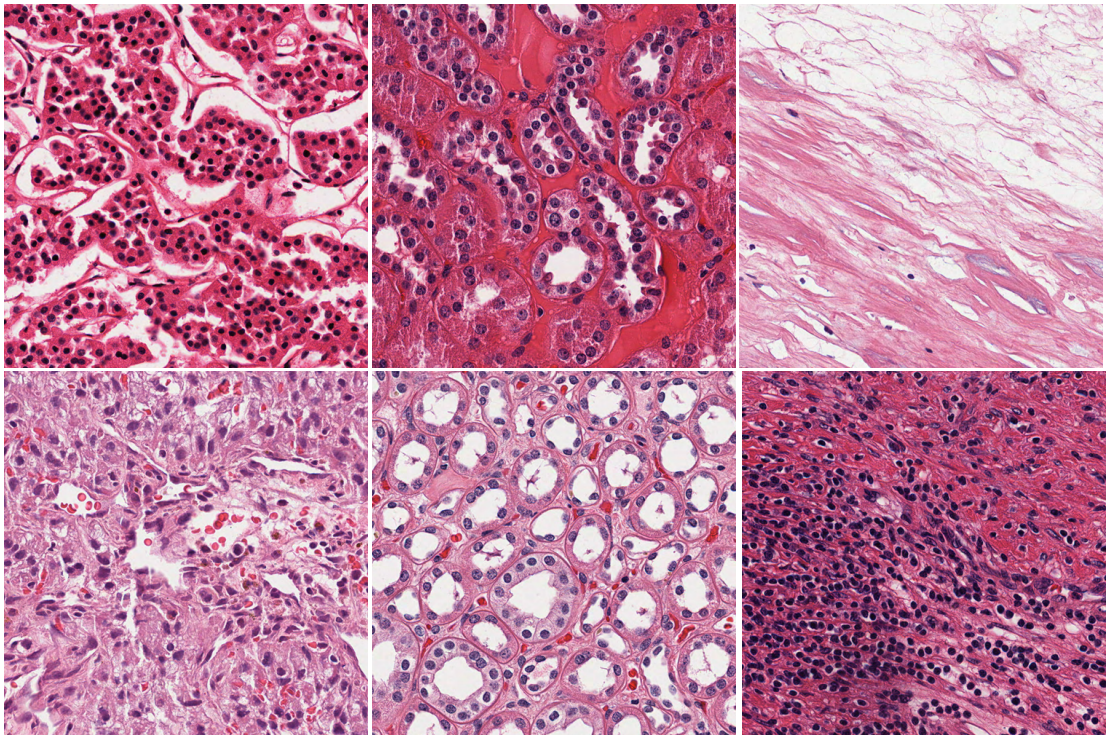


Figure 4.7: KIRC Examples. First column: Tumor; Second column: Normal; Third column: Stromal.

Stromal, which were curated from whole slide images (WSI) scanned with a 40X objective (0.252 micron/pixel). Examples can be found in Figure 4.7. The number of images per category are 568, 796 and 784, respectively. Most images are 1000×1000 pixels. In this experiment, we train on 70, 140 and 280 images per category and tested on the rest, with three different dictionary sizes: 256, 512 and 1024. Detailed comparisons are shown in Table 4.4.

4.6.2 Evaluating the PSD^{n}SPM Algorithm

4.6.2.1 Experimental Configurations

We evaluate the proposed PSD^{n}SPM using the following different setups:

1. PSD^{n}SPM^{NR}: The nonlinear kernel SPM that uses spatial-pyramid histograms of sparse tissue morphometric types. In this implementation,

   (a) $n = 1, 2$;

(b) The nonlinear regressor ($\mathbf{Z} = \mathbf{G}\sigma(\mathbf{WX})$) was trained for the inference of $\mathbf{Z}$;

(c) The image patch size is fixed to be 20×20 and the number of basis functions in the top layer was fixed to be 1024. We adopted the SPAMS optimization toolbox [133] for efficient implementation of OMP to compute the sparse code, $\mathbf{Z}$, with sparsity prior set to 30;

(d) Standard K-means clustering was used for the construction of the dictionary;

(e) The level of pyramid was fixed to be 3; and

(f) The homogeneous kernel map was applied, followed by the linear SVM for classification.

2. PSD^{1}SPM^{LR} [92]: The nonlinear kernel SPM that uses spatial-pyramid histograms of sparse tissue morphometric types. In this implementation,

(a) The linear regressor ($\mathbf{Z} = \mathbf{WX}$) was trained for the inference of $\mathbf{Z}$;

(b) For consistency, the image patch size and the number of basis functions was fixed at 20×20 and 1024, respectively. The sparsity constraint was set at 0.3 for best performance following cross validation.

(c) Standard K-means clustering was used for the construction of the dictionary;

(d) The level of pyramid was fixed to be 3;

(e) The homogeneous kernel map was applied, followed by linear SVM for classification.

3. ScSPM [134]: The linear SPM that utilizes linear kernel on spatial-pyramid pooling of SIFT sparse codes. In this implementation,

(a) The dense SIFT features was extracted on 16×16 patches sampled from each image on a grid with stepsize 8 pixels;

(b) The sparse constraint parameter λ was fixed to be 0.15, which was determined empirically to achieve the best performance;

(c) The level of pyramid was fixed to be 3;

(d) Linear SVM was used for classification.

4. KSPM [97]: The nonlinear kernel SPM that uses spatial-pyramid histograms of SIFT features; In the implementation,

   (a) The dense SIFT features was extracted on 16×16 patches sampled from each image on a grid with stepsize 8 pixels;

   (b) Standard K-means clustering was used for the construction of the dictionary;

   (c) The level of pyramid was fixed to be 3;

   (d) The homogeneous kernel map was applied, followed by linear SVM for classification.

5. CTSPM: The nonlinear kernel SPM that uses spatial-pyramid histograms of color and texture features; In this implementation,

   (a) Color features were extracted from the RGB color space;

   (b) Texture features were extracted via steerable filters [135] with 4 directions ($\theta \in \{0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}\}$) and 5 scales ($\sigma \in \{1, 2, 3, 4, 5\}$) from the grayscale image;

   (c) The feature vector was constructed by concatenating texture and mean color on 20×20 patches, empirically, to achieve the best performance;

   (d) Standard K-means clustering was used for the construction of the dictionary;

   (e) The level of pyramid was fixed to be 3;

   (f) The homogeneous kernel map was applied, followed by linear SVM for classification.

All experimental processes were repeated 10 times with randomly selected training and testing images. The final results were reported as the mean and standard deviation of the classification rates on the two distinct datasets, *i.e.,* GBM and KIRC datasets, which include vastly different tumor types.. The results are summarized as below.

	Method	DictionarySize=256	DictionarySize=512	DictionarySize=1024
160 training	PSD^{2}SPM^{NR}	91.85 ± 1.03	91.86 ± 0.78	**92.07 ± 0.65**
	PSD^{1}SPM^{NR}	91.85 ± 0.69	91.89 ± 0.99	91.74 ± 0.85
	PSD^{1}SPM^{LR} [92]	91.02 ± 1.89	91.41 ± 0.95	91.20 ± 1.29
	ScSPM [134]	79.58 ± 0.61	81.29 ± 0.86	82.36 ± 1.10
	KSPM [97]	85.00 ± 0.79	86.47 ± 0.55	86.81 ± 0.45
	CTSPM	78.61 ± 1.33	78.71 ± 1.18	78.69 ± 0.81
80 training	PSD^{2}SPM^{NR}	90.51 ± 1.06	**90.88 ± 0.66**	90.51 ± 1.06
	PSD^{1}SPM^{NR}	90.74 ± 0.95	90.42 ± 0.94	89.70 ± 1.20
	PSD^{1}SPM^{LR} [92]	88.63 ± 0.91	88.91 ± 1.18	88.64 ± 1.08
	ScSPM [134]	77.65 ± 1.43	78.31 ± 1.13	81.00 ± 0.98
	KSPM [97]	83.81 ± 1.22	84.32 ± 0.67	84.49 ± 0.34
	CTSPM	75.93 ± 1.18	76.06 ± 1.52	76.19 ± 1.33
40 training	PSD^{2}SPM^{NR}	87.90 ± 0.91	**88.21 ± 0.90**	87.71 ± 0.81
	PSD^{1}SPM^{NR}	87.72 ± 1.21	86.99 ± 1.76	86.33 ± 1.32
	PSD^{1}SPM^{LR} [92]	84.06 ± 1.16	83.72 ± 1.46	83.40 ± 1.14
	ScSPM [134]	73.60 ± 1.68	75.58 ± 1.29	76.24 ± 3.05
	KSPM [97]	80.54 ± 1.21	80.56 ± 1.24	80.46 ± 0.56
	CTSPM	73.10 ± 1.51	72.90 ± 1.09	72.65 ± 1.41

Table 4.1: Performance of different methods on the GBM dataset.

	Method	DictionarySize=256	DictionarySize=512	DictionarySize=1024
280 training	PSD^{2}SPM^{NR}	**99.03 ± 0.20**	98.89 ± 0.19	98.92 ± 0.21
	PSD^{1}SPM^{NR}	98.98 ± 0.35	98.81 ± 0.45	98.69 ± 0.41
	PSD^{1}SPM^{LR} [92]	97.19 ± 0.49	97.27 ± 0.44	97.08 ± 0.45
	ScSPM [134]	94.52 ± 0.44	96.37 ± 0.45	96.81 ± 0.50
	KSPM [97]	93.55 ± 0.31	93.76 ± 0.27	93.90 ± 0.19
	CTSPM	87.45 ± 0.59	87.95 ± 0.49	88.53 ± 0.49
140 training	PSD^{2}SPM^{NR}	**98.26 ± 0.34**	98.07 ± 0.46	97.85 ± 0.56
	PSD^{1}SPM^{NR}	98.17 ± 0.72	98.05 ± 0.71	97.99 ± 0.82
	PSD^{1}SPM^{LR} [92]	96.80 ± 0.75	96.52 ± 0.76	96.55 ± 0.84
	ScSPM [134]	93.46 ± 0.55	95.68 ± 0.36	96.76 ± 0.63
	KSPM [97]	92.50 ± 1.12	93.06 ± 0.82	93.26 ± 0.68
	CTSPM	86.55 ± 0.99	86.40 ± 0.54	86.49 ± 0.58
70 training	PSD^{2}SPM^{NR}	**96.67 ± 0.53**	96.20 ± 0.54	95.57 ± 0.66
	PSD^{1}SPM^{NR}	96.42 ± 0.68	96.41 ± 0.59	96.03 ± 0.69
	PSD^{1}SPM^{LR} [92]	95.12 ± 0.54	95.13 ± 0.51	95.09 ± 0.40
	ScSPM [134]	91.93 ± 1.00	93.67 ± 0.72	94.86 ± 0.86
	KSPM [97]	90.78 ± 0.98	91.34 ± 1.13	91.59 ± 0.97
	CTSPM	84.76 ± 1.32	84.29 ± 1.53	83.71 ± 1.42

Table 4.2: Performance of different methods on the KIRC dataset.

4.6.2.2 Discussion

Above experiments indicate that,

1. Features from unsupervised feature learning are more tolerant to the batch effect than human engineered features for tissue classification. Tables 4.1 and 4.2 show that PSD^{n}SPM consistently outperforms KSPM, ScSPM and CTSPM on the two distinct datasets that suffer from technical variations as a result of both sample

preparation and biological heterogeneity, where the latter is due to the variation in tumor phenotype across patients.

2. PSD with nonlinear regressor outperforms PSD with linear regressor in terms of both reconstruction and classification, as shown in Figure 4.8 as well as Tables 4.1 and 4.2.

3. Stacking multiple layers of PSD enables learning higher level features, which further improves the classification performance.

4.6.3 Evaluating the MCSCSPM algorithm

4.6.3.1 Experimental Configurations

We have evaluated the proposed method (MCSCSPM) in three different variations:

1. MCSCSPM-HE: Convolutional filter banks are learned from / applied onto decomposed spectrum (channel) separately. Here, we have two spectrums (channels) after color decomposition, which correspond to nuclear chromatin (stained with hematoxylin) and collagen (stained with eosin), respectively.

2. MCSCSPM-RGB: Convolutional filter banks are learned from / applied onto R, G, and B channels separately.

3. MCSSPM-Gray: Convolutional filter banks are learned from / applied onto the grayscale image.

and compared its performance with other four classification methods on the GBM and KIRC datasets. Implementation details of all approaches involved are listed as follows:

1. MCSCSPM: the nonlinear kernel SPM that uses spatial-pyramid histograms of multispectral tissue types and homogeneous kernel map. In the multispectral case, an input tissue image was decomposed into two spectrums (*i.e.*, $S = 2$) corresponding to the nuclear chromatin and the collagen respectively, based on

the optical density matrix established in [129]. In the RGB and grayscale case, each color channel was treated as one spectrum. For each spectrum, images were preprocessed with a 13×13 Gaussian filter. During training, we set K to 150 and 300 per spectrum for the GBM and KIRC datasets, respectively. The filter dimension was 27×27 for both datasets. The sparsity regularization parameter α was set to 0.1 for best performance. During multispectral feature extraction, we used the same 13×13 Gaussian filter for local contrast normalization and empirically set the max-pooling stepsize to be 27.

2. PSDSPM [92]: the nonlinear kernel SPM that uses spatial-pyramid histograms of sparse tissue morphometric types and homogeneous kernel map. The image patch size was set to 20×20, the number of basis function was empirically set to 1024 and the sparsity regularization parameter was set to 0.3 for best performance.

3. ScSPM [134]: the linear SPM that uses linear kernel on spatial-pyramid pooling of SIFT sparse codes. The dense SIFT features was extracted on 16×16 patches sampled from each image on a grid with stepsize 8 pixels. The sparsity regularization parameter λ was set to 0.15, to achieve the best performance;

4. KSPM [97]: the nonlinear kernel SPM that uses spatial-pyramid histograms of SIFT features and homogeneous kernel map. The dense SIFT features was extracted on 16×16 patches sampled from each image on a grid with stepsize 8 pixels;

5. SMLSPM [91]: the linear SPM that uses linear kernel on spatial-pyramid pooling of cellular morphometric sparse codes.

On the implementation of SPM for MCSCSPM, PSDSPM, KSPM and SMLSPM, we use the standard K-means clustering for constructing the dictionary and set the level of pyramid to be 3. Following the conventional evaluation procedure, we repeat all experiments 10 times with random splits of training and test set to obtain reliable results. The final results are reported as the mean and standard deviation of the

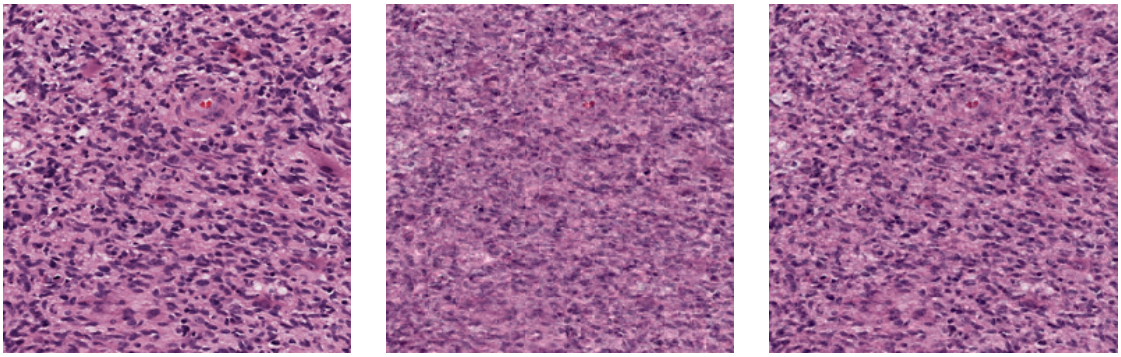
	Method	DictionarySize=256	DictionarySize=512	DictionarySize=1024
160 training	MCSCSPM-HE	92.71 ± 0.91	**93.01 ± 1.10**	92.65 ± 0.75
	MCSCSPM-RGB	**92.58 ± 0.94**	92.50 ± 0.86	92.47 ± 0.73
	MCSCSPM-Gray	86.33 ± 1.12	**86.74 ± 0.91**	86.69 ± 0.81
	PSDSPM [92]	91.02 ± 1.89	**91.41 ± 0.95**	91.20 ± 1.29
	SMLSPM [91]	92.35 ± 0.83	92.57 ± 0.91	**92.91 ± 0.84**
	ScSPM [134]	79.58 ± 0.61	81.29 ± 0.86	**82.36 ± 1.10**
	KSPM [97]	85.00 ± 0.79	86.47 ± 0.55	**86.81 ± 0.45**
80 training	MCSCSPM-HE	**91.41 ± 1.07**	91.19 ± 0.91	91.13 ± 0.93
	MCSCSPM-RGB	90.88 ± 1.06	**91.28 ± 0.82**	90.85 ± 0.67
	MCSCSPM-Gray	**84.67 ± 1.63**	84.53 ± 1.58	84.56 ± 1.62
	PSDSPM [92]	88.63 ± 0.91	**88.91 ± 1.18**	88.64 ± 1.08
	SMLSPM [91]	90.82 ± 1.28	90.29 ± 0.68	**91.08 ± 0.69**
	ScSPM [134]	77.65 ± 1.43	78.31 ± 1.13	**81.00 ± 0.98**
	KSPM [97]	83.81 ± 1.22	84.32 ± 0.67	**84.49 ± 0.34**
40 training	MCSCSPM-HE	89.16 ± 1.04	**89.21 ± 0.75**	88.84 ± 0.83
	MCSCSPM-RGB	89.24 ± 1.03	89.46 ± 1.14	**89.53 ± 1.20**
	MCSCSPM-Gray	**81.37 ± 1.55**	81.31 ± 1.19	80.80 ± 1.71
	PSDSPM [92]	**84.06 ± 1.16**	83.72 ± 1.46	83.40 ± 1.14
	SMLSPM [91]	88.05 ± 1.38	87.88 ± 1.04	**88.54 ± 1.42**
	ScSPM [134]	73.60 ± 1.68	75.58 ± 1.29	**76.24 ± 3.05**
	KSPM [97]	80.54 ± 1.21	**80.56 ± 1.24**	80.46 ± 0.56

Table 4.3: Performance of different methods on the GBM dataset.

	Method	DictionarySize=256	DictionarySize=512	DictionarySize=1024
280 training	MCSCSPM-HE	97.39 ± 0.36	**97.51 ± 0.41**	97.48 ± 0.40
	MCSCSPM-RGB	97.11 ± 0.44	**97.49 ± 0.46**	97.44 ± 0.43
	MCSCSPM-Gray	88.76 ± 0.59	90.50 ± 0.70	**91.28 ± 0.72**
	PSDSPM [92]	97.19 ± 0.49	**97.27 ± 0.44**	97.08 ± 0.45
	SMLSPM	98.15 ± 0.46	**98.50 ± 0.42**	98.21 ± 0.44
	ScSPM [134]	94.52 ± 0.44	96.37 ± 0.45	**96.81 ± 0.50**
	KSPM [97]	93.55 ± 0.31	93.76 ± 0.27	**93.90 ± 0.19**
140 training	MCSCSPM-HE	96.73 ± 0.84	**96.89 ± 0.48**	96.84 ± 0.67
	MCSCSPM-RGB	96.14 ± 1.17	96.46 ± 1.06	**96.64 ± 0.76**
	MCSCSPM-Gray	86.79 ± 0.98	88.26 ± 0.59	**88.50 ± 0.80**
	PSDSPM [92]	**96.80 ± 0.75**	96.52 ± 0.76	96.55 ± 0.84
	SMLSPM	97.40 ± 0.50	**97.98 ± 0.35**	97.35 ± 0.48
	ScSPM [134]	93.46 ± 0.55	95.68 ± 0.36	**96.76 ± 0.63**
	KSPM [97]	92.50 ± 1.12	93.06 ± 0.82	**93.26 ± 0.68**
70 training	MCSCSPM-HE	95.32 ± 0.67	**95.62 ± 0.29**	95.40 ± 0.44
	MCSCSPM-RGB	94.45 ± 0.84	**94.64 ± 0.72**	94.45 ± 0.77
	MCSCSPM-Gray	84.04 ± 1.10	**85.13 ± 0.79**	84.66 ± 1.14
	PSDSPM [92]	95.12 ± 0.54	**95.13 ± 0.51**	95.09 ± 0.40
	SMLSPM	96.20 ± 0.85	**96.37 ± 0.85**	96.19 ± 0.62
	ScSPM [134]	91.93 ± 1.00	93.67 ± 0.72	**94.86 ± 0.86**
	KSPM [97]	90.78 ± 0.98	91.34 ± 1.13	**91.59 ± 0.97**

Table 4.4: Performance of different methods on the KIRC dataset.

classification rates on the two distinct datasets, *i.e.*, GBM and KIRC datasets, which include vastly different tumor types.



(a) (b) (c)

Figure 4.8: Comparison of PSD with linear and nonlinear regressors in terms of reconstruction. (a) Original image; (b) Reconstruction by PSD with linear regressor (SNR=14.9429); (c) Reconstruction by PSD with nonlinear regressor (SNR=19.3436).

4.6.3.2 Discussion

1. Multispectral (HE) v.s. RGB v.s. Gray. For GBM dataset, K was fixed to be 150 per spectrum (channel), which led to a total number of 300, 450 and 150 filters for MCSCSPM-HE, MCSCSPM-RGB and MCSCSPM-Gray, respectively. For the KIRC dataset, K was fixed to be 300 per spectrum (channel), which led to a total number of 600, 900 and 300 filters for MCSCSPM-HE, MCSCSPM-RGB and MCSCSPM-Gray, respectively. Table 4.3 and Table 4.4 show that, even with smaller number of filters, MCSCSPM-HE outperforms MCSCSPM-RGB in most cases. This is due to the fact that, after color decomposition, the resulting two spectrums are biological-component-specific, such that specialized filters can be obtained from each spectrum characterizing nuclear architecture and tissue structural connectivities, respectively, as demonstrated in Figure 4.2. Although the stain information (biological component information) leaks across channels for H&E stained tissue sections in its original RGB presentation, target-specific property can still be preserved to some extent (*e.g.,* most of the nuclear information resides in blue (B) channel); and this explains why MCSCSPM-RGB still has reasonable performance. However, when such a property is completely lost in grayscale, MCSCSPM-Gray sees a dramatic performance drop.

2. Convolutional v.s. patch-based sparse modeling. As listed in Table 4.3 and Table 4.4, the proposed approach, MCSCSPM-HE/MCSCSPM-RGB outperforms patch-based sparse feature learning models, *e.g.,* PSDSPM [92], with fewer filters than PSDSPM. These facts indicate that, in tissue classification, convolutional sparse coding is more effective than traditional sparse coding in terms of using more succinct representations and producing better results, which has already been confirmed in other applications [25].

3. Unsupervised feature learning v.s. hand-engineered features. As shown in Table 4.3 and Table 4.4, the proposed approach significantly outperforms systems that are built on hand-engineered features for general image classification purpose (*e.g.,* KSPM, ScSPM). Even compared to the recently proposed system, SMLSPM [91], which is built upon features with biological prior knowledge, the proposed approach, MCSCSPM, robustly achieves very competitive performance over the two different tumor types, where MCSCSPM-HE performs better on the GBM dataset, while worse on the KIRC dataset. This confirms that the proposed approach, MCSCSPM, is a useful tool for analyzing large cohorts with substantial technical variations and biological heterogeneities.

4.7 Experiments on Nuclei Segmentation

In this section, we present evaluation results of the proposed SCCR for nuclei segmentation. The Cancer Genome Atlas (TCGA) is a publicly accessible repository providing a rich amount of whole mount tumor sections that are collected from different laboratories. Among the images, there exist significant technical and biological variations. The proposed SCCR is evaluated over 21 1000-by-1000 Glioblastoma Multiforme (GBM) image samples (20X), which are manually selected to capture technical variations and are annotated as binary masks. For preprocessing all images, we used color decomposition [129] to accentuate the nuclear dye. The color decomposition generates two channels of the original image, *i.e.,* the nuclear channel and the collagen channel. For our segmentation task, we only keep the images from nuclear channel.

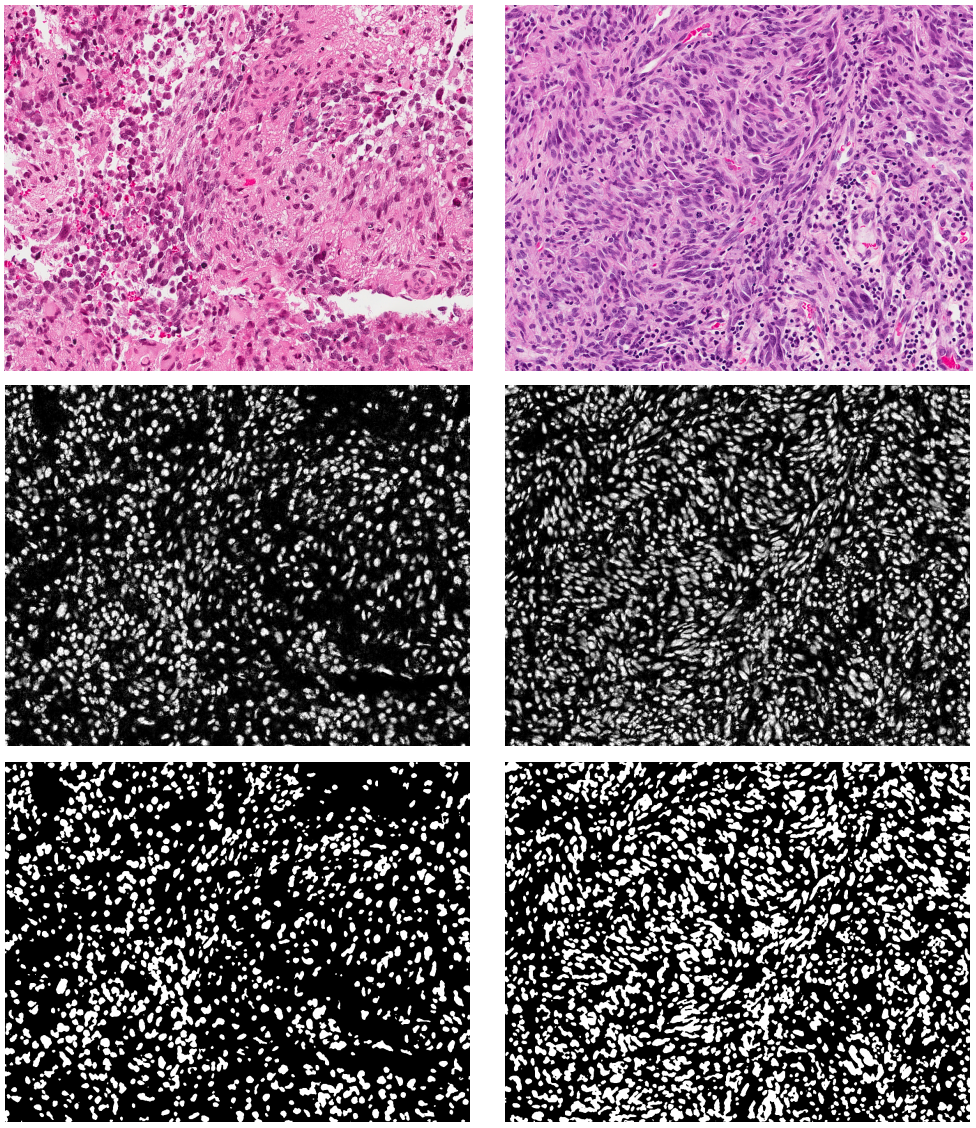


Figure 4.9: GBM Examples. First row: original images. Second row: predictions by SCCR. Third row: final segmentation results.

We randomly cropped 1400 image patches and the corresponding binary masks as training set. The image patches and the binary masks are of size 64-by-64. For training, we empirically set $K = 1500$, $T = 200$, $\alpha = 10^{-4}$, $\beta = 0.1$. We evaluate the proposed SCCR using all the 21 1000-by-1000 images and compare it with several methods reported in the literature [95, 136, 137]. The results are summarized in Table 4.5. Our method outperforms traditional nuclei segmentation algorithms [95, 136]

Table 4.5: Comparison of Segmentation Results.

Method	Precision	Recall	F-Score
Proposed	0.77	0.81	0.790
Chang *et al.* [137]	0.79	0.78	0.785
Chang *et al.* [136]	0.78	0.65	0.709
Sonal *et al.* [95]	0.69	0.75	0.719

and is very competitive with one of the state-of-the-art algorithm [137]. Note that unlike the algorithm in [137] which is built upon human's biological prior knowledge, the proposed SCCR is a generic feature learning model and may be applicable to segmentation tasks of other tumor types. Figure 4.9 illustrates some examples for the original images, the corresponding SCCR predictions and the final segmentation results.

4.8 Conclusion

In this chapter, we demonstrate the promising performance of automatic feature learning in biomedical image analysis. We first present two unsupervised feature learning frameworks for classification of distinct regions of tumor histopathology, *i.e.*, a multi-layer PSD framework (PSD^{n}SPM) and a more advanced model (MCSCSPM). Both approaches outperform traditional human-engineered feature extraction methods that are typically based on pixel- or patch-level features. Our analysis indicates that the proposed approaches are (i) extensible to different tumor types; (ii) robust in the presence of large amounts of technical variations and biological heterogeneities; (iii) scalable with varying training sample sizes; and (iv) competitive with state-of-the-art dedicated systems based on biological domain knowledge.

Then we present a novel method, called sparsity constrained convolutional regression (SCCR), for nuclei segmentation. In contrast to traditional CNN-based models for image labeling, our algorithm aims to accurately classify each pixel into nuclear region or background. Compared to human-engineered nuclei segmentation frameworks, our method does not rely on biological prior knowledge and could be potentially applicable to segmentation tasks of other tumor types. The proposed SCCR outperforms

several traditional nuclei segmentation algorithms and achieves very competitive performance compared to one of the state-of-the-art approaches based on biological prior knowledge.

Chapter 5

KERNEL SPARSE CODING FOR GESTURE RECOGNITION

5.1 Introduction

Sparse representation has achieved state-of-the-art results in many fields, such as image compression and denoising [33], face recognition [1, 11], video-based action classification [138], etc. The success of this technique is partially due to its robustness to noise and missing data. For example, sparse representation-based classification (SRC) [11] yields impressive results in face recognition by encoding a query face image over the entire set of training template images and identifying the label of the query sample by evaluating which class yields the minimum reconstruction error. However, little effort has been made to apply this technique to classifying multi-variate time series (MTS) data.

Classifying multivariate time series (MTS) is a challenging task in many areas, e.g., pattern recognition [28] and computer vision [29]. An MTS is an $m \times n$ matrix, where m is the number of observations on an individual event captured by sensors such as video cameras, position trackers and cybergloves, while n denotes the number of independent attributes [139], also known as variables [28,30] or features [2,140]. For each MTS, m is typically varying due to different motion durations for each instance, while the number of attributes, n, is the same for all the series since they are recorded by the same set of devices. For conventional feature extraction methods, e.g., PCA and LDA, downsampling and interpolation are usually applied on each MTS in order to normalize the data length. However, downsampling may cause a loss of salient information [28], while interpolation may induce distortion to the original data [30].

Gesture MTS data possess both spatial and temporal information. While spatial information depicts the entire static pattern, temporal information contains the

dynamic dependencies between adjacent recordings. Algorithms that exploit chronological order within time series, e.g., Dynamic Time Warping (DTW) [4,141] and Longest Common Subsequence (LCSS) [7], assume that similar signals must be recorded in the same order. However, motion order and direction may vary significantly among users presenting the same gesture. Consequently, such algorithms need to store all possible permutations of each gesture in memory and conduct pair-wise matching during recognition, resulting in excessive computation and storage requirements [142]. For example, a 2-stroke letter "t" requires $2! \times 2^2 = 8$ permutations to represent all possibilities, while an l-stroke gesture takes $l! \times 2^l$ permutations.

Notably, real-world gestures and movements, such as human gait and sign language, are performed according to a strict "grammar". This observation indicates that effectively distinguishing complicated spatial patterns is the key to successful recognition, rather than exploiting temporal order [28,30,139]. Motivated by this observation and reasoning, we consider feature extraction for MTS data ignoring the temporal ordering. More specifically, we generalize the capability of SRC to classifying MTS data.

The performance of SRC relies on the quality of the dictionary. We propose a novel feature extraction technique, called Covariance Matrix Singular Value Decomposition for Kernelization (CovSVDK), which possesses three notable merits: CovSVDK is 1) invariant to inconsistent lengths and temporal disorder across MTS data; 2) robust to the large variability within human gestures; 3) efficient to compute. In particular, the robustness of the feature extraction strategy is attributed to the fact that CovSVDK essentially enforces ℓ_1 minimization algorithms to favor training samples that are consistently close to the query sample in every sub-feature space. Moreover, we propose a new approach to kernelize sparse representation. With this method, dictionary atoms are more separable for sparse coding algorithms and nonlinear relationships among data can be conveniently transformed into linear relations in kernel space, which leads to more effective classification. Finally, we evaluate the proposed framework over extensive datasets. For the Georgia-Tech HG database, a 100% recognition rate is

stably achieved; over the High-quality Australian Sign Language (HAuslan) database, the recognition accuracy is greater than 91.2%; for the univariate UCR Time-Series Repository, the proposed classifier outperforms competing methods by achieving the lowest error rate on 10 out of 20 datasets.

5.2 Related Work and Problem Formulation

5.2.1 Related Work

Many algorithms have been proposed to measure the similarity among multi-dimensional time series, e.g., Hidden Markov models (HMMs) [143], DTW [4, 141], LCSS [7], and Mixture of Bayes Network Classifier [2], among others. Principal components (PCs) based methods are, perhaps, the most widely known similarity measure for multi-attribute time series, with the approach first defined by Krzanowski [144] in 1979. Many subsequent PC efforts focused on computing the similarity value using different weighting strategies to aggregate the inner products between PC pairs [28, 30, 139].

For instance, Li *et al.* proposed a similarity measure for motion streams using only the largest singular value and the corresponding singular vector [139]. In [29], the authors further proposed k Weighted Angular Similarity (kWAS) by considering the k largest singular value/vector pairs. Yang and Shahabi [28] proposed a similarity measure, called Extended Frobenius norm (Eros), which included all the singular values by employing a heuristic aggregating function to compute universal weights for all MTS data. The similarity measure is a weighted sum of inner products between each pair of singular vectors. In practice, however, variance is highly concentrated in the several largest eigenvalues and the small values are typically considered as redundancy or noise. Hence, Eros is vulnerable to noise. Yang and Shahabi further further extended their approach by using Eros for Kernel PCA, termed KEros [30].

Recently, some researchers reported the limitation of SRC [11] in classifying non-linear data. Zhang *et al.* [88] proposed the kernel sparse representation-based classifier (KSRC) by introducing the kernel trick. However, their approach relies on kernel-based dimensionality reduction techniques and thus does not offer a direct generalization to

sparse representation in kernel space. Gao *et al.* [145] proposed kernel sparse representation (KSR). However, the KSR objective function cannot be solved by standard sparse coding algorithms as it requires solving a quadratic programming (QP) problem, which is of higher computational complexity than ℓ_1 minimization.

5.2.2 Problem Formulation

In a k-label MTS data classification problem, we define the training set as $\mathbf{T} = \bigcup_{i=1}^{k} \mathbf{T}_i$, where $\mathbf{T}_i = \bigcup_{j=1}^{n_i} \mathbf{t}_{i,j}$ is a subset for the i-th class with n_i samples, and define the query sample as $\mathbf{x}$. Also, denote $N = \sum_i^k n_i$ as the total number of training samples.

There is significant current interest in using SRC [11] to classify audio, image and video signals. It is therefore desirable to explore its capability in the field of MTS data classification. To achieve this goal, several important issues must be addressed: 1) An effective feature extraction method is needed to process large-scale MTS datasets. The method should be efficient in computation and memory consumption, and invariant to inconsistent lengths and temporal disorder across MTS samples. 2) A general formulation of sparse representation suitable for various pattern recognition tasks is also desired. SRC assumes that training atoms reside on a linear manifold and are distinguishable by ℓ_1 minimization algorithms. While this premise holds for face images, it does not necessarily hold for other types of data.

5.3 Proposed method

This section details methods for effectively extracting MTS data features and present a novel approach to kernelizing sparse representation for classification.

5.3.1 Feature Extraction for MTS Data

5.3.1.1 SVD Properties of MTS Data

For an $m \times n$ MTS $\mathbf{t}$ with m observations and n attributes, m is typically much larger than n and varies across different samples. In order to avoid performing SVD

on m-varying $\mathbf{t}$, we treat each attribute (columns in the $\mathbf{t}$) as a random variable and compute the covariance matrix of $\mathbf{t}$ as

$$\mathbf{\Sigma}_t = \mathbf{E}[\mathbf{t}^T\mathbf{t}] - \mathbf{E}^T[\mathbf{t}]\mathbf{E}[\mathbf{t}], \tag{5.1}$$

where $\mathbf{E}[\cdot]$ denotes the mathematical expectation and $\mathbf{\Sigma}_t$ is of fixed dimension $n \times n$ (here $n \geq 2$). By calculating the $\mathbf{\Sigma}_t$ of $\mathbf{t}$, we discard the ordering information and thus overcome the problem of temporal disorder across MTS samples, since each entry in $\mathbf{\Sigma}_t$ is an inner product between two columns in $\mathbf{t}$ that is invariant to the row-switching of $\mathbf{t}$.

Applying SVD to the covariance matrix yields $\mathbf{\Sigma}_t = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$, where $\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_n]$ is a singular vector matrix with orthonormal columns and $\mathbf{\Lambda} = \texttt{diag}(\rho)$ with $\rho = [\lambda_1, \ldots, \lambda_n]^T$ being a vector with singular values descendingly sorted. `diag` is the operator that transforms ρ into a diagonal matrix by putting entries of ρ along the main diagonal in the matrix. Similarly, the covariance matrix $\mathbf{\Sigma}_p$ of MTS $\mathbf{p}$ can be expressed as $\mathbf{\Sigma}_p = \mathbf{V}\mathbf{\Omega}\mathbf{V}^T$, where $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_n]$ and $\mathbf{\Omega} = \texttt{diag}(\eta)$ with $\eta = [\omega_1, \ldots, \omega_n]^T$. Since $\mathbf{\Sigma}$ is positive semi-definite, its SVD is equivalent to eigenvalue decomposition.

If two MTS $\mathbf{t}$ and $\mathbf{p}$ are similar to each other, $\|\mathbf{\Sigma}_t - \mathbf{\Sigma}_p\|_F$ should be close to zero. In other words, the singular vector $\mathbf{u}_i$ of $\mathbf{\Sigma}_t$ should resemble $\mathbf{v}_i$ of $\mathbf{\Sigma}_p$ in direction and the singular value λ_i of $\mathbf{\Sigma}_t$ should also be close to ω_i of $\mathbf{\Sigma}_p$.

5.3.1.2 Simple features for sparse representation

For simplicity, we indicate the i-th training sample as $\mathbf{t}_i$. Applying SVD to the covariance matrix, we get $\mathbf{\Sigma}_{t_i} = \mathbf{U}_i\mathbf{\Lambda}_i\mathbf{U}_i^T$, where $\mathbf{U}_i = [\mathbf{u}_i^1, \ldots, \mathbf{u}_i^n]$ and $\mathbf{\Lambda}_i = \texttt{diag}(\rho_i)$ with $\rho_i = [\lambda_i^1, \ldots, \lambda_i^n]^T$. Note that $\mathbf{u}_i^j \in \mathbb{R}^n$ and λ_i^j stand for the j-th singular vector (principle component) and the j-th singular value of $\mathbf{t}_i$ respectively. We denote $\mathbf{B}^j = [\mathbf{u}_1^j, \mathbf{u}_2^j, \ldots, \mathbf{u}_N^j] \in \mathbb{R}^{n \times N}$ as the dictionary containing the j-th singular vectors extracted from all $\mathbf{t}_i$ with $\|\mathbf{u}_i^j\|_2 = 1$, for $i = 1, \ldots, N$.

Given a query sample $\mathbf{x}$ and corresponding $\mathbf{\Sigma}_x = \mathbf{V\Omega V}^T$, denote the j-th singular vector of $\mathbf{x}$ as $\mathbf{v}^j$ and let $\eta = [\omega_1, \ldots, \omega_n]^T$ be the vector containing all the singular values in Ω sorted in the descending order. A simple strategy for classifying $\mathbf{x}$ is to treat a particular $\mathbf{v}^j$ as the feature of $\mathbf{x}$ and employ SRC [11] to identify the feature by solving

$$\alpha^j = \arg\min_{\alpha^j} \|\alpha^j\|_1 \quad \text{subject to} \quad \mathbf{B}^j\alpha^j = \mathbf{v}^j, \tag{5.2}$$

Obtaining $\alpha^j \in \mathbb{R}^N$, $\mathbf{x}$ can be classified by evaluating the class-wise reconstruction error based on $\mathbf{B}^j$.

The above strategy using one singular vector (e.g., the top one) may work properly with well-separated data. However, real-world gesture recordings are always vulnerable to noise or large variability among individuals. Therefore it is desirable to take into account several most important singular vectors to improve the robustness of the algorithm. In addition, the discriminative information within the singular values should also be exploited.

5.3.1.3 Robust features for sparse representation

Consider a robust feature vector constructed by unifying the top s singular values and the associated singular vectors ($s \leq n$). Suppose that we have obtained α^j by solving Eq. (5.2), for all $j = 1, \ldots, s$. Without violating the equality in the constraint of Eq. (5.2), we can equivalently rewrite $\mathbf{B}^j\alpha^j = \mathbf{v}^j$ as

$$\hat{\mathbf{B}}^{\mathbf{j}}\hat{\alpha}^j = [\frac{\lambda_1^j}{\|\rho_1\|_2}\mathbf{u}_1^j, \frac{\lambda_2^j}{\|\rho_2\|_2}\mathbf{u}_2^j, \ldots, \frac{\lambda_N^j}{\|\rho_N\|_2}\mathbf{u}_N^j]\hat{\alpha}^j = \frac{\omega^j}{\|\eta\|_2}\mathbf{v}^j \tag{5.3}$$

where $\hat{\alpha}^j = \Delta\alpha^j$ with $\Delta = \texttt{diag}([\frac{\omega^j\|\rho_1\|_2}{\lambda_1^j\|\eta\|_2}, \ldots, \frac{\omega^j\|\rho_N\|_2}{\lambda_N^j\|\eta\|_2}])$. Applying the same procedure to each pair of $\mathbf{B}^j$ and $\mathbf{v}^j$ for all $j = 1, \ldots, s$, we get

$$\begin{aligned} \hat{\mathbf{B}}^1\hat{\alpha}^1 &= \frac{\omega^1}{\|\eta\|_2}\mathbf{v}^1 \\ \hat{\mathbf{B}}^2\hat{\alpha}^2 &= \frac{\omega^2}{\|\eta\|_2}\mathbf{v}^2 \\ \ldots &= \ldots \\ \hat{\mathbf{B}}^s\hat{\alpha}^s &= \frac{\omega^s}{\|\eta\|_2}\mathbf{v}^s \end{aligned} \tag{5.4}$$

Ideally, if $\mathbf{x}$ is sufficiently similar to $\mathbf{t}_i$, $\mathbf{v}^j$ should resemble $\mathbf{u}_i^j$, so should ω^j and λ_i^j for all $j = 1, \ldots, s$. Therefore, in reconstructing each $\mathbf{v}^j$, the $\mathbf{u}_i^j$ of $\mathbf{t}_i$ should be coded with large coefficient. In other words, if each $\mathbf{u}_i^j$ of $\mathbf{t}_i$ contributes most in representing $\mathbf{v}^j$ of $\mathbf{x}$, $\mathbf{t}_i$ should be similar to $\mathbf{x}$. Then, the class to which $\mathbf{t}_i$ belongs should yield the minimum error in reconstructing $\mathbf{x}$, which indicates that $\mathbf{x}$ is of the same label as $\mathbf{t}_i$.

Motivated by this intuition, we enforce each $\mathbf{v}^j$ of $\mathbf{x}$ to be represented via a universal sparse code α over the corresponding $\hat{\mathbf{B}}^j$. By substituting $\hat{\alpha}^j$ with α for all $j = 1, \ldots, s$, Eq. (5.4) can thus be simplified as

$$[\hat{\mathbf{B}}^{1^T}, \hat{\mathbf{B}}^{2^T}, \ldots, \hat{\mathbf{B}}^{s^T}]^T\alpha = [\frac{\omega^1}{\|\eta\|_2}\mathbf{v}^{1^T}, \frac{\omega^2}{\|\eta\|_2}\mathbf{v}^{2^T}, \ldots, \frac{\omega^s}{\|\eta\|_2}\mathbf{v}^{s^T}]^T, \tag{5.5}$$

where $[\hat{\mathbf{B}}^{1^T}, \hat{\mathbf{B}}^{2^T}, \ldots, \hat{\mathbf{B}}^{s^T}]^T$ is a vertical concatenation of all the sub-matrices $\hat{\mathbf{B}}^j$ and the right hand side is a super-vector by concatenating all $\mathbf{v}^j$. Thus the classification scheme based on unifying the top s pairs of singular values/vectors can be formulated as

$$\alpha = \arg\min_{\alpha} \|\alpha\|_1 \quad \text{subject to} \quad \text{Eq. (5.5)}, \tag{5.6}$$

where columns in $[\hat{\mathbf{B}}^{1^T}, \hat{\mathbf{B}}^{2^T}, \ldots, \hat{\mathbf{B}}^{s^T}]^T$ are normalized to unit ℓ_2-norm.

Definition 1 (CovSVDK)**.** *Given an MTS* $\mathbf{t}$*, its covariance matrix is decomposed as* $\mathbf{\Sigma}_t = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$ *by SVD, where* $\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_n]$ *is a singular vector matrix with orthonormal columns and* $\mathbf{\Lambda} = \texttt{diag}(\rho)$ *with* $\rho = [\lambda_1, \ldots, \lambda_n]^T$ *is a diagonal matrix with singular values descendingly sorted on the main diagonal. The CovSVDK feature for* $\mathbf{t}$ *is defined as*

$$\phi(\mathbf{t}) = \left[\frac{\lambda_1}{\parallel \rho \parallel_2}\mathbf{u}_1^T, \frac{\lambda_2}{\parallel \rho \parallel_2}\mathbf{u}_2^T, \ldots, \frac{\lambda_s}{\parallel \rho \parallel_2}\mathbf{u}_s^T\right]^T \in \mathbb{R}^{sn}, \tag{5.7}$$

where s *subjects to*

$$s = \arg\min\left\{\frac{\sum_{i=1}^{s}\lambda_i}{\sum_{i=1}^{n}\lambda_i} \geq c\right\} \tag{5.8}$$

for a pre-selected energy threshold, c*.*

In practice, it is common to empirically set a universal s for all MTS data such that most energy is preserved within the top s singular values. The name CovSVDK stands for Covariance Matrix SVD for Kernelization.

Definition 2. *Given* s*, define* Φ *as a collection of features extracted from the training set* $\mathbf{T}$ *according to Definition 1, and write* Φ *as*

$$\Phi = [\phi(\mathbf{t}_{1,1}), \ldots, \phi(\mathbf{t}_{i,1}), \ldots, \phi(\mathbf{t}_{i,n_i}), \ldots, \phi(\mathbf{t}_{k,n_k})] \in \mathbb{R}^{sn \times N}. \tag{5.9}$$

Furthermore, define $\mathbf{y} = \phi(\mathbf{x})$ *as the feature of the query sample* $\mathbf{x}$**.**

Discussion: If we define $r = \max(mn, N)$ and denote d as the reduced dimension, PCA is of computational complexity $O(r^2 d)$ while CovSVDK is of complexity $O(n^2 dN)$. For the cases where m or N is large, $O(r^2 d) \gg O(n^2 dN)$. Thus, CovSVDK is substantially more efficient than PCA over large-scale datasets or for MTS data with long durations. More importantly, the memory usage by PCA is proportional to N^2 or $m^2 n^2$ while the memory consumption by CovSVDK is proportional to n^2. Hence, CovSVDK is also more memory efficient than PCA.

Revisiting Eq. (5.5), we can substitute $\mathbf{y}$ for $\left[\frac{\omega^1}{\|\eta\|_2}\mathbf{v}^1, \frac{\omega^2}{\|\eta\|_2}\mathbf{v}^2, \ldots, \frac{\omega^s}{\|\eta\|_2}\mathbf{v}^s\right]^T \in \mathbb{R}^{sn}$ and replace $[\hat{\mathbf{B}}^{1^T}, \hat{\mathbf{B}}^{2^T}, \ldots, \hat{\mathbf{B}}^{s^T}]^T \in \mathbb{R}^{sn \times N}$ with Φ. Finally, the classification scheme based on CovSVDK features can be derived from Eq. (5.6) as

$$\alpha = \arg\min_{\alpha} \|\alpha\|_1 \quad \text{subject to} \quad \Phi\alpha = \mathbf{y}, \tag{5.10}$$

where α is the universal sparse code for representing the $\frac{\omega^i}{\|\eta\|_2}\mathbf{v}^i$ over $\hat{\mathbf{B}}^{i^T}$ for all $i = 1, \ldots, s$.

5.3.2 Kernelizing Sparse Representation for Classification

The discrimination capability of SRC relies on the quality of the dictionary. In other words, the atoms associated to different classes must be distinguishable or separable from the perspective of ℓ_1 minimization algorithms. In some real-world applications, however, computing the sparse representation over a dictionary of original training features can yield undesirable classification results. One such example is the Iris dataset (from UCI machine learning archive). As is commonly used for analyzing the performance of various classifiers, two features for each sample, regarding pedal length and pedal width, are extracted and formed into a 2D feature vector, as shown in Fig. 5.1(a). The three classes (points in red, green and blue) are distributed closely along the same radius direction. Obviously, the extracted 2D feature vectors are sufficiently discriminative for traditional classifiers, e.g., k-Nearest-Neighbors (kNN) and Support Vector Machines (SVMs). On the other hand, SRC normalizes training samples with unit ℓ_2-norm and employs the normalized training samples as dictionary atoms[1]. As shown in Fig. 5.1(b), the atoms are located on the unit circle with severe overlapping in the middle of the point scatter. The atoms within the overlapping region

[1] Normalization is typically performed to avoid trivial solution and is reasonable in face recognition, since images of a subject under different intensity levels are still considered to be same-class. In other words, the magnitudes of feature vectors are not considered as discriminative information in face recognition.

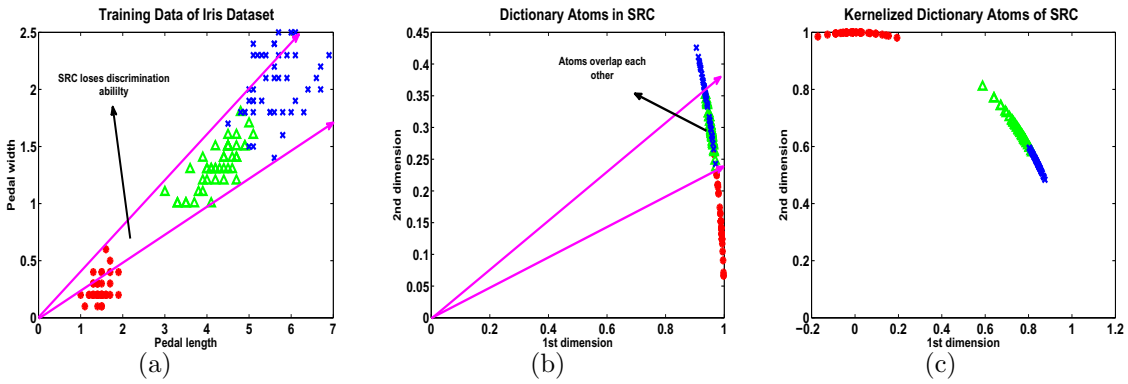


Figure 5.1: Training samples and dictionary atoms of SRC.

are inseparable and consequently cause ℓ_1 minimization algorithms the confusion in selecting the true atoms. Thus, SRC neglects the magnitude information and suffers the drawback of losing its discrimination capability in classifying data that are distributed along the same radius direction [88, 146].

We propose the kernelized sparse representation to overcome this shortcoming of SRC. This is desirable since by kernelizing sparse representation, the classification strategy of SRC can be applied to general pattern recognition tasks including MTS gesture recognition, time series classification, etc.

Kernel trick is a widely applied technique in machine learning that can adapt linear algorithms to nonlinear cases, by mapping training features $\phi(\cdot)$ from the original space $\mathcal{X}$ into some kernel space $\mathcal{F}$, in which the new kernel features $\psi(\cdot)$ are more separable for a certain type of classifiers and the nonlinear relationships among $\phi(\cdot) \in \mathcal{X}$ can be transformed into linear ones among $\psi(\cdot) \in \mathcal{F}$.

Let $\Psi = [\psi(\mathbf{t}_{1,1}), \ldots, \psi(\mathbf{t}_{i,1}), \ldots, \psi(\mathbf{t}_{i,n_i}), \ldots, \psi(\mathbf{t}_{k,n_k})]$ be the collection of training kernel features in $\mathcal{F}$. Given a test sample $\mathbf{x}$, we want to solve the sparse representation α of $\psi(\mathbf{x})$ over Ψ. However, this is typically infeasible, as 1) usually the mapping ψ is implicit, meaning that direct evaluation of the fitness term $\psi(\mathbf{x}) = \Psi\alpha$ is impossible [88]; 2) $\mathcal{F}$ may be of infinite dimension, causing that the computational complexity is intractable; 3) even though we know the mapping explicitly, $\Psi^T\Psi$ may not be invertible, resulting that the left inverse does not exist and thus no explicit

solution to $\psi(\mathbf{x}) = \Psi\alpha$ is available. To overcome these difficulties, we introduce a relaxation to the fitness constraint term as

$$\left\| \begin{bmatrix} \psi(\mathbf{x}) \\ \mathbf{0} \end{bmatrix} - \begin{bmatrix} \Psi \\ \gamma\mathbf{I} \end{bmatrix} \alpha \right\|_2 \leq \varepsilon \tag{5.11}$$

where $\mathbf{0} \in \mathbb{R}^N$ is a zero vector, $\mathbf{I} \in \mathbb{R}^{N\times N}$ is the identity matrix, ε is an arbitrarily small positive constant representing the error tolerance, γ is a small positive constant. Satisfying Eq. (5.11) is equivalent to minimizing the ridge regression problem $L(\alpha) = \|\psi(\mathbf{x}) - \Psi\alpha\|_2^2 + \gamma\alpha^T\alpha$. Setting the gradient of $L(\alpha)$ with respect to α equal to zero, the solution space of α is obtained as

$$\Psi^T\psi(\mathbf{x}) = (\Psi^T\Psi + \gamma\mathbf{I})\alpha \tag{5.12}$$

where $\Psi^T\psi(\mathbf{x})$ is an $N \times 1$ vector and $\Psi^T\Psi$ is an $N \times N$ positive semi-definite matrix. Regularized by γ, $(\Psi^T\Psi + \gamma\mathbf{I})$ is invertible, yielding that α is the global minimizer to $L(\alpha)$. In other words, enabling Eq. (5.12) is equivalent to satisfying Eq. (5.11). Thus, we can employ Eq. (5.12) as the fitness constraint in sparse coding[2].

To improve the efficiency in ℓ_1 minimization and to ensure the solution to be sparse, a random matrix $\mathbf{P} \in \mathbb{R}^{d\times N}$ obeying Gaussian or Bernoulli distribution (we use Gaussian here) is often employed to project vector $\Psi^T\psi(\mathbf{x})$ and columns in $(\Psi^T\Psi + \gamma\mathbf{I})$ into some d-dimensional random subspace, where $d \ll N$.

Define the $\mathbf{K} = \Psi^T\Psi$ as a Gram matrix, with elements $\mathbf{K}_{i,j} = k\big(\phi(\mathbf{t}_i), \phi(\mathbf{t}_j)\big)$, where $k(\cdot,\cdot)$ is a valid kernel function. By denoting $\tilde{\mathbf{y}} = \Psi^T\psi(\mathbf{x}) = k(\cdot, \mathbf{x}) \in \mathbb{R}^N$ and substituting $\mathbf{K}$ for $\Psi^T\Psi$ in the new fitness constraint Eq. (5.12), the kernelized sparse representation under random projection $\mathbf{P}$ is formulated as:

$$\alpha = \arg\min_{\alpha} \|\alpha\|_1 \quad \text{subject to} \quad \mathbf{P}(\mathbf{K} + \gamma\mathbf{I})\alpha = \mathbf{P}\tilde{\mathbf{y}}. \tag{5.13}$$

[2] Note that the proposed relaxation to fitness constraint (Eq. (5.11) and Eq. (5.12)) is a general strategy and is applicable to kernelizing other sparse coding algorithms, such as Orthogonal Matching Pursuit (OMP), but in this work we only focus on ℓ_1 minimization algorithms.

From Eq. (5.13), we can see that the linear relationship between kernel features $\psi(\mathbf{x})$ and columns in Ψ has been depicted entirely in terms of the linear combination between the kernel function values in vector $\tilde{\mathbf{y}}$ and the corresponding ones in matrix $\mathbf{K}$. For the purpose of effectively classifying MTS gestures and time series data, we further propose two kernel functions based on the CovSVDK features.

Proposition 2 (Kernel Function)**.** *Let* $\mathbf{t}$ *and* $\mathbf{p}$ *be two samples and let* $\phi(\mathbf{t})$ *and* $\phi(\mathbf{p})$ *be their extracted feature vectors. The proposed kernel function is defined as*

$$k\big(\phi(\mathbf{t}), \phi(\mathbf{p})\big) = \exp\Big\{k_L\big(\phi(\mathbf{t}), \phi(\mathbf{p})\big)\Big\} = \psi(\mathbf{t})^T\psi(\mathbf{p}) \tag{5.14}$$

where $\psi(\mathbf{t}) \in \mathcal{F}$ *and* $\psi(\mathbf{p}) \in \mathcal{F}$ *are kernel features for* $\mathbf{t}$ *and* $\mathbf{p}$*, via some implicit non-linear mapping* ψ*. In particular, for MTS data,* $\phi(\mathbf{t})$ *and* $\phi(\mathbf{p})$ *are extracted according to Definition 1 and the kernel function* $k_L(\cdot,\cdot)$ *can be written as*

$$k_L\big(\phi(\mathbf{t}), \phi(\mathbf{p})\big) = \phi(\mathbf{t})^T\phi(\mathbf{p}) = \sum_{i=1}^{s}\left(\frac{\lambda_i\omega_i}{\|\rho\|_2\|\eta\|_2}\right)\mathbf{u}_i^T\mathbf{v}_i. \tag{5.15}$$

Note that kernel features $\psi(\cdot) \in \mathcal{F}$ are of infinite dimension. By working directly on the kernel function however, we can implicitly exploit the kernel space of high, or even infinite dimension, without the need of knowing mapping ψ. By using the proposed kernel function $k(\cdot,\cdot)$, the atoms embedded in a 2D random subspace for the Iris dataset are separable for ℓ_1 minimization algorithms, as shown in Fig. 5.1(c).

By incorporating the classification rule of SRC into Eq. (5.13), we obtain the newly proposed classifier, called Kernelized SRC, which shall be discussed in the following two sections.

5.3.3 Algorithm Training Procedure

Building a discriminative dictionary is critical to the effectiveness of sparse representation based classifiers. Given a training set $\mathbf{T}$, we now describe how to construct such a dictionary via kernel trick based on specific feature extraction methods. To

Algorithm 4 Kernelized SRC: Training

Input: Training set $\mathbf{T}$

Preprocess each training sample with median filter (optional)
for $i = 1$ to k **do**
  for $j = 1$ to n_i **do**
    Feature extraction for each $\mathbf{t}_{i,j} \to \phi(\mathbf{t}_{i,j})$
    (for MTS data, $\phi(\mathbf{t}_{i,j})$ is extracted according to Definition 1)
  end for
end for
Compute $\mathbf{K}$ according to Proposition 2
Construct dictionary as $\mathbf{K} + \gamma\mathbf{I}$
Secure sparsity in the solution vector by employing $\mathbf{P}$ for dimensionality reduction (optional)
return $\mathbf{P}$ and $\mathbf{P}(\mathbf{K} + \gamma\mathbf{I})$

elaborate, we first use median filter to preprocess each sample (in the noisy case). Then we loop through all training samples to compute the features. For MTS data, the CovSVDK feature is extracted individually from each training sample. For the case of univariate time series data, we simply employ each raw time series as a feature vector, since CovSVDK is effective only when $n \geq 2$. Next, we construct a dictionary as the regularized kernel matrix $\mathbf{K} + \gamma\mathbf{I}$. Finally, we may employ a random matrix $\mathbf{P}$ to improve the efficiency in classification. The whole training process is summarized in Alg. 4.

5.3.4 Classification Rule

In this section, we discuss how to classify a query sample using the proposed Kernelized SRC. Having $\mathbf{x}$ as a test sample, we first preprocess it with the same technique as in training and extract its feature as $\mathbf{y} = \phi(\mathbf{x})$. Then based on the kernel function defined in Proposition 2, we have $\tilde{\mathbf{y}} = k(\cdot, \mathbf{x}) = [k(\phi(\mathbf{t}_1), \phi(\mathbf{x})), \ldots, k(\phi(\mathbf{t}_N), \phi(\mathbf{x}))]^T \in \mathbb{R}^N$. Next, random projection can be performed to reduce dimensionality. Then, we find the sparse representation α of $\tilde{\mathbf{y}}$ over $\mathbf{P}(\mathbf{K} + \gamma\mathbf{I})$ by solving the optimization problem Eq. (5.13), which is called Basis Pursuit Denoising (BPD) [9].

Algorithm 5 Kernelized SRC: Classification

Input: Test sample $\mathbf{x}$, random matrix $\mathbf{P}$ and dictionary $\mathbf{P}(\mathbf{K}+\gamma\mathbf{I})$

1: Preprocess test sample with median filter (optional)
2: Feature extraction for $\mathbf{x} \rightarrow \mathbf{y} = \phi(\mathbf{x})$ according to Definition 1
3: Based on the kernel function defined in Proposition 2, compute $\tilde{\mathbf{y}} = k(\cdot, \mathbf{x}) = [k(\phi(\mathbf{t}_1), \phi(\mathbf{x})), \ldots, k(\phi(\mathbf{t}_N), \phi(\mathbf{x}))]^T$
4: Random subspace embedding via $\mathbf{P}$ (optional)
5: Find the sparse coefficient vector α by solving Eq.(5.13)
6: $i = \arg\min_{i \in \{1,\ldots,k\}} \|\mathbf{P}\tilde{\mathbf{y}} - \mathbf{P}(\mathbf{K}+\gamma\mathbf{I})\delta_i(\alpha)\|_2$
7: **return** i

Notice that the sparse coefficients, α, can be computed by other fast iterative algorithms, such as Orthogonal Matching Pursuit [14] or Compressive Sampling Matching Pursuit [147]. Experimental results reported in the following sections are based on the the ℓ_1 Magic implementation of BPD [148]. Finally, we identify $\mathbf{x}$ as class i based on the decision rule as:

$$i = \arg \min_{i \in \{1,\ldots,k\}} \|\mathbf{P}\tilde{\mathbf{y}} - \mathbf{P}(\mathbf{K}+\gamma\mathbf{I})\delta_i(\alpha)\|_2, \tag{5.16}$$

where $\delta_i(\alpha) = [0, \ldots, \alpha_{i,1}, \ldots, \alpha_{i,n_i}, \ldots, 0]$. To cope with unbalanced classes, an alternative decision rule $i = \arg\min_{i \in \{1,\ldots,k\}} \frac{\|\mathbf{P}\tilde{\mathbf{y}} - \mathbf{P}(\mathbf{K}+\gamma\mathbf{I})\delta_i(\alpha)\|_2}{\|\delta_i(\alpha)\|_1}$ can be employed. The classification procedure is summarized in Alg. 5.

5.4 Experiments on Classifying Real-World MTS Data

In this section, we conduct experiments to demonstrate the promising performance of the proposed framework, *i.e.*, CovSVDK + Kernelized SRC, over three online public-access databases, *i.e.*, the Georgian-Tech Human Gait (Georgia-Tech HG) database[1], Australian Sign Language (Auslan) database[2] and High-quality Australian Sign Language (HAuslan) database[2]. The Georgia-Tech HG database was obtained via 12 video cameras; the Auslan was generated by Powergloves; and the HAuslan was generated by two 5DT gloves and two position trackers. To verify the effectiveness of the proposed CovSVDK feature, we use the linear kernel $k_L(\cdot, \cdot)$ for all experiments in this section. Feature vector $\phi(\cdot)$ for each MTS is extracted according to Definition 1.

For each particular database, the parameter s is manually selected and is consistent for all MTS data within the database. As in [11], atoms in $\mathbf{P}(\mathbf{K}+\gamma\mathbf{I})$ are normalized to unit ℓ_2-norm prior to ℓ_1 minimization. γ is set to 0.001.

We evaluate and compare the proposed CovSVDK, with Principle Component Analysis (PCA) and Linear Discriminant Analysis (LDA). For PCA and LDA, all MTS data are interpolated or downsampled to the average length, in each database. We compare the proposed classifier Kernelized SRC with two popular classifiers, *i.e.,* K-Nearest-Neighbor (KNN) with $k = 3$, Support Vector Machines (SVM) and with the coding strategy by computing the least square solution to Eq. (5.12), termed LS. For Kernelized SRC and LS, the decision rule is Eq. (5.16). For KNN and SVM, columns in Φ are employed as training data and $\phi(\mathbf{x})$ is used as the test sample. The SVM toolbox can be found at [77]. As shown in the following, our method consistently achieves high performance over these databases.

Georgia-Tech HG database

The Georgia-Tech HG database, used for human identification from a distance, is a collection of human gaits from 15 subjects. Samples of subjects were captured by cameras at 4 different controlled speeds [2]. Every subject was required to walk 9 times at every controlled speed and finally, 36 samples were obtained for every subject. A sample is a time series of gaits with varying length. By means of 22 markers on the subject, a gait is defined by 66 attributes (variables), *i.e.*, the 3-D coordinates of those markers [149, 150]. The evaluation uses all the 540 samples in the database. Among the 36 samples per subject, 30 samples are randomly collected into the training set while the remaining 6 samples are used for testing.

By transforming the kernel matrix into a low dimensional random subspace, we can reduce the computation cost of ℓ_1 minimization. In order to evaluate the effectiveness of random projection, we randomly select parts of the overall 22 markers and set

[1] Published by the Computational Perception Laboratory at Gatech at http://www.cc.gatech.edu/cpl/projects/hid/

[2] Published by UCI KDD at http://kdd.ics.uci.edu/summary.data.date.html

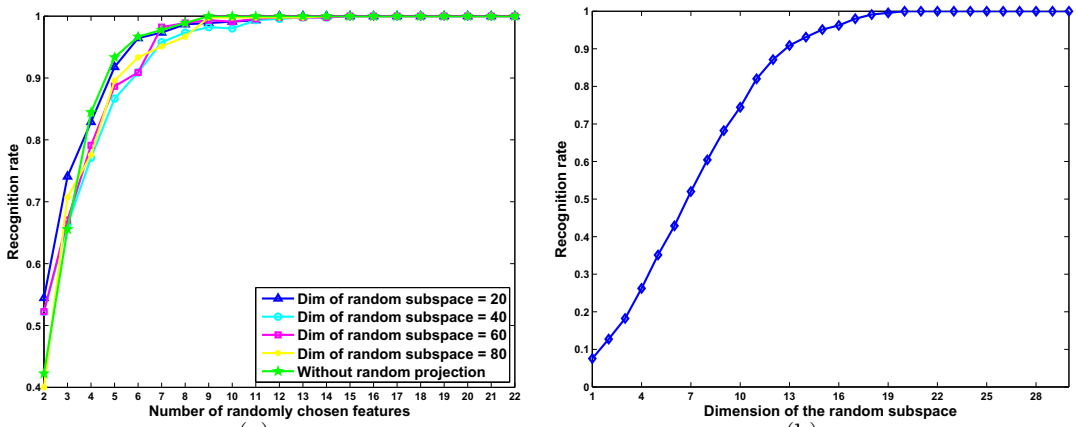


Figure 5.2: Recognition rates for the Georgia-Tech HG database. (a) 15-class problem recognition rate versus selected features (markers) under various random projections. The horizontal axis represents the number of randomly chosen features, ranging from 2 to 22. The curves in different colors represent recognition rates over 5 different random subspaces. (b) 15-class problem recognition rate versus different dimensions of the random subspace; 22 features (markers) are employed.

the parameter $s = 5$ uniformly, such that 5 singular value/vector pairs are extracted by CovSVDK for each MTS. Figure 5.2(a) indicates that the proposed approach can achieve 100% recognition rate when a random subspace is of only 20 dimensions and only 11 markers are utilized. Hence, in the following experiments over this database, kernel matrices are projected onto a random subspace with dimension 20 to improve computation efficiency.

Remark: It is worthy to point out that, for ℓ_1 minimizers, the dimensionality reduction induced by random projection is not a requisite. The purpose of embedding the dictionary atoms into some low-dimensional subspace is two-fold: 1) speed-up ℓ_1 minimization; 2) enforce the dictionary to be overcomplete such that the solution tends to be sparse. The first concern is desired from a practical efficiency perspective while the second concern is preferred by the decision rule (Eq.(5.16)) so as to secure satisfactory recognition rate. We can see from Figure 5.2(b) that the recognition rate increases as the dimension of the random subspace becomes higher. For completeness, we also evaluate the proposed approach over the Georgia-Tech HG database without performing

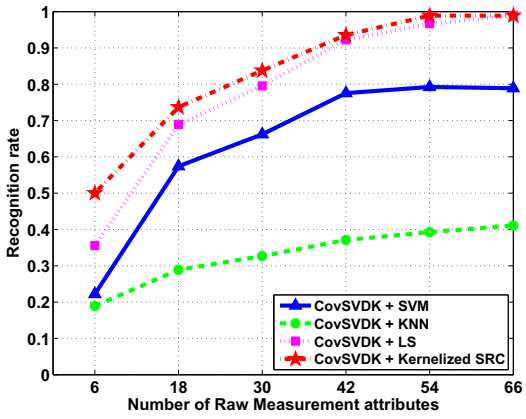
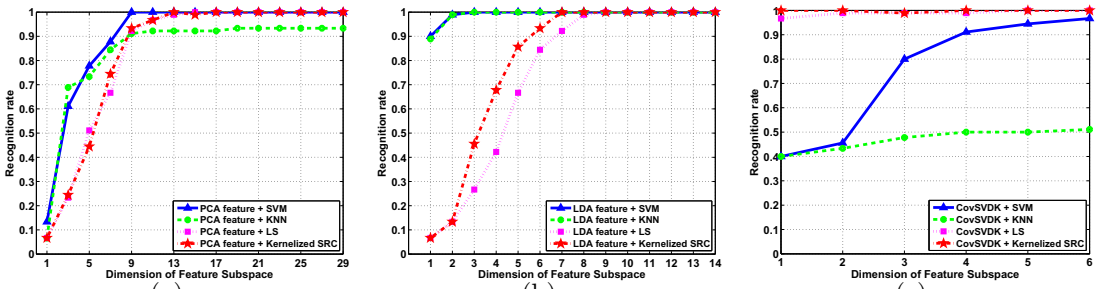


Figure 5.3: Recognition rate for various methods over the Georgia-Tech HG database.



(a) (b) (c)

Figure 5.4: Recognition rate on the Georgia-Tech HG database. (a) PCA feature (b) LDA feature (c) CovSVDK feature (proposed method). All three feature extraction methods are fed to four classifiers, i.e., SVM, KNN, LS, the proposed Kernelized SRC.

dimensionality reduction. Figure 5.2(a) illustrates that the accuracy obtained without dimensionality reduction is similar to those with dimensionality reduction.

To evaluate the proposed framework in a more challenging scenario, we down-sample the raw gesture data into 1/5 of its original length and utilize only part of the overall 66 attributes. As shown in Figure 5.3, our method robustly achieves 98.9% recognition, leading SVM by approximately 10% in accuracy.

As shown in Figure 5.4, at 9, 4, and 1 dimension(s) of the feature subspace respectively, PCA, LDA and CovSVDK achieve 100% recognition rate. Therefore, compared with PCA and LDA, the proposed CovSVDK is more effective in preserving discriminative information for classification. Finally, Table 5.1 shows that in classifying MTS data, the proposed linear kernel function $k_L(\cdot,\cdot)$ significantly outperforms

Table 5.1: Comparison among different kernel functions over the Georgia-Tech HG database.

Database	Proposed k_L	Exponential	Poly.(d = 3)	Gaussian
Gait	100%	92.2%	85.6%	80.4%

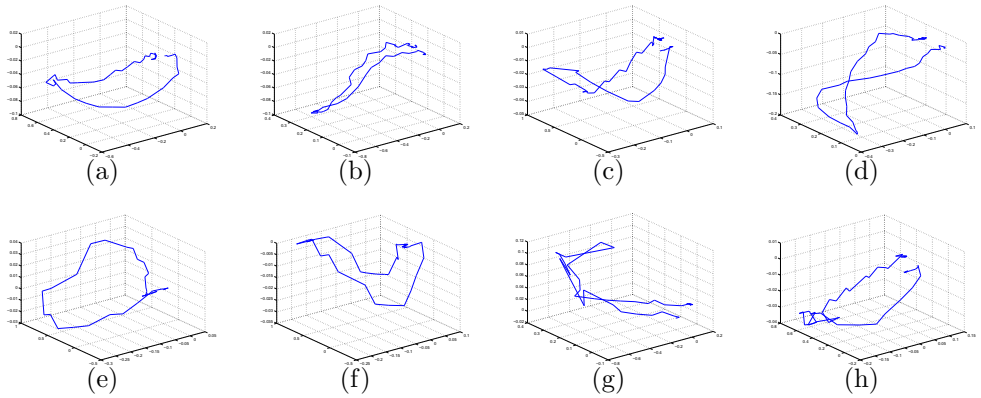


Figure 5.5: 3D trajectories for 8 signs. (a) Eat, (b) Exit, (c) Forget, (d) Give (e) Hello, (f) Know, (g) Love (h) No.

three other popular kernel functions, *i.e.*, exponential, polynomial and Gaussian kernel functions.

Australian Sign Language (Auslan) database

Contributed by 5 individual signers, the Auslan database contains 95 one-hand signs. 70 samples were collected for each sign and a sample is comprised of varying-length time series for a single hand gesture. There are 15 attributes or features for each gesture, *i.e.*, the x, y and z coordinates of the palm, the angles (roll, pitch and yaw) of the palm, the bend values of the 5 fingers and 4 additional setting values. Over this database, we conduct comparative study by evaluating the proposed approach (CovSVDK + Kernelized SRC) against several state-of-the-art algorithms, *i.e.,* discriminative mixture learning (MixCML [2]), Dynamic Time Warping (DTW) [141], Fourier Descriptors [3] and SRC [11]. Recognition rates are cited from literature for the first three methods. Results for SRC are reported based on our own implementation.

In the first experiment over the Auslan database, we consider a binary classification task. With the same experiment setup as [2], we form a subset by using 10 signs

Table 5.2: Binary Classification comparison among various methods over the Auslan database. Recognition rates with $*$ are cited from [2].

Method	Training Set	Test Set	Recognition Rate
Proposed	36	4	96.3%
MixCML [2]	39	1	95.5%$*$
DTW [141]	39	1	88%$*$

Table 5.3: Binary classification result over the Auslan database for various selection of attributes.

Method	Selected Attributes	Recognition Rate
Proposed	$1^{th}-4^{th}$, $7^{th}-10^{th}$	96.3%
	$1^{th}-6^{th}$	94.5%
	$1^{th}-4^{th}$	96.3%
	$7^{th}-10^{th}$	70.0%
	$1^{th}-3^{th}$	96.3%

and choose, from the 15 attributes, 8 attributes, namely the x, y and z coordinates of the palm, the roll angle of the palm, the bend values of the fingers of thumb, fore, index and ring.

For each of the 10 signs, *i.e.*, "eat", "exit", "forget", "give", "hello", "know", "love", "no", "sorry" and "yes", we select approximately 4 samples from each signer. Conducting 10-fold cross-validation yields a training set of 36 samples (18 per sign) and a test set of 4 samples. The proposed framework is compared with MixCML [2] and DTW [141], and the results are listed in Table 5.2. For completeness, the proposed method is further examined by performing binary classification over various selection of attributes. The results are summarized in Table 5.3. Consistent with the argument made by Kim and Pavlovic [2], our observation also reveals that the $7^{th}-10^{th}$ attributes are less discriminative than others as they only provide the finger flexion information.

In literature, we notice that this database has been widely applied to evaluate spatial trajectory recognition algorithms. In the second experiment, for fair comparison, we only keep 3 attributes, *i.e.*, x, y and z coordinates. Figure 5.5 gives some examples for 8 signs. Using the same CovSVDK features, we first compare two classifiers *i.e.,* Kernelized SRC and SRC based on 10-fold cross-validation. Then keeping

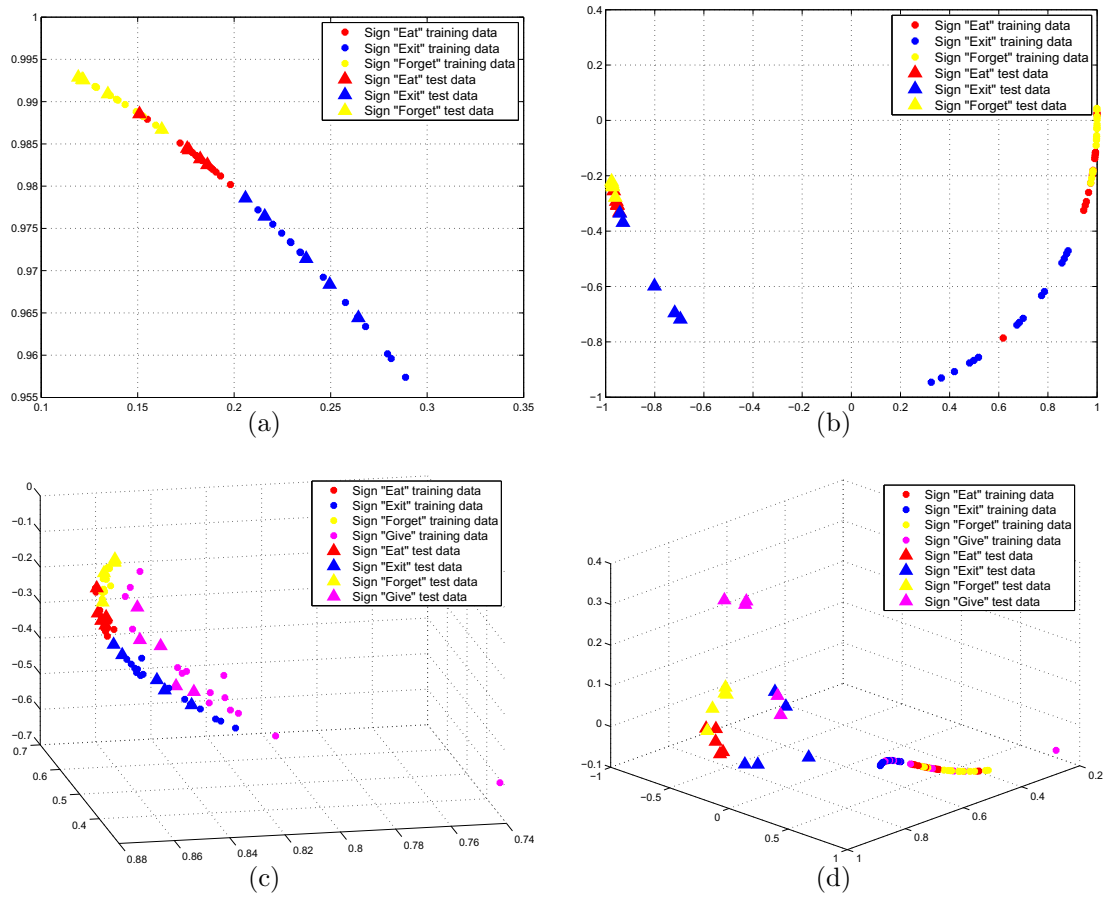


Figure 5.6: Illustrations of manifolds in multi-class classification tasks. Top row: the 3-label task; bottom row: the 4-label task. (a) 2D manifold with the kernel trick, (b) 2D manifold without the kernel trick, (c) 3D manifold with the kernel trick, (d) 3D manifold without the kernel trick.

the experiment setup consistent as in [3], the proposed approach (CovSVDK + Kernelized SRC) is compared with DTW [141] and Fourier Descriptor [3] based on 2-fold cross-validation. Classification results for aforementioned methods are summarized in Table 5.4, which indicates that the proposed algorithm is competitive among these advanced trajectory recognition algorithms.

The effectiveness of Kernelized SRC is illustrated in Figure 5.6, in which, for better visualization, 15 samples per sign are utilized for training while the remaining 5 samples are for testing. The 2/3D manifolds are obtained by projecting the dictionaries (with and without the kernel trick) into random subspace. Clearly, with kernel

Table 5.4: Multi-class Classification comparison among various methods over the Auslan database. Recognition rates with $*$ are cited from [3]. Proposed 1 is based on 10-fold cross-validation; For proposed 2, the data pool is divided into 2 folds, *i.e.*, one fold for training and the other fold for test, according to [3].

Method	Train set : Test set	Classes			
		2	3	4	8
Proposed 1	0.9 : 0.1	96.3%	93.3%	90.6%	80.0%
SRC [11]	0.9 : 0.1	78.5%	73.3%	70.9%	63.0%
Proposed 2	0.5 : 0.5	96.0%	92.7%	88.0%	75.4%
DTW [141]	0.5 : 0.5	89.8%$*$	N/A	83.8%$*$	75.9%$*$
Fourier Descriptor [3]	0.5 : 0.5	82.1%$*$	N/A	63.7%$*$	52.3%$*$

trick, samples from different classes are more separable than those without the kernel trick, which reveals that the proposed classifier is more robust than SRC [11] when dealing with cluttered data. **High-quality Australian Sign Language (HAuslan) database**

The HAuslan database consists of 95 two-hand signs. Compared with the Auslan database, the number of samples per sign is reduced to 27 and the number of attributes is increased to 22, (11 attributes for each hand). The 11 attributes for one hand are the same as those in Auslan database excluding the 4 setting values.

First, to illustrate the capability of our method in classifying large-scale databases, all 95 sign classes are used. Since the HAuslan database contains much more classes but fewer samples per class than previous two databases, 24 randomly selected samples are assigned to training set for each sign, while the remaining 3 samples are collected into the test set. Note that the kernel matrix contributed by all training samples is of size 2280×2280, to which performing ℓ_1 minimization is computationally expensive. For efficient classification, we employ random projection to reduce the row dimension of the kernel matrix to 40, which is just 1.8% of its original size. In addition, considering that the subtle differences among some signs, we set $c = 99.9\%$ so as to involve sufficient gesture details to enable effective classification. To improve robustness and remove outlier atoms from the dictionary, we apply a refinement process to the dictionary by only preserving the atoms with large reconstruction coefficients, based on the

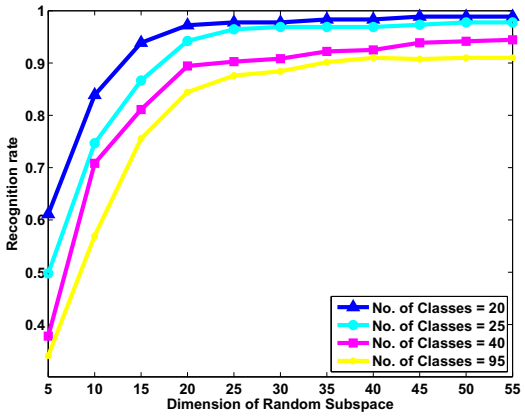


Figure 5.7: Recognition rate for the HAuslan Database.

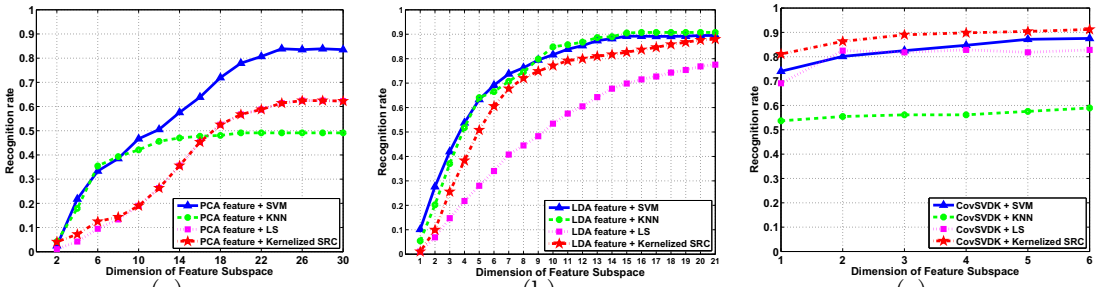
Table 5.5: Recognition rate on the HAuslan database. The dimension of random subspace is fixed at 40 for all the classification tasks.

Classes:samples	20:540	25:675	40:1080	95:2565
Recognition rate	98.2%	97.6%	94.3%	91.2%

solution to Eq. (5.13). Then, the newly formed sub-dictionary is fed to the classifier. The recognition rates of the proposed framework (CovSVDK + Kernelized SRC) are presented in Table 5.5 and in Figure 5.7.

Next, we compare CovSVDK + Kernelized SRC with various combinations of feature extraction strategies and classifiers. For CovSVDK, we set the parameter $s_{max} = 6$ and for PCA, we set the energy preservation ratio $c_{max} = 99.9\%$, which results in a maximal 30 features. The maximal number of linear features for LDA is 21. Figure 5.8 shows that although Kernelized SRC using PCA and LDA features yields inferior performance to SVM[3], when working jointly with CovSVDK, Kernelized SRC outperforms other combinations of features and classifiers. This result confirms the effectiveness of the proposed framework. The highest recognition rates and the corresponding dimensions of feature space for various methods are summarized in Table 5.6. As shown in Table 5.7, in classifying MTS data, the proposed kernel function $k_L(\cdot,\cdot)$ again significantly outperforms three other widely used kernel functions, *i.e.*,

[3] This is due to the fact that Kernelized SRC uses the simplest linear kernel while SVM employs the more advanced RBF kernel.



(a) (b) (c)

Figure 5.8: Recognition rate over the HAuslan database. (a) PCA feature (b) LDA feature (c) CovSVDK feature (proposed method). All three feature extraction methods are fed to four classifiers, i.e., SVM, KNN, LS, the proposed Kernelized SRC.

Table 5.6: Recognition performance on the HAuslan database.

Methods	Proposed	PCA+SVM	LDA+SVM	LDA+KNN
Features	6	28	18	18
Accuracy	91.2%	83.4%	90.0%	90.4%

Table 5.7: Comparison among different kernel functions over the HAuslan database.

Database	Proposed k_L	Exponential	Poly. (d = 3)	Gaussian
HAuslan	91.2%	76%	75.8%	78.9%

Table 5.8: Comparison of recognition rate among various methods over the HAuslan database. Note that recognition rates with $*$ are cited from references.

Method	Proposed	Li [139]	2dSVD [151]	SegSVD [152]
Accuracy	97.6%	89.0%*	95.0%*	93.9%*

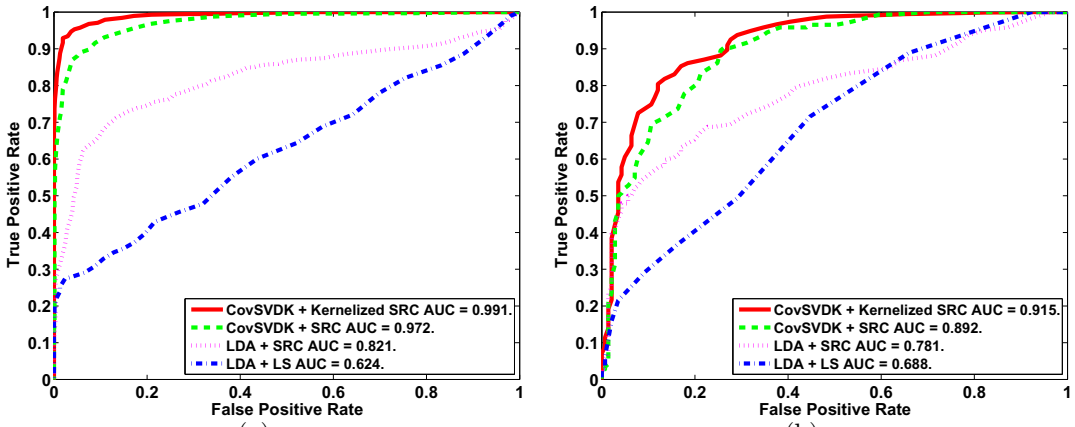
exponential, polynomial and Gaussian kernel functions.

Finally, a comparison among state-of-the-art methods in the 25-label classification problem is given in Table 5.8, which further validates the superiority of the proposed method.

Evaluating the Robustness

In this section, we evaluate the robustness of the proposed framework by employing the Sparsity Concentration Index (SCI) [11] to detect outliers. The SCI is defined as [11]

$$SCI(\alpha) = \frac{k \cdot \max_i \|\delta_i(\alpha)\|_1 / \|\alpha\|_1 - 1}{k - 1}, \tag{5.17}$$



(a) (b)

Figure 5.9: ROC curves for outlier detection over the Georgia-Tech HG and the HAuslan databases. (a) the Georgia-Tech HG database, (b) the HAuslan database. CovSVD means feature extraction following Definition. 1 and Definition. 2.

where α is the solution to Eq. (5.13) and $\delta_i(\alpha)$ is the characteristic function defined in Eq. (5.16). If a test sample can be entirely expressed by the training samples from only a single class, then $SCI(\alpha) = 1$; while, in the other extreme, if the coefficients in α spread evenly over the classes, then $SCI(\alpha) = 0$. The intuition lies in the fact that, for a test sample belonging to a certain class in the training set, the large sparse coefficients should be mostly concentrated on the same-class training samples and therefore yield an SCI that approaches 1. On the other hand, if the test sample is an irrelevant outlier, then its sparse coefficients should spread almost evenly across the whole training set and yield an SCI close to 0. Thus, the outlier detection criterion [11] is established, by setting a threshold $\tau \in (0, 1)$, where a test sample is rejected as outlier if $SCI(\alpha) < \tau$.

We verify the robustness of the proposed method over the Georgia-Tech HG and the HAuslan databases. As recommended in [11], we incorporate approximately half of all the classes into the training set but keep the test set containing samples from all the classes. Thus almost half of the test set are considered as irrelevant outliers with respect to the dictionary. For the two databases, the number of classes employed in the training set are 8 and 48 respectively. We test the performance of the proposed algorithm (CovSVDK + Kernelized SRC) by ranging τ from 0 to 1 with

0.01 step size. The resultingt Receiver Operator Characteristic (ROC) curves, (Figure 5.9), indicate that: 1) the proposed CovSVDK outperforms classical LDA in outlier detection; 2) Kernelized SRC demonstrates improved robustness compared to SRC; and 3) the Area Under Curve (AUC) of the proposed framework exceeds the AUC of other listed approaches.

5.5 Experiments on Classifying Univariate Time Series Data

In this section, we evaluate the proposed classifier Kernelized SRC with non-linear kernel function $k(\cdot, \cdot)$ over 20 datasets (data1) from UCR Time-Series Repository [6]. Raw time series are directly treated as feature vectors $\phi(\cdot)$ without using CovSVDK, which is effective only when $n \geq 2$[4]. The regularization parameter γ is set to 0.001. All columns in $\mathbf{P}(\mathbf{K} + \gamma\mathbf{I})$ are normalized to unit ℓ_2-norm prior to sparse coding. The dictionary employed is the kernel matrix with compression rates $\{\frac{d}{N} = 0.10, 0.25, 0.50, \text{none}\}$ induced by random projection, where none means no dimensionality reduction. The best result from the four cases is reported.

We compare Kernelized SRC with state-of-the-art time series classifiers, *i.e.*, 1NN-Best Warping Window DTW [4], Time Series based on a Bag-of-Features representation (TSBF) [5], as well as 7 classic classifiers[5]. The error rates of all methods are listed in Table 5.9, from which we can see that Kernelized SRC leads other algorithms by yielding the lowest error rate in 7 out of the 20 datasets. In particular, we visualize the accuracy scatter plot between Kernelized SRC and 1NN-Best Warping Window DTW [4], which is considered one of the best time series classifiers. As shown in Figure 5.10, the proposed classifier slightly outperforms 1NN-Best Warping Window DTW in 11 out of 20 datasets.

[4] To avoid the similarity values out of range, a normalizing $\phi(\cdot)$ to unit ℓ_2-norm or dividing the matrix entries by N is needed. We choose the former strategy in this work.

[5] The information regarding classic machine learning algorithms is summarized in http://www.cs.ucr.edu/~eamonn/time_series_data/WekaOnTimeSeries.xls

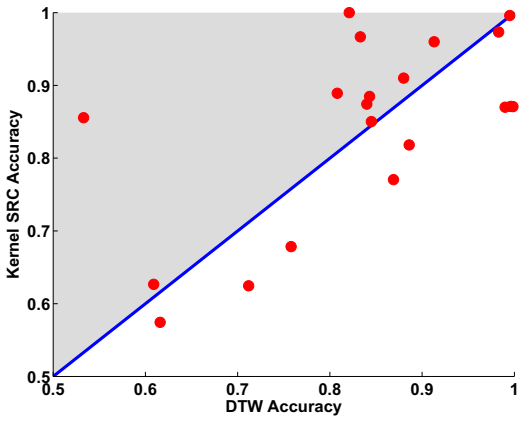


Figure 5.10: Accuracy scatter plot between Kernelized SRC and 1NN-Best Warping Window DTW [6]. Each dot represents a dataset. Dots above the diagonal mean that Kernelized SRC is better than 1NN-Best Warping Window DTW and vice versa. The farther away a dot is from the diagonal, the greater the accuracy improvement achieved [7].

In addition, to fully justify the effectiveness of the proposed kernelization strategy, we test SRC over the 20 datasets and compare it with Kernelized SRC by visualizing the accuracy scatter plot. Figure 5.11(a) shows that using kernel trick significantly improves the classification performance, as Kernelized SRC outperforms SRC in 19 out of 20 datasets. Moreover, a classifier is useful only if we can predict ahead of time on which datasets it will generate higher accuracy. We therefore perform further experiments to verify the reliability of Kernelized SRC by evaluating the expected accuracy gain versus the actual accuracy gain [8]. To acquire the expected accuracy gain, we conduct leave-one-out cross-validation within the training set for both algorithms. The gain is calculated as [8] $g = \frac{\text{Accuracy Kernelized SRC}}{\text{Accuracy SRC}}$. As depicted in Figure 5.11(b), 19 out of the 20 dots are in region TP with the remaining 1 in region TN, which indicates that the performance of Kernelized SRC is completely predictable over the 20 datasets. From the same figure, we also observe that a remarkable 20% or even higher performance increase compared to SRC is achieved via kernelization over a majority of the datasets. The impressive results validate that the proposed Kernelized SRC is very effective for time series classification.

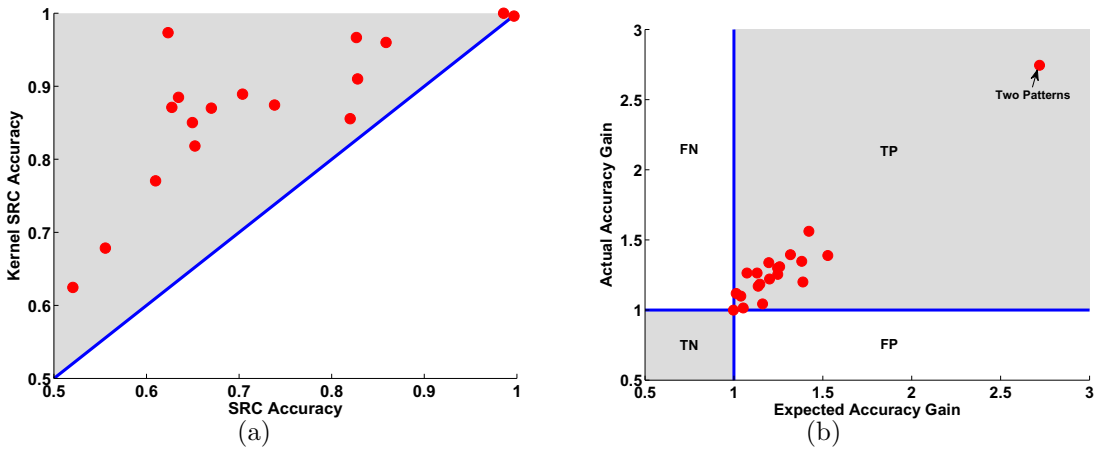


Figure 5.11: Comparison between Kernelized SRC and SRC. (a) accuracy scatter plot; (b) expected accuracy gain versus actual accuracy gain. Note that regions marked as TP/TN represent we correctly predict Kernelized SRC is better/worse than SRC; region FN means that we predict Kernelized SRC is worse than SRC but the fact is the opposite; region FP means that we predict Kernelized SRC is better than SRC but the fact is the opposite. Practically, only FP is the truly bad case [8].

5.6 Conclusion

In this chapter, we propose a novel sparse representation based framework for classifying complicated human gestures captured as multi-variate time series (MTS). First, we propose a feature extraction strategy, called CovSVDK, which is invariant to inconsistent lengths and temporal disorder across MTS data, robust to variability within human gestures, and efficient to compute. In addition, we propose a new approach to kernelize sparse representation by introducing a relaxation to the fitness constraint. This technique is generic and can be applied to kernelizing other sparse coding algorithms. Using this technique, we derive an algorithm called Kernelized SRC, which can be applied to classifying MTS data and univariate time series. Extensive experiments, including 3 MTS datasets from UCI Machine Learning Archive and 20 benchmarks from UCR Time Series Repository, confirm the effectiveness of the proposed framework.

Table 5.9: Classification results on UCR Time-Series Repository. Note that DTW* [4] means 1NN-Best Warping Window DTW and TSBF* [5] represents Time Series based on a Bag-of-Features representation with the optimal parameter setting $z = 0.25$. Results for compared methods are cited from references.

	Knn	NB	$C4_5$	MLP	RandForest	LMT	SVM	DTW* [4]	TSBF* [5]	Kernelized SRC
50words	35.60%	43.74%	58.24%	33.63%	44.84%	43.08%	35.38%	24.20%	**19.10%**	32.16%
Adiac	40.66%	43.22%	46.80%	**25.06%**	42.20%	27.88%	56.01%	39.10%	28.60%	37.34%
Beef	40.00%	50.00%	43.33%	26.67%	50.00%	20.00%	33.33%	46.70%	35.00%	**14.44%**
CBF	15.00%	10.33%	32.67%	14.67%	16.44%	23.00%	12.33%	**0.40%**	0.50%	12.89%
Coffee	25.00%	32.14%	42.86%	3.57%	25.00%	**0.00%**	3.57%	17.90%	0.40%	**0.00%**
ECG200	11.00%	23.00%	28.00%	16.00%	19.00%	18.00%	19.00%	12.00%	13.80%	**9.00%**
FaceAll	31.36%	30.83%	44.97%	17.57%	39.05%	24.26%	28.17%	19.20%	21.70%	**11.08%**
FaceFour	12.50%	15.91%	28.41%	12.50%	21.59%	22.73%	11.36%	11.40%	**3.80%**	18.18%
Fish	21.71%	33.14%	40.00%	16.00%	20.57%	18.29%	14.86%	16.00%	**7.10%**	12.57%
Gun Point	8.00%	21.33%	22.67%	6.67%	10.67%	20.67%	20.00%	8.70%	**1.10%**	4.00%
Lighting2	19.67%	32.79%	37.70%	26.23%	21.31%	36.07%	27.87%	**13.10%**	24.90%	22.95%
Lighting7	36.99%	35.62%	45.21%	35.62%	43.84%	35.62%	**28.77%**	28.80%	30.70%	37.54%
OliveOil	23.33%	23.33%	26.67%	13.33%	13.33%	16.67%	13.33%	16.70%	11.30%	**3.33%**
OSULeaf	45.45%	62.81%	63.22%	55.37%	58.26%	50.83%	56.20%	38.40%	**23.30%**	42.56%
SwedishLeaf	20.32%	14.56%	34.40%	13.44%	22.24%	17.44%	15.84%	15.70%	**8.90%**	11.52%
Synthetic Control	12.00%	4.00%	19.00%	8.67%	14.00%	8.00%	7.67%	**1.70%**	1.90%	2.67%
Trace	18.00%	20.00%	26.00%	23.00%	19.00%	24.00%	27.00%	**1.00%**	2.00%	13.00%
Two Patterns	9.40%	54.33%	34.88%	10.35%	27.50%	16.78%	17.80%	**0.15%**	0.10%	12.92%
Wafer	0.60%	29.17%	1.80%	3.72%	0.68%	1.91%	4.04%	0.50%	0.40%	**0.38%**
Yoga	16.70%	45.77%	30.10%	25.50%	22.13%	28.13%	36.93%	15.50%	16.00%	**14.81%**

Chapter 6

SUMMARY

6.1 Conclusions

In this dissertation, we have explored the capability of sparse signal modeling in addressing various challenging tasks in machine learning and computer vision, which are summarized in the following.

Locality-Constrained Dictionary Learning We show that reconstructing an unobservable intrinsic manifold via a few latent landmark points can be cast, under mild conditions, as a locality constrained dictionary learning problem in the observation space. Utilizing this approach, a novel locality constrained dictionary learning (LCDL) algorithm is introduced. The LCDL algorithm identifies a compact set of landmark points that are simultaneously representational and locality–preserving. Via the landmark points, LCDL naturally embeds training and unseen data onto the intrinsic manifold. We have applied this algorithm to face recognition and demonstrate that LCDL can significantly improve the performance of NLDR algorithms by yielding a more robust low–dimensional embedding at significantly reduced computational complexity.

Discriminative Dictionary Learning LCDL is a new generic dictionary learning algorithm with analytic solution, having the advantages of low computational complexity and capable of capturing nonlinearity of data manifold. We extend LCDL by incorporating classification error into the optimization objective and apply the derived formulation to discriminative learning tasks, such as face recognition, action recognition, hyperspectral image classification, etc. We show with extensive experiments that

by imposing locality constraint, our discriminative dictionary learning algorithm can achieve very impressive performance in recognition with substantially less time cost, compared to traditional sparse coding based approaches.

Automatic Feature Learning We develop two models, *i.e.,* stacked predictive sparse decomposition and multispectral convolutional sparse coding for tissue image classification, which is a challenging problem in computer vision and has significant clinical outcomes. The models extract features in a feed-forward manner and is highly efficient, which is particular useful for processing large quantities of high-resolution biomedical images. In addition, we propose a novel framework, called sparsity constrained convolutional regression, for nuclei segmentation. Compared to many existing approaches, our method does not rely on biological prior knowledge and could be potentially applicable to segmentation tasks of other tumor types. Our study indicates that automatic feature learning can achieve very competitive classification and segmentation performance compared to dedicated systems based on biological prior knowledge. This work is a pioneering exploration in applying automatic feature learning to biomedical image analysis and achieves very promising results.

Kernel Sparse Representation We propose a generic approach to kernelizing sparse representation, such that realized dictionary atoms are more separable for sparse coding algorithms and nonlinear relationships among data are conveniently transformed into linear relationships in the kernel space. In addition, we develop a feature extractor for human gestures captured as multivariate time series. The feature extractor maps raw data into a feature space corresponding to a valid kernel. Combining the two components, we derive a unified kernel sparse representation classifier. Extensive experiments demonstrate that the proposed approach yields superior performance compared to many existing sophisticated time series classification algorithms, *e.g.,* best warping window DTW.

6.2 Future Directions and Open Questions

- LCDL solves local least-square problems and may be affected by outliers. Future research will consider incorporating a sparse outlier term to improve robustness and testing over additional datasets. One open question is how to find low-dimensional embedding without using existing dimensionality reduction algorithms by formulating a unified dictionary learning algorithm and seeking the landmark points jointly in the observation space and the low-dimensional space, when a limited number of high-dimensional observations along with their low-dimensional embeddings are available.

- Regarding the presented automatic feature learning models, future work includes further examining the performance by enlarging the training scale and stacking the model into hierarchies with the aim to learn phenotypic concepts. In tissue classification, one open question is that can we design an automatic mechanism for learning the color decomposition matrix and incorporate it into the existing models? Moreover, seeking a mathematically sound principle for the construction of hierarchical feature learning models remains an open problem in the deep learning research community.

- On kernel sparse representation for gesture recognition, the future work will be incorporating a multi-layer structure into the classification framework and using multi-kernel learning technique to model more complicated temporal variations. Since kernel engineering is an ad-hoc process and is task specific, an open question is how to formulate a mathematical measure guiding our efforts towards designing the optimal kernel.

BIBLIOGRAPHY

[1] Q. Zhang and B. Li. Discriminative k-svd for dictionary learning in face recognition. *CVPR 2010.*

[2] M. Kim and V. Pavlovic. Discriminative learning of mixture of bayesian network classifiers for sequence classification. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 268 – 275, 2006.

[3] Shandong Wu and Y.F. Li. On signature invariants for effective motion trajectory recognition. *The International Journal of Robotics Research.*, 27(8):895 – 917, 2008.

[4] Chotirat (Ann) Ratanamahatana and Eamonn J. Keogh. Making time-series classification more accurate using learned constraints. In *SDM'04*, 2004.

[5] Mustafa Gokce Baydogan, George Runger, and Eugene Tuv. A bag-of-features framework to classify time series. *IEEE Transactions on Pattern Analysis and Machine Intelligence.*, 2012. submitted for publication.

[6] E. Keogh, Q. Zhu, B. Hu, Hao. Y., X. Xi, L. Wei, and Ratanamahatana C. A. (2011). The ucr time series classification/clustering homepage. available at http://www.cs.ucr.edu/eamonn/time_series_data/.

[7] M. Vlachos, M. Hadjieleftheriou, D. Gunopulos, and E. Keogh. Indexing multidimensional time-series with support for multiple distance measures. *ACM SIGMOD*, pages 216 – 225, 2003.

[8] Gustavo E. A. P. A. Batista, Xiaoyue Wang, and Eamonn J. Keogh. A complexity-invariant distance measure for time series. In *SDM*, 2011.

[9] E. J. Candès, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Info. Theory*, 52(2):489 – 509, 2 2006.

[10] D. L. Donoho. Compressed sensing. *IEEE Trans. Info. Theory*, 52(4):1289 – 1306, 9 2006.

[11] J. Wright, A. Yang, A. Ganesh, S. Sastry, and Y. Ma. Robust face recognition via sparse representation. *IEEE Trans. PAMI*, 31(2):210 – 227, 2 2009.

[12] S. Chen, D. Donoho, and M. Saunders. Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing*, 20(1):33–61, 1998.

[13] S.G. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *Signal Processing, IEEE Transactions on*, 41(12):3397–3415, 1993.

[14] J.A. Tropp and A.C. Gilbert. Signal recovery from random measurements via orthogonal matching pursuit. *Information Theory, IEEE Trans.*, 53(12):4655 –4666, 2007.

[15] D.L. Donoho and Y. Tsaig. Fast solution of ℓ_1 -norm minimization problems when the solution may be sparse. *Information Theory, IEEE Transactions on*, 54(11):4789–4812, 2008.

[16] Tong Wu and Kenneth Lange. Coordinate descent algorithms for lasso penalized regression. *Ann. Appl. Stat*, 2008.

[17] Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Img. Sci.*, 2(1):183–202, March 2009.

[18] Honglak Lee, Alexis Battle, Rajat Raina, and Andrew Y. Ng. Efficient sparse coding algorithms. In B. Schölkopf, J. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems 19*, pages 801–808. MIT Press, Cambridge, MA, 2007.

[19] Jorge Silva, Jorge Marques, and Joo Lemos. Selecting landmark points for sparse manifold learning. In *NIPS 18*. 2006.

[20] DavidG. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[21] B. A. Olshausen and D. J. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583):607–609, June 1996.

[22] Koray Kavukcuoglu, Marc'Aurelio Ranzato, and Yann LeCun. Fast inference in sparse coding algorithms with applications to object recognition. Technical Report CBLL-TR-2008-12-01, Computational and Biological Learning Lab, Courant Institute, NYU, 2008.

[23] Koray Kavukcuoglu, Marc'Aurelio Ranzato, Rob Fergus, and Yann LeCun. Learning invariant features through topographic filter maps. In *Proc. International Conference on Computer Vision and Pattern Recognition (CVPR'09)*. IEEE, 2009.

[24] K. Jarrett, K. Kavukcuoglu, M. Ranzato, and Y. LeCun. What is the best multi-stage architecture for object recognition? In *Computer Vision, 2009 IEEE 12th International Conference on*, 2009.

[25] Koray Kavukcuoglu, Pierre Sermanet, Y-Lan Boureau, Karol Gregor, Michael Mathieu, and Yann Le Cun. Learning convolutional feature hierarchies for visual recognition. In *Advances in Neural Information Processing Systems 23*. 2010.

[26] M.D. Zeiler, D. Krishnan, G.W. Taylor, and R. Fergus. Deconvolutional networks. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, 2010.

[27] M.D. Zeiler, G.W. Taylor, and R. Fergus. Adaptive deconvolutional networks for mid and high level feature learning. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, 2011.

[28] K. Yang and C. Shahabi. A pca-based similarity measure for multivariate time series. *MMDB' 04: Proceedings of the 2nd ACM international workshop on Multimedia databases*, pages 65 – 74, 2004.

[29] C. Li, S. Q. Zheng, and B. Prabhakaran. Segmentation and recognition of motion streams by similarity search. *ACM Trans. on Multimedia Computing, Communications and Applications*, 3(3), 8 2007.

[30] K. Yang and C. Shahabi. A pca-based kernel for kernel pca on multivariate time series. *Proceedings of ICDM 2005 Workshop on Temporal Data Mining: Algorithms, Theory and Applications*, pages 149 – 156, 11 2005.

[31] Kai Yu, Tong Zhang, and Yihong Gong. Nonlinear learning using local coordinate coding. In *NIPS09*.

[32] Jinjun Wang, Jianchao Yang, Kai Yu, Fengjun Lv, T. Huang, and Yihong Gong. Locality-constrained linear coding for image classification. In *CVPR 2010*.

[33] M. Aharon, M. Elad, and A. Bruckstein. K-svd: An algorithm for designing overcomplete dictionarys for sparse representation. *IEEE Trans. Signal Processing*, 54(11), 2006.

[34] K. Engan, S.O. Aase, and J.H. Husoy. Frame based signal compression using method of optimal directions (mod). In *ISCAS '99.*

[35] Julien Mairal, Francis Bach, Jean Ponce, and Guillermo Sapiro. Online dictionary learning for sparse coding. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, 2009.

[36] Ke Huang and Selin Aviyente. Sparse representation for signal classification. In *In Adv. NIPS 2010*.

[37] Julien Mairal, Francis Bach, Jean Ponce, Guillermo Sapiro, and Andrew Zisserman. Supervised dictionary learning. In *In Adv. NIPS 2008*.

[38] I. Ramirez, P. Sprechmann, and G. Sapiro. Classification and clustering via dictionary learning with structured incoherence and shared features. In *CVPR 2010.*

[39] Zhuolin Jiang, Zhe Lin, and L.S. Davis. Learning a discriminative dictionary for sparse coding via label consistent k-svd. In *CVPR 2011.*

[40] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, pages 1096–1103, 2008.

[41] G. E. Hinton and R. R. Salakhutdinov. Reducing the Dimensionality of Data with Neural Networks. *Science*, 313(5786):504–507, July 2006.

[42] G. Hinton, Li Deng, Dong Yu, G. E. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, and B. Kingsbury. Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups. *Signal Processing Magazine, IEEE*, 29(6):82–97, November 2012.

[43] Hilton Bristow, Anders Eriksson, and Simon Lucey. Fast convolutional sparse coding. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, 2013.

[44] Pierre Sermanet, Koray Kavukcuoglu, Soumith Chintala, and Yann Lecun. Pedestrian detection with unsupervised multi-stage feature learning. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, 2013.

[45] Roberto Rigamonti, Amos Sironi, Vincent Lepetit, and Pascal Fua. Learning separable filters. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, 2013.

[46] Roberto Rigamonti and Vincent Lepetit. Accurate and efficient linear structure segmentation by leveraging ad hoc features with learned filters. In *Medical Image Computing and Computer-Assisted Intervention MICCAI 2012*. 2012.

[47] S. Roweis and L. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290:2323 – 2326, 2000.

[48] J. B. Tenenbaum, V.de Silva, and J. C. Landford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290:2319 – 2323, 2000.

[49] David L. Donoho and Carrie Grimes. Hessian eigenmaps: New locally linear embedding techniques for high-dimensional data, 2003.

[50] Mikhail Belkin and Partha Niyogi. Laplacian Eigenmaps for Dimensionality Reduction and Data Representation. *Neural Computation*, 2003.

[51] Zhenyue Zhang and Hongyuan Zha. Principal manifolds and nonlinear dimensionality reduction via tangent space alignment. *SIAM J. Sci. Comput.*, 26:313–338, January 2005.

[52] Boaz Nadler, Stéphane Lafon, Ronald R. Coifman, and Ioannis G. Kevrekidis. Diffusion maps, spectral clustering and reaction coordinates of dynamical systems. *Applied and Computational Harmonic Analysis*, 21(1):113 – 127, 2006.

[53] V. de Silva and J. B. Tenenbaum. Global versus local methods in nonlinear dimensionality reduction. In *NIPS*. 2002.

[54] J. Mairal, M. Elad, and G. Sapiro. Sparse representation for color image restoration. *IEEE TIP*, pages 53 –69, 2008.

[55] W. M. Boothby. *An Introduction to Differentiable Manifolds and Riemannian Geometry. Revised 2nd Ed.* Academic, 2003.

[56] K.C. Lee, J. Ho, and D. Kriegman. Acquiring linear subspaces for face recognition under variable lighting. *IEEE TPAMI*, 2005.

[57] Deng Cai, Xiaofei He, and Jiawei Han. Semi-supervised discriminant analysis. In *ICCV 2007.*, pages 1 –7, 2007.

[58] T. Sim, S. Baker, and M. Bsat. The cmu pose, illumination, and expression database. *IEEE TPAMI*, pages 1615 – 1618, 2003.

[59] Ron Rubinstein, Michael Zibulevsky, and Michael Elad. Efficient implementation of the k-svd algorithm using batch orthogonal matching pursuit.

[60] Jan Knopp, Mukta Prasad, Geert Willems, Radu Timofte, and Luc Van Gool. Hough transform and 3d surf for robust three dimensional classification. In *ECCV'10*.

[61] Robert Osada, Thomas Funkhouser, Bernard Chazelle, and David Dobkin. Shape distributions. *ACM Trans. Graph.*, 21(4).

[62] Dietmar Saupe and Dejan V. Vranic. 3d model retrieval with spherical harmonics and moments. In *Proceedings of the 23rd DAGM-Symposium on Pattern Recognition*, 2001.

[63] Michael Kazhdan, Thomas Funkhouser, and Szymon Rusinkiewicz. Rotation invariant spherical harmonic representation of 3d shape descriptors. In *Proceedings of the 2003 Eurographics/ACM SIGGRAPH symposium on Geometry processing.*

[64] Jian Sun, Maks Ovsjanikov, and Leonidas Guibas. A concise and provably informative multi-scale signature based on heat diffusion. In *Proceedings of the Symposium on Geometry Processing*, 2009.

[65] C. Maes, T. Fabry, J. Keustermans, D. Smeets, P. Suetens, and D. Vandermeulen. Feature detection on 3d face surfaces for pose normalisation and recognition. In *Biometrics: Theory Applications and Systems (BTAS), 2010 Fourth IEEE International Conference on*, pages 1 –6, sept. 2010.

[66] Hien Van Nguyen and F. Porikli. Concentric ring signature descriptor for 3d objects. In *Image Processing (ICIP), 2011 18th IEEE International Conference on*, pages 2893 –2896, sept. 2011.

[67] Haichao Zhang, Jianchao Yang, Yanning Zhang, N.M. Nasrabadi, and T.S. Huang. Close the loop: Joint blind image restoration and recognition with sparse representation prior. In *ICCV 2011*, pages 770 –777.

[68] Shenghua Gao, I.W. Tsang, Liang-Tien Chia, and Peilin Zhao. Local features are not lonely: Laplacian sparse coding for image classification. In *CVPR 2010*.

[69] K. Kavukcuoglu, M.A. Ranzato, R. Fergus, and Yann Le-Cun. Learning invariant features through topographic filter maps. In *CVPR 2009*.

[70] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. Discriminative learned dictionaries for local image analysis. In *CVPR 2008*, june 2008.

[71] Yi-Chen Chen, C.S. Sastry, V.M. Patel, P.J. Phillips, and R. Chellappa. Rotation invariant simultaneous clustering and dictionary learning. In *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*, pages 1053 –1056, march 2012.

[72] Y. Zhou and K. E. Barner. Locality constrained dictionary learning for nonlinear dimensionality reduction. *Signal Processing Letters, IEEE*, 20(4):335–338, April.

[73] Lijun Zhang, Chun Chen, Jiajun Bu, Deng Cai, Xiaofei He, and T.S. Huang. Active learning based on locally linear reconstruction. *PAMI, IEEE Trans.*, 33(10):2026 –2038, 2011.

[74] Gene H. Golub, Per Christian Hansen, and Dianne P. O'Leary. Tikhonov regularization and total least squares. *SIAM J. Matrix Anal. Appl.*, 21(1):185–194, October 1999.

[75] P. Sinha, B. Balas, Y. Ostrovsky, and R. Russell. Face recognition by humans: Nineteen results all computer vision researchers should know about. *Proceedings of the IEEE*, 94(11):1948 –1962, nov. 2006.

[76] SHREC'11 track: shape retrieval on non-rigid 3D watertight meshes. In *Proc. Eurographics 2011 Workshop on 3D Object Retrieval (3DOR'11)*, pages 79–88, 2011.

[77] C.-C. Chang and C.-J. Lin. Libsvm: a library for support vector machines, 2001. available at http://www.csie.ntu.edu.tw/cjlin/libsvm.

[78] Yin Zhou, Kai Liu, Jinglun Gao, and Kenneth E. Barner. High-speed structured light scanning system and 3d gestural point cloud recognition. In *Proceedings, IEEE Int. Conf. on Information Sciences and Systems*, Baltimore, USA, May 2013. to appear.

[79] Kai Liu, Yongchang Wang, Daniel L. Lau, Qi Hao, and Laurence G. Hassebrook. Dual-frequency pattern scheme for high-speed 3-d shape measurement. *Opt. Express*, 18(5):5229–5244, Mar 2010.

[80] Kai Liu, Yongchang Wang, Daniel L. Lau, Qi Hao, and Laurence G. Hassebrook. Gamma model and its analysis for phase measuring profilometry. *J. Opt. Soc. Am. A*, 27(3):553–562, Mar 2010.

[81] J. Yang, Z.Wang, Z. Lin, X. Shu, and T. Huang. Bilevel sparse coding for coupled feature spaces. In *CVPR 2012*.

[82] A. Martinez and R. Benavente. The ar face database. *CVC Tech. Report*, (24), 1998.

[83] Moshe Blank, Lena Gorelick, Eli Shechtman, Michal Irani, and Ronen Basri. Actions as space-time shapes. In *ICCV'05*.

[84] James Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. *J. Mach. Learn. Res.*, 13, March 2012.

[85] Lei Zhang, Meng Yang, and Xiangchu Feng. Sparse representation or collaborative representation: Which helps face recognition? In *ICCV 2011*.

[86] Lena Gorelick, Moshe Blank, Eli Shechtman, Michal Irani, and Ronen Basri. Actions as space-time shapes. *IEEE Trans. PAMI*, 29(12):2247–2253, December 2007.

[87] A.F. Bobick and J.W. Davis. The recognition of human movement using temporal templates. *PAMI, IEEE Trans.*, 23(3):257 –267, 2001.

[88] L. Zhang, W.-D. Zhou, P.-C. Chang, J. Liu, Z. Yan, T. Wang, and F.-Z. Li. Kernel sparse representation-based classifier. *Signal Processing, IEEE Transactions on*, 2011.

[89] F. Pacifici, Q. Du, and S. Prasad. Report on the 2013 ieee grss data fusion contest: Fusion of hyperspectral and lidar data [technical committees]. *Geoscience and Remote Sensing Magazine, IEEE*, 1(3):36–38, 2013.

[90] J.A. Benediktsson, J.A. Palmason, and J.R. Sveinsson. Classification of hyperspectral data from urban areas based on extended morphological profiles. *Geoscience and Remote Sensing, IEEE Transactions on*, 43(3):480–491, 2005.

[91] Hang Chang, Alexander Borowsky, Paul Spellman, and Bahram Parvin. Classification of tumor histology via morphometric context. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, 2013.

[92] Hang Chang, Nandita Nayak, Paul Spellman, and Bahram Parvin. Characterization of tissue histopathology via predictive sparse decomposition and spatial pyramid matching. *Medical image computing and computed-assisted intervention–MICCAI*, 2013.

[93] R. Bhagavatula, M. Fickus, W. Kelly, C. Guo, J. Ozolek, C. Castro, and J. Kovacevic. Automatic identification and delineation of germ layer components in *h&e* stained images of teratomas derived from human and nonhuman primate embryonic stem cells. In *ISBI*, pages 1041–1044, 2010.

[94] J. Kong, L. Cooper, A. Sharma, T. Kurk, D. Brat, and J. Saltz. Texture based image recognition in microscopy images of diffuse gliomas with multi-class gentle boosting mechanism. In *ICASSAP*, pages 457–460, 2010.

[95] S Kothari, JH Phan, AO Osunkoya, and MD Wang. Biological interpretation of morphological patterns in histopathological whole slide images. In *ACM Conference on Bioinformatics, Computational Biology and Biomedicine*, 2012.

[96] CH Huang, A Veillard, N Lomeine, D Racoceanu, and L Roux. Time efficient sparse analysis of histopathological whole slide images. *Computerized medical imaging and graphics*, 35(7-8):579–591, 2011.

[97] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 2169–2178, 2006.

[98] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874, 2008.

[99] L. Latson, N. Sebek, and K. Powell. Automated cell nuclear segmentation in color images of hematoxylin and eosin-stained breast biopsy. *Analytical and Quantitative Cytology and Histology*, 26(6):321–331, 2003.

[100] B. Ballaro, A. Florena, V. Franco, D. Tegolo, C. Tripodo, and C. Valenti. An automated image analysis methodology for classifying megakaryocytes in chronic myeloproliferative disorders. *Medical Image Analysis*, 12:703–712, 2008.

[101] M. Datar, D. Padfield, and H. Cline. Color and texture based segmentation of molecular pathology images using HSOMs. In *ISBI*, pages 292–295, 2008.

[102] H. Chang, R.A. Defilippis, T.D. Tlsty, and B. Parvin. Graphical methods for quantifying macromolecules through bright field imaging. *Bioinformatics*, 25(8):1070–1075, 2009.

[103] H. Fatakdawala, J. Xu, A. Basavanhally, G. Bhanot, S. Ganesan, F. Feldman, J. Tomaszewski, and A. Madabhushi. Expectation-maximization-driven geodesic active contours with overlap resolution (EMaGACOR): Application to lymphocyte segmentation on breast cancer histopathology. *IEEE Transactions on Biomedical Engineering*, 57(7):1676–1690, 2010.

[104] Y. Al-Kofahi, W. Lassoued, W. Lee, and B. Roysam. Improved automatic detection and segmentation of cell nuclei in histopathology images. *IEEE Transactions on Biomedical Engineering*, 57(4):841–852, 2010.

[105] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[106] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J.C. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[107] C. Demir and B. Yener. Automated cancer diagnosis based on histopathological images: A systematic survey. *Technical Report, Rensselaer Polytechnic Institute, Department of Computer Science.*, 2009.

[108] M. Gurcan, LE Boucheron, A. Can, A. Madabhushi, NM Rajpoot, and Y. Bulent. Histopathological image analysis: a review. *IEEE Transactions on Biomedical Engineering*, 2:147–171, 2009.

[109] D. Axelrod, N. Miller, H. Lickley, J. Qian, W. Christens-Barry, Y. Yuan, Y. Fu, and J. Chapman. Effect of quantitative nuclear features on recurrence of ductal carcinoma in situ (DCIS) of breast. *Cancer Informatics*, 4:99–109, 2008.

[110] A. Basavanhally, J. Xu, A. Madabhushu, and S. Ganesan. Computer-aided prognosis of ER+ breast cancer histopathology and correlating survival outcome with oncotype DX assay. In *ISBI*, pages 851–854, 2009.

[111] S. Doyle, M. Feldman, J. Tomaszewski, N. Shih, and A. Madabhushu. Cascaded multi-class pairwise classifier (CASCAMPA) for normal, cancerous, and cancer confounder classes in prostate histology. In *ISBI*, pages 715–718, 2011.

[112] J. Han, H. Chang, L. Loss, K. Zhang, FL Baehner, JW Gray, PT Spellman, and Bahram Parvin. Comparison of sparse coding and kernel methods for histopathological classification of glioblastoma multiforme. In *ISBI*, pages 711–714, 2011.

[113] E. Acar, GE Plopper, and B. Yener. Coupled analysis of in vitro and histology samples to quantify structure-function relationships. *PLoS One*, 7(3):e32227, 2012.

[114] CC Bilgin, S. Ray, B. Baydil, WP Daley, M. Larsen, and B. Yener. Multiscale feature analysis of salivary gland branching morphogenesis. *PLoS One*, 7(3):e32906, 2012.

[115] Anna Bosch, Andrew Zisserman, and Xavier Muñoz. Scene classification using a hybrid generative/discriminative approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(4):712–727, April 2008.

[116] O. Boiman, E. Shechtman, and M. Irani. In defense of nearest-neighbor based image classification. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008.

[117] M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image Processing*, 15(12):3736–3745, December 2006.

[118] F. Moosmann, B. Triggs, and F. Jurie. Randomized clustering forests for building fast and discriminative visual vocabularies. In *NIPS*, 2006.

[119] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html.

[120] QV Le, J Han, JW Gray, PT Spellman, AF Borowsky, and B Parvin. Learning invariant features from tumor signature. In *ISBI*, pages 302–305, 2012.

[121] Nandita Nayak, Hang Chang, Alexander Borowsky, Paul Spellman, and Bahram Parvin. Classification of tumor histopathology via sparse feature learning. In *Proc. ISBI*, pages 410–413, 2013.

[122] Marc'Aurelio Ranzato, Y-Lan Boureau, and Yann LeCun. Sparse feature learning for deep belief networks. In J.C. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 1185–1192. MIT Press, Cambridge, MA, 2008.

[123] Honglak Lee, Alexis Battle, Rajat Raina, and Andrew Y. Ng. Efficient sparse coding algorithms. In *In NIPS*, pages 801–808. NIPS, 2007.

[124] Honglak Lee, Chaitanya Ekanadham, and Andrew Y. Ng. Sparse deep belief net model for visual area v2. In *Advances in Neural Information Processing Systems 20*. MIT Press, 2008.

[125] Christopher Poultney, Sumit Chopra, and Yann Lecun. Efficient learning of sparse representations with an energy-based model. In *Advances in Neural Information Processing Systems (NIPS 2006*. MIT Press, 2006.

[126] Kai Yu, Tong Zhang, and Yihong Gong. Nonlinear learning using local coordinate coding. In Y. Bengio, D. Schuurmans, J. Lafferty, C. K. I. Williams, and A. Culotta, editors, *Advances in Neural Information Processing Systems 22*, pages 2223–2231, 2009.

[127] J.A. Tropp and A.C. Gilbert. Signal recovery from random measurements via orthogonal matching pursuit. *Information Theory, IEEE Transactions on*, 53(12):4655–4666, 2007.

[128] Andrea Vedaldi and Andrew Zisserman. Efficient additive kernels via explicit feature maps. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(3):480–492, 2012.

[129] A. Ruifork and D. Johnston. Quantification of histochemical staining by color decomposition. *Anal Quant Cytol Histology*, 23(4):291–299, 2001.

[130] Heng Huang, C. Ding, Deguang Kong, and Haifeng Zhao. Multi-label relieff and f-statistic feature selections for image annotation. *2013 IEEE Conference on Computer Vision and Pattern Recognition*, 0:2352–2359, 2012.

[131] Quoc V. Le, Jiquan Ngiam, Adam Coates, Abhik Lahiri, Bobby Prochnow, and Andrew Y. Ng. On Optimization Methods for Deep Learning. 2011.

[132] Jonathan R Shewchuk. An introduction to the conjugate gradient method without the agonizing pain. Technical report, Pittsburgh, PA, USA, 1994.

[133] Julien Mairal, Francis Bach, Jean Ponce, and Guillermo Sapiro. Online learning for matrix factorization and sparse coding. *J. Mach. Learn. Res.*, 11:19–60, March 2010.

[134] Jianchao Yang, Kai Yu, Yihong Gong, and Thomas Huang. Linear spatial pyramid matching using sparse coding for image classification. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 1794–1801, 2009.

[135] Richard A. Young and Ronald M. Lesperance. The gaussian derivative model for spatial-temporal vision. *I. Cortical Model. Spatial Vision*, 2001:3–4, 2001.

[136] Hang Chang, Gerald Fontenay, Ju Han, Ge Cong, Fredrick Baehner, Joe Gray, Paul Spellman, and Bahram Parvin. Morphometric analysis of TCGA Gliobastoma Multiforme. *BMC Bioinformatics*, 12(1), 2011.

[137] Hang Chang, L.A. Loss, P.T. Spellman, A. Borowsky, and B. Parvin. Batch-invariant nuclear segmentation in whole mount histology sections. In *Biomedical Imaging (ISBI), 2012 9th IEEE International Symposium on*, pages 856–859, May 2012.

[138] Y. Li, C. Fermuller, Y. Aloimonos, and H. Ji. Learning shift-invariant sparse representation of actions. *CVPR 2010.*

[139] C. Li, P. Zhai, S.Q. Zheng, and B. Prabhakaran. Segmentation and recognition of multi-attribute motion sequences. *Proceedings of the ACM Multimedia Conference 2004*, pages 836 – 843, 2004.

[140] Y. Yuan and K. E. Barner. Hybrid feature selection for gesture recognition using support vector machines. *IEEE Conference on ICASSP*, pages 1941 – 1944, 3 2008.

[141] H. Sakoe and S. Chiba. Dynamic programming algorithm optimization for spoken word recognition. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 26(1):43 – 49, February 1978.

[142] Anthony L. Vatavu, R.-D. and J.O. Wobbrock. Gestures as point clouds: A $ p recognizer for user interface prototypes. In *ICMI*, 2012.

[143] F.I. Bashir, A.A. Khokhar, and D. Schonfeld. Object trajectory-based activity classification and recognition using hidden markov models. *Image Processing, IEEE Transactions on*, 16(7):1912 –1919, 2007.

[144] W. J. Krzanowski. Between-groups comparison of principal components. *JASA*, 74(367):703 – 707, 1979.

[145] Shenghua Gao, Ivor Tsang, and Liang-Tien Chia. Kernel sparse representation for image classification and face recognition. In *Computer Vision ECCV 2010.*

[146] Yin Zhou, Jinglun Gao, and Kenneth E. Barner. An enhanced sparse representation strategy for signal classification. In *Proceedings, SPIE Defense, Security, and Sensing*, 2012.

[147] Deanna Needell and Joel A. Tropp. Cosamp: iterative signal recovery from incomplete and inaccurate samples. *Commun. ACM*, 53(12):93–100, 2010.

[148] E. Candès and J. Romberg. l1-magic: Recovery of sparse signals via convex programming. 2005.

[149] R. Tanawongsuwan and A. Bobick. Characteristics of time-distance gait parameters across speeds. *GVU Technical Report*, 2003.

[150] R. Tanawongsuwan and A. Bobick. Performance analysis of time-distance gait parameters under different speeds. *4th International Conference on Audio and Video Based Biometric Person Authentication*, 2003.

[151] X. Weng and J. Shen. Classification of multivariate time series using two-dimensional singular value decomposition. *Knowledge-Based Systems*, 21(7):535 – 539, 2008.

[152] J. Liu and M. Kavakli. Hand gesture recognition based on segmented singular value decomposition. In *Knowledge-Based and Intelligent Information and Engineering Systems*, volume 6277, pages 214–223. 2010.

Appendix

COPYRIGHT PERMISSIONS

Title: Locality Constrained Dictionary Learning for Nonlinear Dimensionality Reduction
Author: Yin Zhou; Barner, K.E.
Publication: IEEE Signal Processing Letters
Publisher: IEEE
Date: April 2013

Thesis / Dissertation Reuse

The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:

Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:

1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line © 2011 IEEE.
2) In the case of illustrations or tabular material, we require that the copyright line © [Year of original publication] IEEE appear prominently with each reprinted figure and/or table.
3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:

1) The following IEEE copyright/ credit notice should be placed prominently in the references: © [year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]
2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis on-line.
3) In placing the thesis on the author's university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to http://www.ieee.org/publications_standards/publications/rights/rights_link.html to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

Title:	Non-rigid 3D shape recognition via dictionary learning
Conference Proceedings:	Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on
Author:	Yin Zhou; Kai Liu; Barner, K.E.
Publisher:	IEEE
Date:	26-31 May 2013

Thesis / Dissertation Reuse

The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:

Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:

1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line © 2011 IEEE.
2) In the case of illustrations or tabular material, we require that the copyright line © [Year of original publication] IEEE appear prominently with each reprinted figure and/or table.
3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:

1) The following IEEE copyright/ credit notice should be placed prominently in the references: © [year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]
2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis on-line.
3) In placing the thesis on the author's university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to http://www.ieee.org/publications_standards/publications/rights/rights_link.html to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

Title:	Stacked Predictive Sparse Coding for Classification of Distinct Regions in Tumor Histopathology
Conference Proceedings:	Computer Vision (ICCV), 2013 IEEE International Conference on
Author:	Hang Chang; Yin Zhou; Spellman, P.; Parvin, B.
Publisher:	IEEE
Date:	1-8 Dec. 2013

Thesis / Dissertation Reuse

The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:

Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:

1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line © 2011 IEEE.
2) In the case of illustrations or tabular material, we require that the copyright line © [Year of original publication] IEEE appear prominently with each reprinted figure and/or table.
3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:

1) The following IEEE copyright/ credit notice should be placed prominently in the references: © [year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]
2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis on-line.
3) In placing the thesis on the author's university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to http://www.ieee.org/publications_standards/publications/rights/rights_link.html to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

Title:	Classification of Histology Sections via Multispectral Convolutional Sparse Coding
Conference Proceedings:	Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on
Author:	Yin Zhou; Hang Chang; Barner, K.; Spellman, P.; Parvin, B.
Publisher:	IEEE
Date:	23-28 June 2014

Thesis / Dissertation Reuse

The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:

Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:

1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line © 2011 IEEE.
2) In the case of illustrations or tabular material, we require that the copyright line © [Year of original publication] IEEE appear prominently with each reprinted figure and/or table.
3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:

1) The following IEEE copyright/ credit notice should be placed prominently in the references: © [year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]
2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis on-line.
3) In placing the thesis on the author's university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to http://www.ieee.org/publications_standards/publications/rights/rights_link.html to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

Title: Kernel-based sparse representation for gesture recognition

Author: Yin Zhou,Kai Liu,Rafael E. Carrillo,Kenneth E. Barner,Fouad Kiamilev

Publication: Pattern Recognition

Publisher: Elsevier

Date: Dec 1, 2013

Order Completed

Thank you for your order.

This Agreement between ("You") and Elsevier ("Elsevier") consists of your order details and the terms and conditions provided by Elsevier and Copyright Clearance Center.

License number	Reference confirmation email for license number
License date	Dec 31, 2014
Licensed content publisher	Elsevier
Licensed content publication	Pattern Recognition
Licensed content title	Kernel-based sparse representation for gesture recognition
Licensed content author	Yin Zhou,Kai Liu,Rafael E. Carrillo,Kenneth E. Barner,Fouad Kiamilev
Licensed content date	December 2013
Licensed content volume number	46
Licensed content issue number	12
Number of pages	15
Type of Use	reuse in a thesis/dissertation
Portion	full article
Format	electronic
Are you the author of this Elsevier article?	Yes
Will you be translating?	No
Title of your thesis/dissertation	Sparse Signal Processing for Machine Learning and Computer Vision
Expected completion date	Dec 2014
Elsevier VAT number	GB 494 6272 12
Billing Type	Invoice
Billing address	Yin Zhou 148 Evans Hall University of Delaware NEWARK, DE 19716 United States Attn: Yin Zhou
Permissions price	0.00 USD
VAT/Local Sales Tax	0.00 USD / 0.00 GBP
Total	0.00 USD

CLOSE WINDOW