Learning Multimodal Structures in Computer Vision

A Dissertation Presented for the

Doctor of Philosophy

Degree

The University of Tennessee, Knoxville

Ali Taalimi

August 2017

© by Ali Taalimi, 2017
All Rights Reserved.

To my parents and my wife.

Acknowledgments

This dissertation would not have been possible without the guidance and support of several individuals who have extended their valuable assistance in the completion of my Ph.D. studies. First, I would like to express my sincere appreciation and gratitude to my advisor, Dr. Hairong Qi, for giving me the opportunity to be part of her research group. I am especially thankful for her continuous support of my study and research, and also for her kindness, patience, motivation, and enthusiasm. Her calm and friendly attitude combined with her immense knowledge of this field made my learning process and research experience much more enjoyable. Second, I would like to thank my dissertation committee members, Dr. Dean, Dr. Djouadi, Dr. Gregor and Dr. Ostrowski for all the support, guidance, and understanding. I really appreciate their insightful comments about my dissertation and great discussions that we had about my research problem.

My friends and labmates were also an integral part of my research experience. I want to thank them all for their comments and suggestions and wish them all the best in their research and career. Last, but certainly not the least, I would like to dedicate this dissertation to my parents Shirin and Ali Mohammad, and my wife May. I could not have accomplished this without their love and constant support through tough times.

Abstract

A phenomenon or event can be received from various kinds of detectors or under different conditions. Each such acquisition framework is a modality of the phenomenon. Due to the relation between the modalities of multimodal phenomena, a single modality cannot fully describe the event of interest. Since several modalities report on the same event introduces new challenges comparing to the case of exploiting each modality separately.

We are interested in designing new algorithmic tools to apply sensor fusion techniques in the particular signal representation of sparse coding which is a favorite methodology in signal processing, machine learning and statistics to represent data. This coding scheme is based on a machine learning technique and has been demonstrated to be capable of representing many modalities like natural images. We will consider situations where we are not only interested in support of the model to be sparse, but also to reflect a-priorily known knowledge about the application in hand.

Our goal is to extract a discriminative representation of the multimodal data that leads to easily finding its essential characteristics in the subsequent analysis step, e.g., regression and classification. To be more precise, sparse coding is about representing signals as linear combinations of a small number of bases from a dictionary. The idea is to learn a dictionary that encodes intrinsic properties of the multimodal data in a decomposition coefficient vector that is favorable towards the maximal discriminatory power.

We carefully design a multimodal representation framework to learn discriminative feature representations by fully exploiting, the modality-shared which is the information shared by various modalities, and modality-specific which is the information content of

each modality individually. Plus, it automatically learns the weights for various feature components in a data-driven scheme. In other words, the physical interpretation of our learning framework is to fully exploit the correlated characteristics of the available modalities, while at the same time leverage the modality-specific character of each modality and change their corresponding weights for different parts of the feature in recognition.

Table of Contents

1 Introduction **1**

1.1 Motivation and Background 1

1.2 Sparse Representation 4

1.2.1 Variable Selection by Sparsity Regularization 6

1.2.2 Sparse Based Regularization 6

1.3 Dictionary Learning 8

1.4 Contributions and Outline 11

2 Sparse Representation Classification **14**

2.1 Introduction 14

2.1.1 Single Modal Case 15

2.1.2 Multimodal Case 17

2.2 Unsupervised Dictionary Learning 19

2.2.1 Single Modal Case 20

2.2.2 Multimodal Case 24

2.2.3 Estimate Multimodal Sparse Codes 26

2.2.4 Learn Dictionary 27

2.3 Application: HEp-2 Cell Classification 29

2.3.1 HEp-2 Background and Related Work 30

2.3.2 Sparse Codes Pooling 34

2.3.3 Dataset 36

2.3.4 Results . . . 39
2.4 Conclusion . . . 45

3 Tree-Structured Hierarchical Coding **48**
3.1 Introduction . . . 48
3.2 Tree-Structured Hierarchical Groups . . . 52
3.3 Application: Visual Tracking . . . 59
3.4 Related Works . . . 61
3.4.1 Sparse Trackers . . . 62
3.4.2 Joint Sparsity Trackers . . . 63
3.5 The Proposed Visual Tracker - MM-THM . . . 65
3.5.1 Optimization . . . 66
3.5.2 Multimodal Dictionary Learning . . . 68
3.5.3 Classification and Template Update . . . 70
3.6 Experiments and Results . . . 72
3.7 Conclusion . . . 76

4 Supervised Dictionary Learning **84**
4.1 Introduction . . . 84
4.2 Single Modal . . . 86
4.2.1 Estimation of Dictionary and Classifier: Independent . . . 86
4.2.2 Estimation of Dictionary and Classifier: Jointly . . . 89
4.3 Multimodal: All-Against-All . . . 90
4.3.1 Coupling Latent Feature Spaces . . . 91
4.4 Multimodal: One-Against-All . . . 93
4.4.1 Multimodal Weighted Dictionary Learning . . . 94
4.5 Implicitly Defined Dictionary . . . 98
4.5.1 Extension . . . 102
4.6 Optimization . . . 103

4.6.1 Algorithm . . . 106
4.7 Proof . . . 109
4.7.1 Case : M=1 . . . 109
4.7.2 Case: Multimodal with Joint Sparsity . . . 111
4.7.3 case : Multimodal with M features with Tree-Structure . . . 115
4.7.4 Multi-Task Learning of Hierarchical Structures . . . 116
4.8 Experiment . . . 118
4.8.1 Gender Classification . . . 119
4.8.2 Multimodal Face Recognition . . . 120
4.8.3 Multi-View Face Recognition on UMIST Dataset . . . 125
4.8.4 Multi-View Object Recognition . . . 126
4.8.5 Multi-View Action Recognition . . . 128
4.9 Conclusion . . . 130

5 Conclusions and Future Work **133**

Bibliography **136**

Appendices **163**

Publications **164**

Vita **167**

List of Tables

2.1 The MCA accuracy on ICPR2012 dataset by using two evaluation strategies "Test set" and "Leave-One-Specimen-Out (LOSO)" for Cell Level classification (Task 1). 40

2.2 The MCA accuracy on ICPR2012 dataset by using two evaluation strategies "Test set" and "Leave-One-Specimen-Out (LOSO)" for Specimen Level classification (Task 2). 41

2.3 The MCA accuracy on ICIP2013 dataset by using two evaluation strategies "HSM" [57] and "Leave-One-Specimen-Out (LOSO)". 42

2.4 The Cell Level confusion matrices by using Leave-One-Specimen-Out method. 43

2.5 The comparison of proposed SCP with SPM strategy by using different pooling functions and using LOSO evaluation method On Cell Level (Task 1). 44

3.1 The average overlap score of 5 trackers on 7 different videos. The best is shown by red and blue is the second best. 73

3.2 Precision (Center Location Error) in OTB-100 (sequence average). The trackers are ordered by the average overlap scores, and the top 5 methods in each attribute are denoted by different colors: **red**, **green**, **blue**, **cyan**, and **magenta**. 80

3.3 Success rate (overlap) in OTB-100 (sequence average). Each entry contains the average overlap in percentage at the overlap threshold of 0.5. The trackers are ordered by the average overlap scores, and the top 5 methods in each attribute are denoted by different colors: **red**, **green**, **blue**, **cyan**, and **magenta**. 81
3.4 Comparing the best trackers of OTB-100, and Deep Learning trackers with MM-THM using Precision rate (Center Location Error) in OTB-100 (sequence average). Each entry contains the average overlap in percentage at the overlap threshold of 0.5. The trackers are ordered by the average overlap scores, and the top 5 methods in each attribute are denoted by different colors: **red**, **green**, **blue**, **cyan**, and **magenta**. 82
3.5 The best trackers of OTB-100, and Deep Learning trackers are compared with MM-THM using Success rate (overlap) in OTB-100 (sequence average). Each entry contains the average overlap in percentage at the overlap threshold of 0.5. The trackers are ordered by the average overlap scores, and the top 5 methods in each attribute are denoted by different colors: **red**, **green**, **blue**, **cyan**, and **magenta**. 83

4.1 The gender classification accuracy (%) with $p = 250$. 120
4.2 Gender classification rates obtained with $p = 25$ atoms. 120
4.3 Face recognition accuracy with the whole face modality 121
4.4 Recognition performance of each single modality in AR database. Modalities include left periocular, right periocular, nose, mouth, and face. 122
4.5 Modalities include 1. left periocular, 2. right periocular, 3. nose, 4. mouth, and 5. face. 123
4.6 Multimodal face recognition results for the AR dataset 123
4.7 Multiview face recognition results for the UMIST datasets 125
4.8 The recognition rate obtained for the "large-baseline" evaluation of BMW. . 127

4.9 Evaluation of MWDL, JTLDL and HTLDL for the recognition rate on "large-baseline" evaluation of BMW 127
4.10 Multiview action recognition on the IXMAS (%) 129

List of Figures

2.1 The illustration for the joint sparse modeling for classification task of two classes with two modalities: (a) The patches from two classes have $M = 2$ modalities that are shown as red and green. There is a color coded dictionary corresponding to each modality. The multimodal sparse representation of each patch is obtained by multimodal dictionaries $\{\boldsymbol{D}_m\}_{m=1}^{M}$ and joint sparsity regularization. The entries of sparse codes have different colors and represent different learned values; the white entries indicate the zero rows and columns. (b) The joint sparsity regularizer that is used to impose high correlation between the sparse representation of a sample in two modalities of {**red**, **green**}. (c) Modality-based sparse codes of all classes $\boldsymbol{X}_m = [\boldsymbol{x}_m^1, \boldsymbol{x}_m^2]$ is used to update $\boldsymbol{D}_m$. (d) The sparse code polling method is used to aggregate local sparse codes and train the SVM classifier in training stage. 30

2.2 SPM method. 35

2.3 Proposed SCP method. 35

2.4 The *Cell Level* images of six classes for the ICPR2012 dataset in (a) and the ICIP2013 dataset in (b): First rows are the *positive* and second rows are the *intermediate* intensity level images. 37

2.5 Representation coefficients generated by proposed regularization for SIFT, SURF and SIFTSURF features. There are six columns corresponding to the six classes. The x-axis is the dictionary columns and the x-axis is the sparse code values corresponding to each dictionary column. nz is the number of non-zero elements in the sparse code vector. 44

2.6 The accuracy of ICPR2012 positive test set versus different dictionary atoms in (a) and λ_1 values in (b). 45

3.1 Illustration of independent vs overlapped coupling. Consider the case with $M = 3$ modalities of **red**, **green** and **blue**. (a) shows a multimodal signal $\boldsymbol{X} = \{\boldsymbol{x}_{red}, \boldsymbol{x}_{green}, \boldsymbol{x}_{blue}\}$ that has a mixture information of mostly red and a smaller amount of green. (b) a multimodal atom $\{\boldsymbol{d}\} = \boldsymbol{d}_{red}, \boldsymbol{d}_{green}, \boldsymbol{d}_{blue}$. The goal is to decompose the multimodal input $\boldsymbol{X}$ using $\{\boldsymbol{d}\}$ to multimodal coefficients $\boldsymbol{A} = [\boldsymbol{\alpha}_{red}, \boldsymbol{\alpha}_{green}, \boldsymbol{\alpha}_{blue}]$. (c) the result of independent coupling using ℓ_{12}. All three values of decompositions are equal. (d) the result of overlapping coupling. The $\boldsymbol{\alpha}_{red} \gg \boldsymbol{\alpha}_{green}$ and $\boldsymbol{\alpha}_{blue} = 0$. 51

3.2 Illustration of a hierarchical structure between various modalities of input data. The tree is $\mathcal{G} = \{g_5, g_4, g_3, \{g_3, g_4\}, \{g_2, g_5\}\}$. It has three leaves of $\{g_3, g_4, g_5\}$, and g_2 enforces coupling between g_3, g_4, and the root of the tree g_1 enforces grouping between g_2 and g_5. Internal nodes near the leaves of the tree correspond to modalities that we expect highly related while the internal nodes near the root represent weakly-correlated sparse codes in its subtree. Any path from leaves to the root, is a possible solution. 53

3.3 Illustration of joint sparsity vs tree-structured grouping. Consider the case with $M = 3$ modalities of **red**, **green** and **blue**. Top: ℓ_{12} all modalities are in one group. Down: The tree-structure regularization enforces hierarchical fusion between various modalities of input data in the space of sparse codes. The tree is $\mathcal{G} = \{g_5, g_4, g_3, \{g_3, g_4\}, \{g_2, g_5\}\}$. It has three leaves $\{g_3, g_4, g_5\}$, and g_2 enforces grouping between g_3, g_4, and the root of the tree g_1 enforces grouping between g_2 and g_5, and is a hierarchical grouping between red and the group of blue and green. The key here is that partially correlated coupling is not allowed in the tree structure. The groups of variables either are independent or one is subset of the other. 54

3.4 We employ the blue rectangular masks and cropping out the corresponding areas. These, along with the whole face, were taken for fusion. Simple intensity values were used as features for all of them. Tree-structure $\mathcal{G}$ corresponding to the four weak modalities of left periocular, right periocular, nose, and mouth, and a strong modality face. The tree represents a set of groups $\mathcal{G}$ between left and right periocular and all modalities at the root. . . 55

3.5 Illustration of intersection closed coupling. The tree-structure $\mathcal{G}$ corresponding to the four weak modalities of left eye, right eye, nose, and mouth, and a strong modality face, $\mathcal{G} = \{g_1, g_2, g_3, g_4, g_5, g_6, g_7\}$. If variable left eye is non-zero, *i.e.* $\boldsymbol{a}^{(g_3)} \neq 0$,then automatically, g_2 is non-zero, and also the root g_1 is selected. This path from $g_3 \rightarrow g_2 \rightarrow g_1$ is a valid solution with three activated groups out of total 7 groups. This solution gets high punishment from reconstruction part of the optimization problem. Intuitively, each interior node that is activated, here g_2, favors to see all of its members, (g_3 and g_4) to be non-zero, to get less reconstruction punishment. 58

3.6 Illustration of the proposed MM-THM framework. Tracking can be seen as a binary classification of target and background. Consider each patch has M different modalities. Originally, the physical attributes are not discriminative enough to distinguish the target from the background. Our method learns a set of dictionaries to find the representation of data in latent space of sparse codes, to make the target more distinctive in each modality, and, from multimodal stand-point. 61
3.7 The hand-coded tree-structure norm-based regularization in space of sparse codes for MM-THM. This tree has $M = 7$ leaves (features) and 5 groups. From left to right: fhog, HSV and CIE Lab channels. 71
3.8 Tracking results of selected 11 trackers in representative frames. Frame indices are shown in the top left of each figure. The showing examples are from sequences carDark, Jogging, Singer1, Bolt, Walking2, Basketball, respectively. 73
3.9 (a) precision and (b) success plots for the 50 videos with all available trackers in the benchmark OTB-50. The proposed MM-THM achieves overall the best performance in both metrics and outperforms the second best tracker SMS and SCM more than 10% and 15%, respectively. 74
3.10 (a) precision and (b) success plots for the 50 videos with all available trackers in the benchmark OTB-50 and recent high-performance trackers in [92]: CN [30], MUSTer [61], KCF [59] and MEEM [204]. The proposed MM-THM outperforms MEEM by 8.2% and has similar performance with MUSTer in success plot and also achieves third best overall performance in precision plot with precision 3.26% less than MUSTer. 75
3.11 (a) precision and (b) success plots for the 100 videos with all available trackers in the benchmark OTB-100. The proposed MM-THM achieves overall the best performance in both metrics and outperforms the second best tracker SMS and STRUCK more than 5% and 13%, respectively. 76

3.12 Comparison with state-of-the-art deep learning trackers. (a) precision and (b) success plots trackers in the benchmark OTB-100. 77
3.13 Comparing MM-THM in precision plot with trackers in OTB-100 in all attributes. The score for each tracker is shown in the legend. The top 10 trackers are presented for the sake of clarity, and the rest are shown as gray dashed curves. 78
3.14 Comparing MM-THM in success plot with OTB-100 trackers in all attributes. The score for each tracker is shown in the legend. The top 10 trackers are presented for the sake of clarity, and the rest are shown as gray dashed curves. 79

4.1 Samples of male and female with extracted modalities in AR dataset. 121
4.2 We employ the blue rectangular masks and cropping out the corresponding areas. These, along with the whole face, were taken for fusion. Simple intensity values were used as features for all of them. Tree-structure $\mathcal{G}$ corresponding to the four weak modalities of left periocular, right periocular, nose, and mouth, and a strong modality face. 122
4.3 Illustration of 3 view-range (modalities) in UMIST. Different poses of a subject from UMIST database. Each row is a view-range or modality for the subject. 126
4.4 (a) Apparatus which instruments five camera sensors [129]. (b) Five "large baseline" images captured at different vantage points. 126
4.5 Check watch action sample from the IXMAS dataset [183]. Each action is viewed by 5 cameras ($M = 5$). 128

Chapter 1

Introduction

1.1 Motivation and Background

A significant amount of studies in signal and image processing has been done to represent signals in a proper fashion for the specific task. Restoration in general and in particular denoising and reconstruction are emerging estimation problems in these fields; that may become difficult to solve without an arbitrary a priori model of the data source. In machine learning and computational statistics, various research tries to answer the question of how to learn a set of parameters from data while a predefined criterion is maximized, in both a supervised or unsupervised scheme. For instance, to find the connection between input data and output response, or when one may need to summarize (compress) the data.

A simple a priori model is to assume the solution to be sparse. This bias towards sparsity can emerge in two scenarios: First, we know that the problem at hand has a sparse solution, or in the absence of sparsity prior information, our interest lies in seeking a simple reasoning for the task that is easy to interpret and has a low processing complexity. This is known as sparsity and can be assumed as selecting a small number of parameters to solve the problems. In early studies, a pre-defined dictionary is used which was made out of a set of orthonormal basis. Then, the signal can be represented using a linear combination of the dictionary elements also known as atoms.

The dictionary should be designed so that it can successfully reconstruct the data while at the same time, has a poor performance in modeling the noise. In that case, the sparse decomposition coefficients and the dictionary together may have a good representation out of the pure signal. That is, to obtain a proper representation of the signals, the design of the dictionary has a significant role and is an active topic of research.

Let us emphasize on the difference between the terminology *models* in this dissertation with *generative models*. We use this terminology to define classes of regularized signals that we design to have interesting characteristics, but it may also have irrelevant representations. However, in generative settings, models are the probability distribution of input data.

The dictionary in statistics and machine learning may be simplified as a set of fixed variables or predictors and then seeking for the solution as a linear combination of variables in the dictionary. However, the method should be designed so that it can successfully generalize the unseen and new data; that is to make sure that the model does not suffer from the overfitting problem. It can occur due to a large number of basis or a small number of training samples. The prior information about the data or the form of the solution leads to the concept of regularization that shows promising results to deal with the overfitting problem. For instance, the Tikhonov regularization that favors towards a smooth solution is a well-known prior among various fields. In this dissertation, the sparse solutions are preferred, which leads to ℓ_1-norm regularization. Particularly, beside the sparsity, we are interested in encoding different prior information over the data or the characteristics of the solution in the pattern of the non-zero coefficients. Sparse models have been successfully applied in the recent two decades in many scientific disciplines: simple model selection out of a pool of possible choices is done using the sparsity principle in statistics and machine learning. Sparsity models try to explain the observed data by selecting a few predictors (atoms of the dictionary). In signal processing, sparsity is used to approximate signals as a linear mixture of a small number of dictionary elements, imposing a union-of-subspaces model on the data. Sparse coding representation has been the topic of a large amount of work in image processing and computer vision.

Classification is a well-known problem in computer vision and machine learning communities. The classification accuracy is mostly investigated from one individual source of information. However, any source of information is limited to its neighborhood, and its efficiency is bounded, and they are prone to be corrupted and become unreliable. So, making decision relying on a single source can jeopardize the decision making process [179, 31]. One solution is to use multiple sources of information when it is possible. The information fusion is split into two broad categories: feature fusion [151] and classifier fusion [153, 164]. In feature fusion, we have features at the input and output of the fusion process. The goal is to make or improve a new feature type from input features. The fusion system has various extracted features from each source at the input level. Classification is done based on the new feature set obtained at the output of the fusion process. That is why feature fusion is called feature in-feature out (FEI-FEO), as well. The simplest way of feature fusion is by concatenation of different features into a vector. In [190] different features from wearable sensors are concatenated to a longer vector to do action classification. In classifier fusion, a classifier that is trained based on each feature type makes its decision. The fusion system combines input decisions to obtain better or new decisions. For example, in [150, 81, 83] classifier fusion is applied in majority voting fashion [219, 82] in biometric recognition using classifiers built based on iris, finger and face data. Beside majority voting, different mixing policies like Bayesian inference are used to do the fusion [93].

The majority of studies in information fusion are based on classifier fusion. However, it cannot fully exploit the cross-correlation between multiple sources of information because each classifier is local and is independent of others. On the other hand, feature fusion showed to get superior performance than classifier fusion in the presence of highly related feature modalities [88]. However, the design of the feature fusion system is more challenging especially when the size of features are not the same. The easiest way to fuse different features is to concatenate them in one large vector. This method has two major drawbacks: 1. The new feature vector is large that may lead to the curse of dimensionality especially

when training data set is small. 2. it neglects the useful cross-correlation information and even may contain noise and outlier which is reported problematic in noisy environments [151].

Our focus in this dissertation is to creating a joint multimodal representation by embedding the representation of every single-modal into a common (latent) representation space. There are two main groups of such approaches:

1. The initially disjoint modalities are exploited to create a joint representation. The goal of this step is to make a proper representation in the latent space. To be concise, this step does not necessarily provide a bidirectional mapping. In other words, we do not necessarily seek to regenerate original physical space from joint multimodal latent space. These approaches are typically used in retrieval and classification tasks.

2. Bi-directional mapping is mainly discussed in cross-modal approaches, which may or may not include learning a joint representation space. The main focus is to generate one modality from the latent representation of another modality and back, as well as represent them in a joint representation space. These methods are popular when there is a need for cross-modal translation. For instance, cross-modal retrieval.

1.2 Sparse Representation

Sparse representation is a well-accepted method to describe signals mainly because natural signals are in fact sparse when the description is done in space of specific basis. These set of bases that describe the space for signal representation is called dictionary in the signal processing community. Each column of the dictionary is called an atom, and usually, the number of atoms are more than the dimension of the signal especially for reconstruction tasks. Modeling data in sparse representation scheme is based on an ability to represent input data as linear combinations of a few dictionary elements. Therefore, the model is shown to be promising when the dictionary is chosen so that it can generate proper sparse decomposition coefficients. The proper model of a dictionary is selected in two ways: i) a mathematical model of the data is the lead to obtain a dictionary, or ii) learning a dictionary to perform

best on a training set. In the early research, dictionaries are obtained using the Fourier and wavelet basis [184, 166, 168]. The method performed well for 1-dimensional signals, especially, for the signal approximation, denoising, and reconstruction from incomplete data. The curvelets used to build dictionary elements in [22] which was extended in [34] to introduce a new sampling method called the Compressive Sensing (CS). However, these dictionaries perform poorly in more complex scenarios like high-dimensional signals.

Although Compressive Sensing (CS) was first introduced for the signal approximation and compression with potentially lower sampling rates than the Shannon bound, recent research has shown the superior performance of the sparse coding scheme for discriminative tasks as well [184, 185]. In the early works, the dictionary was made from all training samples, and the test data is assumed to be reconstructed from training samples inside the dictionary that have the same label as the query. In other words, the test sample is approximated with a few training samples belonging to the same class as test data and not the other classes. In this scenario, the dictionary is made by horizontally concatenating training samples of all classes, without any update or learning involved.

Recently, there has been much interest in applying sparse representation methods to model fusion at the feature level also known as "multi-task learning". The idea is to reconstruct a multimodal sample from several tasks (sources, views, *etc.*) by adopting various sparsity models [156, 203, 133]. In [203], a joint sparse model is applied to represent the observations from the same class simultaneously using a few train samples. That is, different observation of test data would result in the same sparsity pattern that lies in a low dimensional subspace. In [156] joint sparsity model is used to modelling the heterogeneous sources and showed to be promising for biometric recognition. A kernelized version is proposed in [201] to handle non-linearity in feature domain and applied to visual recognition problem.

1.2.1 Variable Selection by Sparsity Regularization

In a broad sense, an important part of this dissertation is about variable selection or feature selection. Variables/features are descriptors used to represent the data, such as the intensity of a pixel in an image or the frequency of a word in a document. Nowadays, data are becoming abundant in various scientific and industrial domains, and also, they are available in elegant and more involved representations (*e.g.*, high resolution images).

In this context, variable selection is crucial for three tasks [56, 168, 167]: (1) "summarizing" the representation of the data to become more interpretable and understandable, (2) achieving a more small but effective representation, for instance, for compression, (3) examining the predictive ability of the different features, especially for the tasks like classification and recognition that prediction accuracy matters.

In this dissertation, we are interested in these three aspects and mostly focus on the first and last purposes. By variable selection, our goal is to find a small subset of related covariates between a total of p variables which is learning a sparse vector of parameters $\boldsymbol{\alpha}$ in $\mathbf{R}^p$ whose set of nonzero coefficients models the corresponding set of selected features. We will express the precise definitions and formulations of the underlying learning problems in the upcoming sections. Let us introduce more formally the concept of sparsity-inducing regularization.

1.2.2 Sparse Based Regularization

It is a common approach in statistics, machine learning, and signal processing that in order to learn a vector of parameters $\boldsymbol{\alpha}$ in $\mathbf{R}^p$, a convex function $f : \mathbf{R}^p \to \mathbf{R}_+$ is subject to minimization that measures how well $\boldsymbol{\alpha}$ fits some data. We consider the function f to be differentiable with Lipschitz continuous gradient in all of the scenarios in this dissertation. The criterion to choose the function f strongly depends on the application. In general, it corresponds to either a data-fitting term or the average of a loss function over a training set of data, also known as empirical risk [155].

The function f does not model the prior information that we have about the task in hand. In sparse coding, the a priori assumption to perform variable selection is that the learned vector $\boldsymbol{\alpha}$ should be sparse. A regularization term $\Omega : \mathbf{R}^p \rightarrow \mathbf{R}_+$, is considered to enforce the prior knowledge. Hence, our formulation becomes

$$\underset{\boldsymbol{\alpha} \in \mathcal{A}}{\operatorname{argmin}} f(\boldsymbol{\alpha}) + \lambda \Omega(\boldsymbol{\alpha}) \tag{1.1}$$

The scalar $\lambda \geq 0$ is known as the regularization parameter, and it controls the trade-off between the data fidelity term f and the model term Ω. The convex set $\mathcal{A} \subseteq \mathbf{R}^p$ identifies the attributes that we are interested in the design of the problem, such as the non-negativity of the coefficients of $\boldsymbol{\alpha}$. To promote sparse solutions, Ω should intuitively punish vectors $\boldsymbol{\alpha}$ that has many nonzero elements. Thus, the ℓ_0 pseudo-norm is considered,

$$\|\boldsymbol{\alpha}\|_0^0 \triangleq |\{j \in \{1, \ldots, p\} \text{ s.t. } \boldsymbol{\alpha}_j \neq 0\}|.$$

ℓ_0 in Eq. (1.1), promotes the vector to be more sparse. However, this regularizer is not continuous, and soon will turn to combinatorial problems and is NP-hard in general [131]. To deal with ℓ_0-norm computational challenges a surrogates (or relaxations) is considered via an efficient ℓ_1 optimization problem [12]. The relaxation preserves the desired sparsity properties, and also makes the optimization computationally-tractable and has been successfully applied for face recognition [185, 54], ear recognition [84, 85], person re-identification [182] and tracking [171, 117, 170, 169].

Lasso and Basis Pursuit. To elaborate the key properties common to more general sparsity-inducing norms, we first focus on the ℓ_1-norm as the most popular sparsity norm. The ℓ_1-norm regularization was subject to many studies and research for the last decade to expand its theoretical frameworks [176, 26] and to provide efficient tools with various applications, such as compressed sensing [21], and image reconstruction [103]. In statistics ℓ_1-norm regularization is studied within the context of least-squares regression and is known as Lasso [176] while it is known as basis pursuit in signal processing [26]. We have

written both formulas to highlight the fact that although both of them are similar from optimization viewpoint, the ℓ_1-norm regularization is observed differently in statistics and signal processing. In statistics formulation Eq. (1.1) is known as Lasso and is written as

$$\underset{\boldsymbol{w}\in\mathbf{R}^p}{\operatorname{argmin}} \ \frac{1}{2}\|\boldsymbol{y}-\boldsymbol{X}\boldsymbol{w}\|_2^2+\|\boldsymbol{w}\|_1 \tag{1.2}$$

while in signal processing it is known as basis pursuit

$$\underset{\boldsymbol{\alpha}\in\mathbf{R}^p}{\operatorname{argmin}} \ \frac{1}{2}\|\boldsymbol{x}-\boldsymbol{D}\boldsymbol{\alpha}\|_2^2+\|\boldsymbol{\alpha}\|_1 \tag{1.3}$$

We use $\boldsymbol{X}\in\mathbf{R}^{C\times p}$ to determine a set of C observations described by p variables, while we try to predict $\boldsymbol{y}$ in $\mathbf{R}^C$ as the corresponding target value of observations. For classification, the elements of $\boldsymbol{y}$ are the label of the C observations. However, in basis pursuit, m-dimensional signal $\boldsymbol{x}$ in $\mathbf{R}^m$ is represented as a linear combination of p columns $\boldsymbol{d}^1,\ldots,\boldsymbol{d}^p$ of the dictionary $\boldsymbol{D}\in\mathbf{R}^{m\times p}$. The dictionary $\boldsymbol{D}$ is either fixed or made from learned representations as in [135].

It is worth mentioning that the primary goal of ℓ_1 regularizer is to penalize vectors of parameters with a large number of nonzero elements and treat each variable separately. We are interested to model the a priori known structural information about the variables using sparsity-inducing norms. The structural information is assumed to be available and known a priori.

1.3 Dictionary Learning

The fixed dictionaries are usually made by linear combination of a few elements from wavelets, discrete cosine transform [112, 142, 3, 10, 8]. Restoration and reconstruction of natural images are modeled successfully by predefined fixed dictionaries. The fixed dictionaries do not have any learning step involve and they simply are constructed by putting all training samples together and make one large dictionary [185, 143, 144]. This large dictionary is fixed and despite other classification methods, is not going to be updated.

Dictionary learning methods can be divided to two groups of unsupervised and supervised methods. The optimization formula in unsupervised dictionary learning only has reconstruction regularization and mostly used for denoising and reconstruction applications in signal and image processing [123, 2, 7]. Supervised dictionary learning exploits labels of training data and beside reconstructive regularization has discriminative prior as well which leads to better result in discriminative tasks [104, 125, 126, 6]. Despite principal component analysis (PCA) that basis are required to be orthogonal, the atoms of the dictionary do not have to be independent. This advantage gives more flexibility in design of dictionary learning methods and consequently makes it easy for the algorithm to be tuned for different input data.

Assume N signals with m dimension as $\boldsymbol{X} = [\boldsymbol{x}^1, \dots, \boldsymbol{x}^N] \in \mathbf{R}^{m \times N}$. For example it may represent N patches with size m pixels. Also, consider the dictionary with p elements or atoms as $\boldsymbol{D} = [\boldsymbol{d}^1, \dots, \boldsymbol{d}^p] \in \mathbf{R}^{m \times p}$. The dictionary learning methods try to represent each signal $\boldsymbol{x}$ as a linear combination of atoms $\{\boldsymbol{d}^i\}_{i=1}^p$. The matrix $\boldsymbol{A} = [\boldsymbol{\alpha}^1, \dots, \boldsymbol{\alpha}^N] \in \mathbf{R}^{p \times N}$ includes decompositions, also known as codes for the N signals. The goal is to jointly learn dictionary and decompositions $(\boldsymbol{D}, \boldsymbol{A})$ so that we can express the input signals as $\boldsymbol{X} \approx \boldsymbol{D}\boldsymbol{A}$. We measure the quality of the data-fitting with mostly square loss function since $\boldsymbol{X}$, $\boldsymbol{D}$ and $\boldsymbol{A}$ are in matrix form. The number of possible candidate pairs in the space $(\boldsymbol{D}, \boldsymbol{A})$ can be reduced by some priors on $\boldsymbol{D}$ and/or $\boldsymbol{A}$. The constraints are useful to model the knowledge that we have about the task. As an example, consider non-negative matrix factorization which basically enforces both $\boldsymbol{A}$ and $\boldsymbol{D}$ to be non-negative:

$$\underset{\boldsymbol{A} \in \mathbf{R}_+^{p \times n}, \boldsymbol{D} \in \mathbf{R}_+^{m \times p}}{\operatorname{argmin}} \|\boldsymbol{X} - \boldsymbol{D}\boldsymbol{A}\|_F^2 .$$

The first application of non-negative matrix factorization was for face recognition, where the signals are expected to be non negative [48, 121]. Assume $\mathcal{A} \subseteq \mathbf{R}^{p \times N}$ and $\mathcal{D} \subseteq \mathbf{R}^{m \times p}$ as convex set of all possible candidates for $\boldsymbol{\alpha}$ and $\boldsymbol{D}$, respectively and Ω as sparsity regularization on $\boldsymbol{A}$. Then, the dictionary learning with sparsity-inducing regularization

is

$$\underset{\boldsymbol{A}\in\mathcal{A},\boldsymbol{D}\in\mathcal{D}}{\operatorname{argmin}} \ \frac{1}{2}\|\boldsymbol{X}-\boldsymbol{D}\boldsymbol{A}\|_F^2+\lambda\Omega(\boldsymbol{A}) \tag{1.4}$$

where λ is the regularization parameter for Ω. Usually, Ω decomposes to sum of independent regularizations of the columns/rows of the $\boldsymbol{A}$. In Eq. (1.4), Ω penalizes $\boldsymbol{A}$, and there is no regularization over $\boldsymbol{D}$ which may cause the coefficients of the matrix $\boldsymbol{A}$ to be small. That is, we enforce the set $\mathcal{D}$ to be the set of matrices whose columns are bounded by unit ℓ_2-norm ball.

The optimization problem (1.4) has the product of the two variables as $\boldsymbol{D}\boldsymbol{A}$, so, the problem is not joint convex in the space of $(\boldsymbol{A},\boldsymbol{D})$. But, when one of the two optimization variable is fixed, the problem (1.4) is convex with respect to the other variable [108, 9, 122].

Sparse coding is one example of (1.4), where the goal is to learn a dictionary which represent all the signals properly so that the obtained decompositions would be sparse

$$\underset{\boldsymbol{A}\in\mathbf{R}^{p\times N},\boldsymbol{D}\in\mathcal{D}}{\operatorname{argmin}} \ \frac{1}{2}\|\boldsymbol{X}-\boldsymbol{D}\boldsymbol{A}\|_F^2+\lambda\sum_{i=1}^{N}\|\boldsymbol{\alpha}^i\|_{\ell_1} \tag{1.5}$$

where the constraint over $\mathcal{D}$ usually is chosen as projection to unit norm ball so that the each atom of the dictionary has ℓ_2-norm of smaller than or equal to one. Dictionary learning using structured sparsity successfully applied to localized features for face recognition [74, 54] and the denoising of natural image patches [73, 45, 124]. We can encode prior information in different ways within the sparse coding paradigm of (1.4), because we have access to the factorization $\boldsymbol{D}\boldsymbol{A}$: 1. applying sparsity regularization on dictionary elements which change the m-dimensional features, 2. applying regularization on columns of $\boldsymbol{A}$ or the rows of $\boldsymbol{D}$ affect the latent variables, and 3. regularizing rows of $\boldsymbol{A}$ to impose grouping between different signals.

1.4 Contributions and Outline

In this dissertation, we study the problem of multimodal signal processing in a particular representation called sparse coding, which has proven to be effective for many applications. Our goal is to produce new algorithmic mechanisms and applications to this scheme, and in particular, exploit structured sparsity in order to apply feature fusion to obtain better classification accuracy when possible. Specifically, within each modality, we need the dictionary to be reconstructive, so that it can successfully reconstruct the data while at the same time, has a poor performance in modeling the noise. Also, the dictionary of each modality should be discriminative, so that it can decompose the input data to sparse coefficients that are distinctive enough between the classes, that even a simple linear classifier that is trained over the sparse codes can generate high classification accuracy.

On the other hand, the relation between different modalities in physical space is translated as grouping between their corresponding decomposition coefficient vectors in the space of sparse codes: the sparsity pattern of the multimodal sparse coefficient vectors is enforced to convey the desired prior information (here coupling structure between modalities). Our intuition is that this may provide codes that are more distinctive between different classes and so; better classification accuracy in the end.

- In Chapter 2, we begin with introducing a family of structured sparsity-inducing norms and investigate their characteristics. In particular, the connection between different regularization and their grouping effect are elaborated. Then, we study the unsupervised dictionary learning as a convex non-smooth matrix factorization optimization problem, while feature fusion is embodied in the space of sparse codes, and propose a new solution to the corresponding challenging optimization problems. The dictionary learning method obtains a dictionary for each modality in an online scheme based on stochastic approximation.

  We elaborate our proposed multimodal learning approach that fully exploits the information of all modalities, and also embed the correlation between modalities. Our proposed model is carefully designed not to neglect the modal-specific information.

This is an important aspect of the fusion design because the fusion technique should not contaminate the modality-specific part by the modality-shared information which degrades the discriminative power of the learned features. We evaluate the proposed methods on various real-world discrimination tasks from several fields, to clarify when and why feature fusion in space of sparse codes is useful. Specifically, we investigate our proposed method for HEp2 cell classification from biomedical community in Chapter 2.

- In Chapter 3 we extend our method to include fusion between features when multimodal dictionaries are embodied in a hierarchical tree structure. The superior performance of our framework is reported for visual tracking task in Computer Vision community. The visual tracking in the sparsity scheme was studied and a method was proposed to learn the unsupervised dictionary and classifier while obtaining multimodal sparse representation of each positive and negative patches using tree-structure sparsity model. The imposed tree-structured joint sparsity enabled the algorithm to fuse information at feature-level in different granularity by forcing their sparse codes to have similar basis within each group and at decision-level by augmenting the classifier decisions.

- We turn into supervised learning methods in Chapter 4 and try to obtain the dictionary that is learned to adapt to the specific task and not only to the data. We intend to design methods that are able to obtain *reconstructive* and *discriminative* dictionary. Similar to unsupervised methods, dictionary should be reconstructive, *i.e.*, it should represent data well and perform poor to reconstruct the noise. Also, it should be discriminative, *i.e.*, the dictionary is able to encode intrinsic properties of the multimodal data in a decomposition coefficient vector that is favorable towards the maximal discriminatory power. To meet this goal, we extend the optimization problems of Chapter 2 to include a set of multimodal classifiers. In Chapter 4, we investigate an efficient optimization when the relation between multimodal dictionaries and classifiers are explicitly defined and provide an exact solution to the problem.

Furthermore, we evaluate the proposed method on multimodal face recognition, multiview object recognition, and multiview action recognition. We extend our approach in Chapter 4, where we intend to study the supervised dictionary learning methods when the multimodal dictionaries are defined implicitly in the sparse coding step. We introduce required propositions to show the differentiability and gradients of loss function and provide the exact proof for them.

Chapter 2

Sparse Representation Classification

2.1 Introduction

In this chapter, we briefly cover sparse representation classification for single modality and its extension for multimodal data. Understanding SRC is vital for the discussions in Chapters 2, 3 and 4, and is explained in Section 2.1.1. Then, we will extend it to include unsupervised dictionary learning in the Section 2.2. The solution to the the proposed non-convex optimization is illustrated in Section 2.2.3 and 2.2.4. The superior performance of the proposed method is evaluated for the task of HEp2 cell classification in Section 2.3.

Notation. We indicate vectors by bold lower case letters, and matrices by bold upper case ones. For a vector $\boldsymbol{x}$ in $\mathbf{R}^m$ and integer j in $[\![1;m]\!] \triangleq \{1,\ldots,m\}$, the j-th entry of $\boldsymbol{x}$ is denoted by $\boldsymbol{x}_j$. For a matrix $\boldsymbol{X}$ in $\mathbf{R}^{m\times n}$, and a pair of integers $(i,j) \in [\![1;m]\!] \times [\![1;n]\!]$, the entry at row i and column j of $\boldsymbol{X}$ is denoted by $\boldsymbol{X}_{ij}$ and we show the vector of i-th row in $\mathbf{R}^n$ as $\boldsymbol{X}_{i\rightarrow}$, the vector of j-th column in $\mathbf{R}^m$ as $\boldsymbol{X}^{j\downarrow}$. When Λ is a finite set of indices, the vector $\boldsymbol{x}_\Lambda$ of size $|\Lambda|$ contains the entries of $\boldsymbol{x}$ corresponding to the indices in Λ. Similarly, when $\boldsymbol{X}$ is a matrix of size $m\times n$ and $\Lambda \subseteq [\![1;n]\!]$, $\boldsymbol{X}_\Lambda$ is the matrix of size $m\times|\Lambda|$ containing the columns of $\boldsymbol{X}$ corresponding to the indices in Λ.

Let us define $\text{supp}(\boldsymbol{X}_{i\rightarrow}) \subset [1, M]$ as the support of the i-th row vector $\boldsymbol{X}_{i\rightarrow}$, *i.e.*, the set of variables $m \in [1, M]$ such that $\boldsymbol{X}_{ij} \neq 0$. A group of variables is a subset $g \subset [1, M]$. We define the M dimensional vector $\boldsymbol{X}^{(g)}_{r\rightarrow} = [\boldsymbol{X}^{(g)}_{r1}, \ldots, \boldsymbol{X}^{(g)}_{rM}]^\top$ contains the entries of $\boldsymbol{X}_{r\rightarrow}$ corresponding to the indices in g and zero otherwise.

The ℓ_q-norm of a vector $\boldsymbol{x} \in \mathbf{R}^m$ for $q \geq 1$ would be: $\|\boldsymbol{x}\|_q \triangleq \left(\sum_{j=1}^m |\boldsymbol{x}_j|^q \right)^{\frac{1}{q}}$. We denote the Frobenius norm of a matrix $\boldsymbol{X} \in \mathbf{R}^{m \times n}$ by:

$$\|\boldsymbol{X}\|_F \triangleq \left(\sum_{i=1}^{m} \sum_{j=1}^{n} \boldsymbol{X}_{ij}^2 \right)^{1/2}$$

For any matrix $\boldsymbol{A} = [\boldsymbol{\alpha}^1, \boldsymbol{\alpha}^2, \ldots, \boldsymbol{\alpha}^p]$ in $\mathbf{R}^{n \times p}$, for the j-th column of $\boldsymbol{A}$ with size p, we write $\boldsymbol{\alpha}^j$ or $\boldsymbol{A}^{j\downarrow}$. We refer to the set $\{j \in [\![1; p]\!]; \boldsymbol{\alpha}_j \neq 0\}$ as the support, or nonzero pattern of the vector $\boldsymbol{\alpha} \in \mathbf{R}^p$. Let C represent the number of classes in the data set, N_c as the number of training data from the c-th class and $N = \sum_{c=1}^{C} N_c$ as a total number of statistically independent and normalized training data. The $\{i\}_{[\![1;N]\!]}$-th sample that has label c, $\boldsymbol{X}^i_c$ with label $y^i = c$, is multimodal and is observed from M different feature modalities $\boldsymbol{X}^i_c = \{\boldsymbol{x}^i_{c,m} \in \mathbf{R}^{n_m}\}_{m \in [\![1;M]\!]}$ where n_m is the dimension of the m-th feature modality and $\boldsymbol{x}^i_{c,m}$ is the m-th modality of the i-th sample that belongs to the class c. Let us denote the set of training samples of the c-th class in m-th modality as $\boldsymbol{X}_{c,m} = \{\boldsymbol{x}^i_{c,m} | i \in \{1, \ldots, N\}, y^i = c\}$ shortly as $\{\boldsymbol{x}^i_{c,m}\}$, and also the set of all N training samples from m-th modality as $\boldsymbol{X}_m = [\boldsymbol{x}^1_m, \ldots, \boldsymbol{x}^N_m]$.

2.1.1 Single Modal Case

The sparse representation classification was introduced for application of face recognition in [185]. The class specific dictionary $\boldsymbol{D}_c$ is fixed and is made by concatenating all training samples that belong to the c-th class as $\boldsymbol{D}_c = \boldsymbol{X}_c \in \mathcal{R}^{n \times N_c}$. The final dictionary is made by putting together all class specific dictionaries as $\boldsymbol{D} = [\boldsymbol{D}_1, \ldots, \boldsymbol{D}_C] \in \mathbf{R}^{n \times N}$ in the one-against-all scheme. Hence, we know the label of each atom. The task is to identify the label of a test sample $\boldsymbol{x}_t \in \mathbf{R}^n$. Sparse representation classification (SRC) assumes that the test

signal from c-th class can be represented using atoms that belong to the c-th class (the $\boldsymbol{D}_c$ part of the dictionary $\boldsymbol{D}$). In other words, the test sample $\boldsymbol{x}_t$ from c-th class, is assumed to lie in the space span by the $\boldsymbol{D}_c$ and can be approximated using few number of training samples in $\boldsymbol{D}_c$:

$$\boldsymbol{x}_t = \boldsymbol{D}\boldsymbol{\alpha}_t + \boldsymbol{e} \tag{2.1}$$

where $\boldsymbol{\alpha}_t$ is the decomposition coefficient vector that SRC expects it to be zero everywhere except for atom indices that belong to the c-th class, *i.e.* $\boldsymbol{\alpha}_t = [\boldsymbol{0}^\top, \ldots, \boldsymbol{\alpha}_c^\top, \ldots, \boldsymbol{0}^\top]^\top$ and $\boldsymbol{e}$ is the noise. That is, to reconstruct the query using the minimum number of atoms, which is equal to search for sparse decomposition vector $\boldsymbol{\alpha}_t$ to reconstruct the test signal $\boldsymbol{x}_t$ by minimizing the ℓ_0 norm as follows:

$$\underset{\boldsymbol{\alpha}}{\operatorname{argmin}} \|\boldsymbol{\alpha}\|_{\ell_0} \quad \text{s.t.} \|\boldsymbol{x}_t - \boldsymbol{D}\boldsymbol{\alpha}_t\|_{\ell_2}^2 \leq \epsilon \tag{2.2}$$

where $\|.\|_{\ell_0}$ is the zero norm defined as the number of nonzero entries in $\boldsymbol{\alpha}_t$ and ϵ is the upper bound of noise energy. The ℓ_0 regularization is a discontinuous function and highly sensitive to noise, plus its minimization requires combinatorial search. That is why the proposed methods to solve this NP-hard optimization problem, *e.g.* Iterative Hard Thresholding [17] and Orthogonal Matching Pursuit [178] only find the sub-optimal solution. SRC problem in Eq. (2.2) is reformulated with ℓ_1-norm as:

$$\underset{\boldsymbol{\alpha}_t}{\operatorname{argmin}} \|\boldsymbol{\alpha}_t\|_{\ell_1} \quad \text{s.t.} \|\boldsymbol{x}_t - \boldsymbol{D}\boldsymbol{\alpha}\|_{\ell_2}^2 \leq \epsilon \tag{2.3}$$

where $\|\boldsymbol{\alpha}_t\|_{\ell_1}$ is defined as summation of absolute value of entries of the decomposition vector. Although in general there is no analytical way to show the link between the sparsity and ℓ_1-norm, it is intuitively clear why ℓ_1-norm leads to sparse solution. In the current application in hand, in the presence of sufficient training samples for each class, the solution of ℓ_1-norm leads to sparse solution. The reason lies in the fact that with a large number of atoms in $\boldsymbol{D}$, we can expect $\boldsymbol{\alpha}_t$ to be highly sparse. As discussed in [34, 185] when $\boldsymbol{\alpha}_t$ is highly sparse the convex ℓ_1 can be used instead of ℓ_0-minimization.

Assuming the query to belong to the c-th class, the vector $\delta_c(\boldsymbol{\alpha}_t)$ in $\mathbf{R}^N$ is zero every where except entries that are associated with the c-th class: $\delta_c(\boldsymbol{\alpha}_t) = [\mathbf{0}^\top, \ldots, \mathbf{0}^\top, \boldsymbol{\alpha}_c^\top, \mathbf{0}^\top, \ldots, \mathbf{0}^\top]^\top$. The test data is approximated as: $\hat{\boldsymbol{x}}_t = \boldsymbol{D}\delta_c(\boldsymbol{\alpha}_t)$. The test data will be assigned to the class label, c^* that can reconstruct the query with least reconstruction error:

$$c^* = \underset{c}{\operatorname{argmin}} \ \|\boldsymbol{x}_t - \boldsymbol{D}\delta_c(\boldsymbol{\alpha}_t)\|_{\ell_2} \tag{2.4}$$

In SRC, the dictionary is made by concatenation of all training samples of all classes which means it does not need to be carefully designed features. But, the accuracy of classification depends strongly on a sufficient number of training samples from each class so that the distribution of each class can be approximately obtained.

2.1.2 Multimodal Case

In Section 2.1.1 we briefly cover classification using SRC while only single source of information is provided. Now, we will extend SRC to be able to do fusion at feature-level similar to [133]. The idea is to exploit correlation between different sources of information in the space of sparse codes. The fusion between different modalities of each sample is modeled using joint sparsity regularization on the corresponding sparse representations.

The class-specific dictionary from the m-th modality is made by concatenating all N_c samples as $\boldsymbol{D}_{c,m} = \boldsymbol{X}_{c,m}$ in $\mathbf{R}^{n_m \times N_c}$. The m-th modality dictionary, $\boldsymbol{D}_m$, is made by putting together dictionaries of all classes in that modality: $\boldsymbol{D}_m = [\boldsymbol{D}_{1,m}, \boldsymbol{D}_{2,m}, \ldots, \boldsymbol{D}_{C,m}] \in \mathbf{R}^{n_m \times N}$ and $m \in [\![1; M]\!]$. Therefore, the dictionary of each modality is fixed and is made from all training samples from that modality. Given a set of multimodal dictionaries $\{\boldsymbol{D}_m\}$ and $m \in [\![1; M]\!]$ the goal is to classify the multimodal test signal $\boldsymbol{X}_t$ which is observed from M modalities, $\boldsymbol{X}_t = \{\boldsymbol{x}_{t,m}\}_{m \in [\![1;M]\!]}$.

According to sparse representation, each modality of a signal can be approximated well using a linear combination of a few most relevant dictionary elements. Hence, for the test signal $\boldsymbol{X}_t$ with label c all M modalities should vote for the c-th class. The signal in each

modality is reconstructed using the corresponding dictionary: $\boldsymbol{x}_{t,m} \approx \boldsymbol{D}_m \boldsymbol{\alpha}_{t,m}$, where $\boldsymbol{\alpha}_{t,m} \in \mathbf{R}^N$ is sparse representation of test signal in m-th modality. The non-zero entries of $\boldsymbol{\alpha}_{t,m}$ should relate to those atoms inside $\boldsymbol{D}_m$ that belong to c-th class. Consider $\boldsymbol{A}_t$ which is made by putting together sparse codes of M different modalities as $\boldsymbol{A}_t = [\boldsymbol{\alpha}_{t,1}, \ldots, \boldsymbol{\alpha}_{t,M}] \in \mathbf{R}^{N \times M}$. Hence, it is reasonable to expect that the columns of $\boldsymbol{A}_t$, or different modalities of the multimodal signal, in space of sparse codes, vote for the same class label. This expectation originated from the fact that, the $\boldsymbol{\alpha}_{t,1}, \ldots, \boldsymbol{\alpha}_{t,M}$ are the representation of same observation from different sources of information. The joint sparsity regularization enforces $\boldsymbol{A}_t$ to be row sparse (only small number of rows in $\boldsymbol{A}_t$ are non-zero). In other words, joint sparsity assumes that the test signal should be reconstructed using the same set of index of training samples in dictionary of each modality. The non-zero rows are related to training samples of specific class.

The $\{j\}_{j \in [\![1;N]\!]}$-th atom, $\{\boldsymbol{d}_1^j, \boldsymbol{d}_2^j, \ldots, \boldsymbol{d}_M^j\}$ is a multimodal feature with a structural relation. If we assume atom indices that belong to c-th class as a group, g_c, then we have C groups: $\mathcal{G} = \{g_1, g_2, \cdots, g_C\}$. In other words, g_c has indices of those N_c atoms of $\boldsymbol{D}_{c,m}$ inside $\boldsymbol{D}_m$ and $\mathcal{G}$ segments N rows of $\boldsymbol{A}_t$ to C groups. Joint sparse modeling tries to select or remove simultaneously all the variables forming a group which leads to the $\boldsymbol{A}_t$ that has a few non-zero rows. That is, common column support from each modality-based dictionary $\boldsymbol{D}_m$ and m in $\{1, \ldots, M\}$ are chosen to reconstruct the multimodal input data. The joint sparsity constraint is applied using ℓ_1/ℓ_q with $q > 1$:

$$\underset{\boldsymbol{A}_t \in \mathbf{R}^{N \times M}}{\operatorname{argmin}} \quad f(\boldsymbol{A}_t) + \lambda \Omega(\boldsymbol{A}_t) \tag{2.5}$$

where λ is the regularization parameter, loss function $f : \mathbf{R}^{N \times M} \to \mathbf{R}$ is a convex and smooth defined as: $f(\boldsymbol{A}_t) = \sum_{m=1}^{M} \frac{1}{2} \|\boldsymbol{x}_{t,m} - \boldsymbol{D}_m \boldsymbol{\alpha}_{t,m}\|_{\ell_2}^2$, and $\Omega : \mathbf{R}^{N \times M} \to \mathbf{R}$ is known as mixed ℓ_1/ℓ_q regularization function defined as [12]:

$$\Omega(\boldsymbol{A}) = \|\boldsymbol{A}\|_{\ell_1/\ell_q} = \sum_{i=1}^{N} \sum_{g \in \mathcal{G}} \left\{ \left| \boldsymbol{A}_{i|g} \right|^q \right\}^{1/q} \tag{2.6}$$

where $\boldsymbol{A}_{i|g}$ is the i-th row of $\boldsymbol{A}$ with size M whose coordinates are equal to those of $\boldsymbol{A}_{i\rightarrow}$ for indices in the set g, and 0 otherwise. In fact, ℓ_1/ℓ_q imposes ℓ_1 norm on groups and that is why $\Omega(\boldsymbol{A}_t)$ supports group sparsity. It is important to note that applied ℓ_2 norm inside each group g does not promote sparsity. The group Lasso formulation is obtained by combination of ℓ_1/ℓ_q and square loss function f [200, 165]. In other words, joint sparsity (mixed ℓ_1/ℓ_q) is a set-partitioning problem that represent independent grouping between modalities of each signal in the space of sparse codes.

Assume $\delta_c \in \mathbf{R}^N$ as an operator which is applied on m-th column of $\boldsymbol{A}$ and it only keeps coefficients that are corresponding to atoms of the c-th class in that modality and make the rest coefficients zero. To find the label of test signal, the sparse representation of it from each modality is obtained by:

$$\underset{\boldsymbol{A}_t=[\boldsymbol{\alpha}_{t,1},\ldots,\boldsymbol{\alpha}_{t,M}]}{\operatorname{argmin}} \sum_{m=1}^{M} \frac{1}{2}\|\boldsymbol{x}_{t,m} - \boldsymbol{D}_m\boldsymbol{\alpha}_{t,m}\|_{\ell_2}^2 + \lambda\Omega(\boldsymbol{A}_t) \tag{2.7}$$

Therefore, the test signal in each modality m in $\{1,\ldots,M\}$ is reconstructed from each class as $\hat{\boldsymbol{x}}_{m,c} \approx \boldsymbol{D}_m\delta_c(\boldsymbol{\alpha}_{t,m})$. The query is assigned to the class that can reconstruct it with the least error:

$$c^* = \underset{c}{\operatorname{argmin}} \sum_{m=1}^{M} \|\boldsymbol{x}_{t,m} - \hat{\boldsymbol{x}}_{m,c}\|_{\ell_2}^2 \tag{2.8}$$

2.2 Unsupervised Dictionary Learning

So far we assume that there is a dictionary for each class which is made by concatenating all training samples of that class. The dictionary is used to reconstruct the test data. The test signal belongs to the class with the minimum reconstruction error. The dictionaries are fixed without involving any training step and is made simply by putting together all training samples. This fixed and pre-defined dictionary has two issues: 1. To get a high accuracy, the dictionary of each class should have a sufficient number of training samples from that class. That is the atoms of the dictionary should see enough samples of each class. Increasing the number of training samples ends up with the large dictionary that has

many atoms. Therefore, we can expect to have higher computational complexity for the optimization process to estimate sparse codes. 2. It has been shown that the dictionary obtained by simply putting together the training samples does not lead to the optimal solution in reconstructive nor discriminative tasks [105, 111].

Learning dictionary from the data has proven to be effective to solve the above issues. Dictionary learning from the optimization perspective is a non-convex matrix factorization problem. In the community of machine learning and signal processing dictionary learning and non-negative matrix factorization are formulated as different matrix factorization problems but with the same goal: to get a few basis elements from data. In this dissertation, the dictionary is learned as the optimization of a smooth non-convex objective function over a convex set.

In the line of image and video processing research, dictionary learning shows promising results in the reconstructive tasks like image restoration [109] and discriminative tasks like face recognition [73], and object recognition [77] comparing to the fixed dictionaries [106]. Dictionary learning methods can be categorized to two parts: unsupervised and supervised algorithms. In unsupervised dictionary learning the optimization formula only has reconstruction penalty, and therefore, the dictionary is adapted to the data. The unsupervised dictionary learning methods are applied for mostly reconstructive tasks like image inpainting [105] and signal and image denoising [36]. Although in unsupervised approach there is no discriminative penalty, the obtained dictionary is applied for discriminative tasks like classification [14].

2.2.1 Single Modal Case

Various studies in machine learning, statistics and signal processing, have been proposed to find the atoms as the interpretable basis elements from a set of data vectors [36, 109, 110].

Problem Statement. Assume a finite set of training samples $\boldsymbol{X} = [\boldsymbol{x}^1, \ldots, \boldsymbol{x}^N] \in \mathbf{R}^{m \times N}$. The classical dictionary learning estimate the dictionary $\boldsymbol{D} = [\boldsymbol{d}^1, \ldots, \boldsymbol{d}^p] \in \mathbf{R}^{m \times p}$ with p

elements or atoms, from the data, using empirical cost function

$$f_n(\boldsymbol{D}) \triangleq \frac{1}{N}\sum_{i=1}^{N}(\mathcal{L}_u(\boldsymbol{x}^i, \boldsymbol{D})) \tag{2.9}$$

where $\mathcal{L}_u(\boldsymbol{x}, \boldsymbol{D})$ is an unsupervised loss function which is small if the dictionary $\boldsymbol{D}$ is "good" at reconstructing input signals; $\boldsymbol{x} \approx \boldsymbol{D}\boldsymbol{\alpha}$ when $\boldsymbol{\alpha}$ is a sparse vector in $\mathbf{R}^p$. The vector $\boldsymbol{\alpha}$ may be called as the decomposition, or the code of the signal $\boldsymbol{x}$. Without enforcing any constraint on $\boldsymbol{D}$, the atoms may get large which leads to degenerate and small sparse codes $\boldsymbol{\alpha}$. To solve this issue, the ℓ_2 norm of each dictionary element $\{\boldsymbol{d}^i\}_{i\in[\![1;p]\!]}$ is regularized to be less than or equal to one. The convex set of all eligible dictionary candidates is shown as $\mathcal{D}$

$$\mathcal{D} \triangleq \{\boldsymbol{D} \in \mathbf{R}^{m\times p} \text{ s.t. } \forall k \in \{1, 2, \ldots, p\}, \|\boldsymbol{d}^k\|_2^2 \leq 1\} \tag{2.10}$$

Following elastic-net [217], the data-driven loss function is designed as

$$\mathcal{L}_u(\boldsymbol{x}^i, \boldsymbol{D}) \triangleq \underset{\boldsymbol{\alpha}^i\in\mathbf{R}^p}{\operatorname{argmin}} \frac{1}{2}\|\boldsymbol{x}^i - \boldsymbol{D}\boldsymbol{\alpha}^i\|_2^2 + \lambda_1\|\boldsymbol{\alpha}^i\|_1 + \frac{\lambda_2}{2}\|\boldsymbol{\alpha}^i\|_2^2 \tag{2.11}$$

with λ_1 and λ_2 as regularization parameters. Here, when $\lambda_2 = 0$, elastic-net would be same as Lasso or basis pursuit. Elastic-net formulation in (2.11) with $\lambda_2 > 0$ has been shown to be strongly convex with a unique solution that is Lipschitz with respect to $\boldsymbol{x}$ and $\boldsymbol{D}$ with a constant depending on λ_2 [104].

The problem (2.9) has the product of the two variables as $\boldsymbol{D}\{\boldsymbol{\alpha}^i\}_{i\in[\![1;N]\!]}$, and therefore the problem is not joint convex in the space of coefficients $\boldsymbol{A} = [\boldsymbol{\alpha}^1, \ldots, \boldsymbol{\alpha}^N]$ and the dictionary $(\boldsymbol{A}, \boldsymbol{D})$. However, when one of the two optimization variables is fixed, the problem (2.9) is convex with respect to the other variable [154, 108]. Since estimation of sparse codes takes most of the computation in each iteration, one may want to use a second-order optimization technique to learn the dictionary more accurately at each step when $\{\boldsymbol{\alpha}^i\}$ is fixed.

$$\underset{\boldsymbol{D}\in\mathcal{D},\boldsymbol{A}}{\operatorname{argmin}} \sum_{i=1}^{N}\left(\frac{1}{2}\|\boldsymbol{x}^i - \boldsymbol{D}\boldsymbol{\alpha}^i\|_{\ell_2}^2 + \lambda\|\boldsymbol{\alpha}^i\|_1\right) \tag{2.12}$$

The dictionary learning method which is purely based on minimizing reconstruction error has been shown to be equivalent to a large-scale matrix factorization problem. The optimization problem (2.12) for matrix factorization is written as

$$\underset{\boldsymbol{D}\in\mathcal{D},\boldsymbol{A}}{\operatorname{argmin}} \ \frac{1}{2}\|\boldsymbol{X}-\boldsymbol{D}\boldsymbol{A}\|_F^2+\lambda\|\boldsymbol{A}\|_{\ell_{11}}$$

where matrix of data and sparse codes are obtained by horizontally concatenating vectors as: $\boldsymbol{X}=[\boldsymbol{x}^1,\ldots,\boldsymbol{x}^N]$ and $\boldsymbol{A}=[\boldsymbol{\alpha}^1,\ldots,\boldsymbol{\alpha}^N]$, and ℓ_1 norm of the matrix $\boldsymbol{A}$ is shown as $\|\boldsymbol{A}\|_{\ell_{11}}$, where the result would be summation over absolute value of all coefficients.

Bottou et al. in [19] suggested to learn the dictionary by minimizing expected cost $\hat{f}(\boldsymbol{D})$ defined in Eq. (2.13). Minimizing empirical cost $f_n(\boldsymbol{D})$ with high precision obtains a dictionary that is sub-optimum to represent data in general. The reason lies in the fact that the empirical cost is an approximation of the expected cost. In [105] an inaccurate solution but with better expected cost for $\boldsymbol{D}$ is proposed in online scheme by

$$\hat{f}(\boldsymbol{D}) \triangleq \ \mathbb{E}_x[\mathcal{L}_u(\boldsymbol{x},\boldsymbol{D})]=\lim_{n\to\infty} f_n(\boldsymbol{D}) \text{ a.s.} \tag{2.13}$$

where the data $\boldsymbol{x}$ is assumed to be drawn from an (unknown) finite probability distribution $p(\boldsymbol{x})$. In other words, $\hat{f}(\boldsymbol{D})$ behaves as a surrogate for empirical cost $f_n(\boldsymbol{D})$. Also, it is demonstrated both theoretically and empirically in [19] that first order methods like stochastic gradient descent that has a poor rate of convergence in conventional optimization terms may in fact in certain scenarios be faster in reaching to a solution with low expected cost than second-order batch methods. In the presence of a large number of training data, it is less probable to have overfitting, but as a matter of speed or memory requirements, classical optimization techniques may become impractical. Interested readers to know more about other applications of first-order stochastic gradient descent in matrix factorization problems are referred to [91].

The dictionary learning methods like [135, 1], updates the dictionary at iteration τ using the classical first-order stochastic gradient descent and orthogonally projected onto the unit

norm ball $\mathcal{D}$ using operator $\Pi_{\mathcal{D}}$

$$\boldsymbol{D}^{(\tau)} = \Pi_{\mathcal{D}}[\boldsymbol{D}^{(\tau-1)} - \rho_\tau \nabla_{\boldsymbol{D}} \mathcal{L}_u\big(\boldsymbol{x}^{(\tau)}, \boldsymbol{D}^{(\tau-1)}\big), \tag{2.14}$$

where ρ_τ is the gradient step and $\boldsymbol{x}^{(\tau)}$ are i.i.d sample vectors drawn from the possible unknown and compact distribution $p(\boldsymbol{x})$. We follow the heuristic proposed in [104] to set the learning rate $\rho = a/(\tau+b)$, and a and b are based on the dataset. The obtained dictionary by minimizing optimization problem (2.13) leads to a dictionary that can properly reconstruct data and remove the noise. So, the dictionary is adapted to the data and has a good performance for reconstruction tasks like denoising [36] and restoration [109].

Although the unsupervised dictionary is learned in a data-driven fashion, it has been used for discriminative tasks like classification [185, 193]. The framework is to learn an unsupervised dictionary in training phase in a data-driven scheme. The learned dictionary is used to extract sparse code coefficients of the test signal using Lasso or basis pursuit Eq. (1.2) and Eq. (1.3). In [14, 185] the test signal is assigned to the class that can approximate it with minimum reconstruction error. But, utilizing class labels of the data in a misclassification error is more reasonable for the classification task. Therefore, some methods adopt sparse code $\boldsymbol{\alpha}^*(\boldsymbol{x}, \boldsymbol{D})$ as latent features for the training data $\boldsymbol{x}$ and learn a classifier in a classical expected risk minimization formulation

$$\underset{\boldsymbol{W}\in\mathcal{W}}{\operatorname{argmin}} \; f(\boldsymbol{W}) + \frac{\nu}{2}\|\boldsymbol{W}\|_F^2 \tag{2.15}$$

where $\mathcal{W}$ is a convex set of all acceptable classifier with parameters $\boldsymbol{W}$ and ν is the regularization parameter. The function f is a loss function over classifier parameters. Consider $\boldsymbol{y}^i$ as the label of the i-th training sample $\boldsymbol{x}^i$. Then, the loss function over $\boldsymbol{W}$ can be represented as:

$$f(\boldsymbol{W}) \triangleq \mathbb{E}_{\boldsymbol{y},\boldsymbol{x}}[\mathcal{L}_s\big(\boldsymbol{y}, \boldsymbol{W}, \boldsymbol{\alpha}^*(\boldsymbol{x}, \boldsymbol{D})\big)] \tag{2.16}$$

where $\mathcal{L}_s$ is the supervised convex loss function and in the literature based on the application, mostly square, logistic, or hinge loss from support vector machines are used [155]. In problem

(2.16) the expectation over the data and its label $(\boldsymbol{x}, \boldsymbol{y})$ should be calculated. However, the joint probability distribution $p(\boldsymbol{x}, \boldsymbol{y})$ is not known. In the case that we have a sufficient number of $\boldsymbol{x}$ and $\boldsymbol{y}$ in the training data we can expect a good sampling from the unknown probability distribution $p(\boldsymbol{x}, \boldsymbol{y})$.

2.2.2 Multimodal Case

We explained so far that the assumption in single modality case is that the data $\boldsymbol{x}$ is assumed to be drawn from an (unknown) finite probability distribution $p(\boldsymbol{x})$. The generalization of the assumption to the multiple modalities would be as follows:

(A) The joint probability density $p(\boldsymbol{X}, \boldsymbol{y})$ of the multimodal data in image and video processing and its corresponding variable $(\boldsymbol{X} = \{\boldsymbol{x}_m\}_{m=1}^{M}, \boldsymbol{y})$ can be supported by compact distribution. This is a valid assumption since sensors in the image and video data acquisition generate bounded values.

(B) For classification task of finite number of classes, $c \in \{1, \ldots, C\}$, for any label $\boldsymbol{y}$, the distribution $p(\boldsymbol{y}, .)$ is continuous and the supervised loss function $\mathcal{L}_s(\boldsymbol{y}, .)$ is twice continuously differentiable.

Problem Statement. The N multimodal input data $\{\boldsymbol{X}^i, \boldsymbol{y}^i\}_{i \in [\![1;N]\!]}$ are normalized and assumed statistically independent. The i-th sample that belongs to the c-th class is seen from M modalities: $\boldsymbol{X}_c^i = \{\boldsymbol{x}_{c,1}^i, \ldots, \boldsymbol{x}_{c,M}^i\}$. We want to learn a dictionary $\boldsymbol{D}_m$ for each modality m in $\{1, \ldots, M\}$ that is "good" to reconstruct the data (*i.e.* input data yield sparse representations over the dictionary) and "bad" to reconstruct the noise: $\boldsymbol{x}_{c,m}^i \approx \boldsymbol{D}_m \boldsymbol{\alpha}_{c,m}^i$, where $\boldsymbol{\alpha}_{c,m}^i \in \mathbf{R}^p$ is sparse representation of training data in m-th modality. The dictionary is obtained by extending $g_N(\boldsymbol{D}) \triangleq \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}_u(\boldsymbol{x}^i, \boldsymbol{D})$ to include joint sparse representation of different modalities in order to force similar pattern in different modalities. The problem is formulated as to find the multimodal sparse representation matrix $\boldsymbol{A}^i = [\boldsymbol{\alpha}_1^i, \ldots, \boldsymbol{\alpha}_M^i]$ in $\mathbf{R}^{p \times M}$ and the set of dictionaries with p elements or atoms, $\boldsymbol{D}_m = [\boldsymbol{d}_m^1, \ldots, \boldsymbol{d}_m^p]$ for m in

$\{1,\ldots,M\}$, given the multimodal sample i as $\boldsymbol{X}_c^i = \{\boldsymbol{x}_{c,m}^i\}_{m=1}^M$ and i in $\{1,\ldots,N\}$ and c in $\{1,\ldots,C\}$, by extension of the method presented in previous section to include multimodality

$$\mathcal{L}_{mu}(\{\boldsymbol{x}_m^i, \boldsymbol{D}_m\}) \triangleq \underset{\boldsymbol{A}^i,\{\boldsymbol{D}_m\}}{\operatorname{argmin}} \sum_{m=1}^{M} \frac{1}{2}\|\boldsymbol{x}_m^i - \boldsymbol{D}_m\boldsymbol{\alpha}_m^i\|_2^2 + \lambda_1\Omega(\boldsymbol{A}^i) + \frac{\lambda_2}{2}\|\boldsymbol{A}^i\|_F^2 \tag{2.17}$$

where λ_1 and λ_2 are the regularizing parameters, and $\mathcal{L}_{mu}$ is the multimodal unsupervised loss function. The Frobenius norm in Eq. (2.17) is defined as: $\sqrt{\sum_{i=1}^{p}\sum_{j=1}^{M}|\boldsymbol{A}_{i,j}|^2}$ where $\boldsymbol{A}_{i,j}$ is the element of $\boldsymbol{A}$ in i-th row and j-th column. The Eq. (2.17) has the Frobenius norm as an extra term comparing to the Eq. (2.7). This extra term is useful to prove the exitance of unique solution for the joint sparse optimization problem [104]. In the simpler case of having only one feature modality $M = 1$, Eq. (2.17) will be the well known elastic-net formulation [217].

By extending Eq. (2.13), the dictionary in m-th modality is obtained by minimization of expected cost with respect to $\boldsymbol{D}_m$:

$$\boldsymbol{D}_m \triangleq \underset{\boldsymbol{D}_m\in\mathcal{D}_m}{\operatorname{argmin}}\, \mathbb{E}_{\boldsymbol{x}_m}\left[\mathcal{L}_{mu}(\{\boldsymbol{x}_m, \boldsymbol{D}_m\})\right] \tag{2.18}$$

The convex set of all dictionaries can be defined as: $\mathcal{D} = \{\mathcal{D}_m\}_{m=1}^M$; where:

$$\mathcal{D}_m \triangleq \left\{\boldsymbol{D}_m \in \mathbf{R}^{n_m\times p} \middle| \forall j \in \{1,\cdots,p\}, \|\boldsymbol{d}_m^j\|_2 \leq 1\right\} \tag{2.19}$$

where the loss function $\mathcal{L}_{mu}$ is defined as Eq. (2.17). Note that the expectation in Eq. (2.18) is taken over a possible unknown probability distribution $p(\boldsymbol{x}_m)$.

The joint optimization problem (2.17) and (2.18) has the product of two optimization variables $\boldsymbol{D}_m\boldsymbol{\alpha}_m$; which implies that this problem is not joint convex in the space of variables. However, when one of the two optimization variables are fixed, the problem (2.17) is convex with respect to the other variable [104]. Hence, the problem (2.17) is solved by splitting to two sub-problems: 1. given dictionaries $\{\boldsymbol{D}_m\}_{m=1}^M$, estimate the multimodal sparse codes

$\{\boldsymbol{\alpha}_m^i\}_{m=1}^M$ for all i in $\{1,\ldots,N\}$ as described in 2.2.3 ; 2. given sparse codes $\{\boldsymbol{\alpha}_m^i\}_{i=1}^N$, update the corresponding dictionary of m-th modality $\boldsymbol{D}_m$, as described in (2.2.4);

2.2.3 Estimate Multimodal Sparse Codes

In this section, we fix $\{\boldsymbol{D}_m\}_{m=1}^M$ and treat them as data for the problem (2.17). We initialize the multimodal dictionaries $\{\boldsymbol{D}_m\}_{m=1}^M$ by training samples of all classes same as [77, 104]. The problem (2.17) is converted to (2.20) to find an optimal $\boldsymbol{A}^\star = [\boldsymbol{\alpha}_1^{i\star},\ldots,\boldsymbol{\alpha}_M^{i\star}]$ in $\mathbf{R}^{p\times M}$ for all i in $\{1,\ldots,N\}$:

$$\underset{\boldsymbol{A}^i}{\operatorname{argmin}} \sum_{m=1}^{M} \frac{1}{2}\|\boldsymbol{x}_m^i - \boldsymbol{D}_m\boldsymbol{\alpha}_m^i\|_2^2 + \lambda_1\Omega(\boldsymbol{A}^i) + \frac{\lambda_2}{2}\|\boldsymbol{A}^i\|_F^2 \tag{2.20}$$

Assume $\boldsymbol{Z} \in \mathbf{R}^{p\times M} = [\boldsymbol{z}_1,\ldots,\boldsymbol{z}_M]$ and $\boldsymbol{U} \in \mathbf{R}^{p\times M} = [\boldsymbol{u}_1,\ldots,\boldsymbol{u}_M]$ and both initialized as zero. We denote the proximal operator associated with the norm Ω as $\mathbf{prox}_{\lambda\Omega}$ that maps its domain, vector $\mathbf{p}$, to the vector $\mathbf{q}$, both in $\mathbf{R}^M$: $\mathbf{prox}_{\lambda\Omega}(\mathbf{p}) \triangleq \operatorname{argmin}_{\mathbf{q}} \frac{1}{2}\|\mathbf{p}-\mathbf{q}\|_2^2 + \lambda\Omega(\mathbf{q})$. Then in iteration τ we have:

$$\boldsymbol{A}^{(\tau+1)} = \mathbf{prox}_{\lambda_1 f}(\boldsymbol{Z}^{(\tau)} - \boldsymbol{U}^{(\tau)}) \tag{2.21a}$$

$$\boldsymbol{Z}^{(\tau+1)} = \mathbf{prox}_{\lambda_1\Omega}(\boldsymbol{A}^{(\tau+1)} + \boldsymbol{U}^{(\tau)}) \tag{2.21b}$$

$$\boldsymbol{U}^{(\tau+1)} = \boldsymbol{U}^{(\tau)} + \boldsymbol{A}^{(\tau+1)} - \boldsymbol{Z}^{(\tau+1)} \tag{2.21c}$$

where data-fidelity term $f(.) \triangleq \sum_{m=1}^M \frac{1}{2}\|\boldsymbol{x}_m^i - \boldsymbol{D}_m\boldsymbol{\alpha}_m^i\|_{\ell_2}^2 + \frac{\lambda_2}{2}\|\boldsymbol{\alpha}_m^i\|_2$ is smooth and differentiable. The optimization variables $\boldsymbol{A}^{(\tau)}$ and $\boldsymbol{Z}^{(\tau)}$ are the solution of minimizing the smooth and non-smooth part of the problem (2.17) at iteration τ, respectively and they will eventually converge to each other, ($\boldsymbol{U}^{(\tau+1)} = \boldsymbol{U}^{(\tau)}$). The proximal step of problem (2.21a) is defined for each modality independently as:

$$\mathbf{prox}_{\lambda_1 f}(\boldsymbol{z}_m^{(\tau)} - \boldsymbol{u}_m^{(\tau)}) = \underset{\boldsymbol{\alpha}_m}{\operatorname{argmin}}\ \lambda_1 f(\boldsymbol{\alpha}_m^{(\tau)}) + \frac{1}{2}\|\boldsymbol{\alpha}_m^{(\tau)} - (\boldsymbol{z}_m^{(\tau)} - \boldsymbol{u}_m^{(\tau)})\|_2^2 \tag{2.22}$$

f is smooth with gradient $\nabla_{\boldsymbol{\alpha}_m} f = -\boldsymbol{D}_m^\top(\boldsymbol{x}_m - \boldsymbol{D}_m\boldsymbol{\alpha}_m) + \lambda_2\boldsymbol{\alpha}_m$, we compute the solution to problem (2.22) in iteration $\tau+1$:

$$\boldsymbol{\alpha}_m^{(\tau+1)} = (\boldsymbol{D}_m^\top\boldsymbol{D}_m + \frac{1}{\lambda_1}\boldsymbol{I} + \lambda_2\boldsymbol{I})^{-1}(\boldsymbol{D}_m^\top\boldsymbol{x}_m + \frac{1}{\lambda_1}(\boldsymbol{z}_m^{(\tau)} - \boldsymbol{u}_m^{(\tau)})) \tag{2.23}$$

the method is designed to get high classification accuracy while $\{\boldsymbol{D}_m\}_{m=1}^M$ have small numbers of atoms; but, this may increase the chance of singularity in (2.23). However, $\lambda_1 > 0$ and $\lambda_2 > 0$ makes the denominator $(\boldsymbol{D}_m^\top\boldsymbol{D}_m + 1/\lambda_1\boldsymbol{I} + \lambda_2\boldsymbol{I})$ positive definite. We solve (2.23) for each modality separately and concatenate the results to make $\boldsymbol{A}^{(\tau+1)} = [\boldsymbol{\alpha}_1^{(\tau+1)}, \ldots, \boldsymbol{\alpha}_M^{(\tau+1)}]$. Next, we solve the proximal step over $\boldsymbol{Z}_{r\rightarrow}$ in (2.21b) for each row r of $\boldsymbol{A}$ and r in $\{1, \ldots, p\}$:

$$\mathbf{prox}_{\lambda_1\Omega}(\boldsymbol{A}_{r\rightarrow}^{(\tau+1)} + \boldsymbol{U}_{r\rightarrow}^{(\tau)}) = \underset{\boldsymbol{Z}_{r\rightarrow}}{\operatorname{argmin}}\ \lambda_1\Omega(\boldsymbol{Z}_{r\rightarrow}^{(\tau+1)}) + \frac{1}{2}\|\boldsymbol{Z}_{r\rightarrow}^{(\tau+1)} - (\boldsymbol{A}_{r\rightarrow}^{(\tau+1)} + \boldsymbol{U}_{r\rightarrow}^{(\tau)})\|_2^2 \tag{2.24}$$

The optimization problem (2.24) is solved in p independent optimizations corresponding to p rows, while each optimization is done on an M-dimensional vector, $\boldsymbol{Z}_{r\rightarrow}^\top$. Since the groups are ordered, each of the p optimization can be done in one iteration using the dual form [72], which means that proximal step (2.24) can be solved with the same computational cost as joint sparsity. By extension of optimization algorithm in [72], we solve the proximal step of (2.24) for optimization variable $\boldsymbol{Z}_{r\rightarrow}^\top$ in Algorithm (2). We solve the optimization problem (2.24) using the SPArse Modeling Software [72]. After $\boldsymbol{Z}$ is obtained, this iteration would be finished by updating $\boldsymbol{U}$ according to (2.21c).

2.2.4 Learn Dictionary

The optimization problem (2.18) with respect to $\boldsymbol{D}_m$ is solved using a sequence of updates in the classical projected stochastic gradient scheme. The multimodal data-driven dictionaries can be computed by extending Eq. (2.14) to multimodal case [13]

$$\boldsymbol{D}_m^{(\tau)} = \Pi_{\mathcal{D}_m}\left[\boldsymbol{D}_m^{(\tau-1)} - \rho_\tau\nabla_{\boldsymbol{D}_m}\mathcal{L}_u\left(\boldsymbol{x}_m^{(\tau)}, \boldsymbol{D}_m^{(\tau-1)}\right)\right], \tag{2.25}$$

where $\Pi_{\mathcal{D}_m}$ projects the dictionary $\boldsymbol{D}_m$ orthogonally to the convex set $\mathcal{D}_m$ defined as Eq. (2.19). We construct sparse representation of m-th modality by horizontally concatenating sparse codes of all training data from the same modality: $\boldsymbol{\Gamma}_m = [\boldsymbol{\alpha}_m^1, \ldots, \boldsymbol{\alpha}_m^N]$. The dictionary $\boldsymbol{D}_m = [\boldsymbol{d}_m^1, \ldots, \boldsymbol{d}_m^p]$ will be updated by solving the optimization (2.18) such that the ℓ_2-norm of each atom should not be greater than unit-norm (2.19). We obtain multimodal dictionaries using the Iterative Projection Method proposed in [149]. Note that, in this part, multimodal sparse codes are fixed. With $\boldsymbol{\Gamma}_m = [\boldsymbol{\alpha}_m^1, \ldots, \boldsymbol{\alpha}_m^N]$ and $\boldsymbol{Y}_m = [\boldsymbol{x}_m^1, \ldots, \boldsymbol{x}_m^N]$:

$$\underset{\boldsymbol{D}_m}{\operatorname{argmin}} \ \|\boldsymbol{Y}_m - \boldsymbol{D}_m\boldsymbol{\Gamma}_m\|_F^2 \ \text{ s.t } \boldsymbol{D}_m \in \mathcal{D} \tag{2.26}$$

Now, the dictionary is updated atom by atom. The q-th dictionary atom is updating and the problem is rewritten to (2.27).

$$\underset{\boldsymbol{D}_m^{q\downarrow}}{\operatorname{argmin}} \ Tr(\boldsymbol{D}_m{}^\top \boldsymbol{D}_m \boldsymbol{\Gamma}_m \boldsymbol{\Gamma}_m{}^\top - 2\boldsymbol{D}_m{}^\top \boldsymbol{Y}_m \boldsymbol{\Gamma}_m{}^\top) \quad \text{s.t } \|\boldsymbol{D}_m^{q\downarrow}\|_{\ell_2} \leqslant 1 \tag{2.27}$$

Let $\boldsymbol{\Theta} = \boldsymbol{\Gamma}_m\boldsymbol{\Gamma}_m{}^\top$, $\boldsymbol{\Upsilon}_m = \boldsymbol{Y}_m\boldsymbol{\Gamma}_m{}^\top$. The q-th dictionary atom is updated and the problem is reformulated as follows.

$$\underset{\boldsymbol{D}_m^{q\downarrow}}{\operatorname{argmin}} \ Tr(\boldsymbol{D}_m{}^\top \boldsymbol{D}_m \boldsymbol{\Theta}_m - 2\boldsymbol{D}_m{}^\top \boldsymbol{\Upsilon}_m) \quad \text{s.t } \|\boldsymbol{D}_m^{q\downarrow}\|_{\ell_2} \leqslant 1 \tag{2.28}$$

where $\boldsymbol{D}_m^{q\downarrow}$ is the q-th column vectors of $\boldsymbol{D}_m$. Let $\Theta_m[q,q]$ be the element in q-th column and q-th row of $\boldsymbol{\Theta}_m$, $\boldsymbol{\Theta}_m^{q\downarrow}$ be the q-th column vectors of $\boldsymbol{\Theta}_m$, and $\boldsymbol{\Upsilon}_m^{q\downarrow}$ be the q-th column vectors of $\boldsymbol{\Upsilon}_m$. According to the algorithm of dictionary updating proposed in [104], dictionary atom $\boldsymbol{D}_m^{q\downarrow}$ with corresponding $\Theta_m[q,q] > 0$, is updated and is normalized to have unit l_2-norm as follows:

$$\boldsymbol{D}_m^{q\downarrow} = \frac{\boldsymbol{\Upsilon}_m^{q\downarrow} - \boldsymbol{D}_m\boldsymbol{\Theta}_m^{q\downarrow}}{\Theta_m[q,q] + 1/\alpha} \tag{2.29a}$$

$$\Pi_{\mathcal{D}} = \{\boldsymbol{D}_m^{q\downarrow}\}_{q=1}^p = \begin{cases} \boldsymbol{D}_m^{q\downarrow} & \text{if } \|\boldsymbol{D}_m^{q\downarrow}\|_{\ell_2} < 1 \\ \frac{\boldsymbol{D}_m^{q\downarrow}}{\|\boldsymbol{D}_m^{q\downarrow}\|_{\ell_2}} & \text{otherwise} \end{cases} \tag{2.29b}$$

Algorithm 1 Multimodal dictionary learning and joint sparse modeling

Input: $\boldsymbol{x}_m^i,\ \forall m \in \{1 \cdots M\},\ \forall i \in \{1 \cdots N\}$, iter

Initialize $\boldsymbol{D}_m$ with samples of m-th modality of all classes.
for $k = 1$ to *iter* **do**
    Fix $\{\boldsymbol{D}_m\}_{m=1}^M$ and find $\boldsymbol{A}$ using (2.21)
    for Each data $i \in \{1, \ldots, N\}$ **do**
        Obtain multimodal $\boldsymbol{A} = [\boldsymbol{\alpha}_1^i, \ldots, \boldsymbol{\alpha}_M^i]$ using joint sparse modeling.
    end for
    for each modality $m \in \{1, \ldots, M\}$ **do**
        Construct $\boldsymbol{\Gamma}_m = [\boldsymbol{\alpha}_m^1, \ldots, \boldsymbol{\alpha}_m^N]$.
        Update dictionary $\boldsymbol{D}_m$ according to (2.29)
    end for
end for

It will converge after several iterations. Algorithm 1 shows the steps required to learn the multimodal unsupervised dictionary and the joint sparse modeling using joint sparsity regularization.

We show the superior performance of the proposed joint optimization problem of (2.17) and (2.18) for the task of HEp2 Cell classification. Specifically, we show that the dictionary learning method produces a set of discriminative dictionaries with few atoms for each modality. At the same time, multimodal sparse representations of each class are forced to share the same sparsity patterns at the column level, which is imposed by joint sparsity regularization. The optimization problem over multimodal dictionaries and multimodal sparse representations is solved jointly. This method can combine information from different feature types and force them to have common sparsity patterns for each class, which is presented in Fig. 2.1. The proposed method is evaluated on two publicly available HEp-2 datasets and obtained state-of-the-art performance.

2.3 Application: HEp-2 Cell Classification

Diagnosing the Autoimmune Diseases (ADs) plays an important role in the curing process, which needs regular examinations. The Indirect Immunofluorescence (IIF) imaging technique is applied to the HEp-2 cells of the serum, where the captured pattern represents the type and severity of the AD. The interest in classification of the HEp-2 cells using a variety of

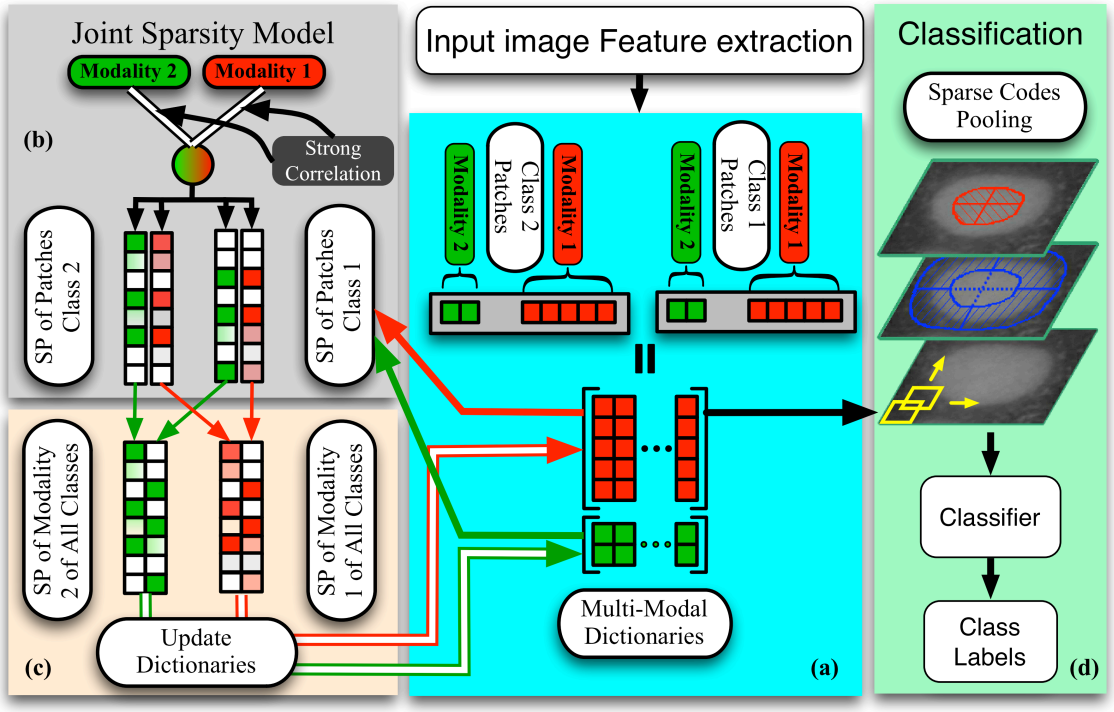


Figure 2.1: The illustration for the joint sparse modeling for classification task of two classes with two modalities: (a) The patches from two classes have $M = 2$ modalities that are shown as red and green. There is a color coded dictionary corresponding to each modality. The multimodal sparse representation of each patch is obtained by multimodal dictionaries $\{\boldsymbol{D}_m\}_{m=1}^{M}$ and joint sparsity regularization. The entries of sparse codes have different colors and represent different learned values; the white entries indicate the zero rows and columns. (b) The joint sparsity regularizer that is used to impose high correlation between the sparse representation of a sample in two modalities of {**red**, **green**}. (c) Modality-based sparse codes of all classes $\boldsymbol{X}_m = [\boldsymbol{x}_m^1, \boldsymbol{x}_m^2]$ is used to update $\boldsymbol{D}_m$. (d) The sparse code polling method is used to aggregate local sparse codes and train the SVM classifier in training stage.

machine learning algorithms specifically dictionary learning, and sparse coding methods are rapidly increasing.

2.3.1 HEp-2 Background and Related Work

The basic element of the body's immune system is a "Y" shape protein named "antibody", which is produced by the plasma cells. The main role of antibodies is to identify and mark the molecules of harmful agents, called "antigens". Antigens are foreign substances from the

environment, such as chemicals, bacteria, viruses, or pollen. In particular, the antibody uses its Y-shape tips to bind to the antigen and tags it for neutralization by the other parts of the immune system [120].

When the immune system fails to recognize a body's normal protein as "self", it produces another type of antibody, called "autoantibody", directed against that protein. This response of the immune system against individual's tissues is called "autoimmunity", and the related diseases are named Autoimmune Diseases (ADs). Antinuclear Antibodies (ANAs), which are found in many disorders including autoimmunity, cancer, and infection, are kind of antibodies that bind to contents of the cell nucleus. By screening the blood serum, the presence of ANA can be confirmed which in turn leads to a diagnosis of some autoimmune disorders. According to American College of Rheumatology, the golden standard test for detecting and qualifying ANAs is called Indirect Immunofluorescence (IIF) which uses the Human Epithelial Type-2 (HEp-2) tissue.

Immunofluorescence is an imaging technique that uses fluorescence microscope on microbiological samples that are stained with the fluorescent chemical compound. The IIF uses two antibodies, where the first antibody is unlabeled and binds to the target antigen. The second antibody, labeled with fluorophore, detects the first antibody and binds to it. One of the good properties of IIF is that multiple secondary antibodies can bind to the primary one and amplify the emitted light for each antigen, which results in a high contrast of the captured images [162].

The HEp-2 cell is a protein that contains hundreds of antigens used as an ideal substrate for the IIF test. Antibodies are first stained in HEp-2 tissue and then bound to a fluorescent chemical compound. Depending on the antibody present in the blood serum and the localization of the antigen in the cell, the patterns of fluorescence will be seen on the HEp-2 cells [50]. These patterns are then classified to diagnose ADs. Image intensity variation makes interpretation of fluorescence patterns very challenging. To make the pattern interpretation more consistent, automated methods for classifying the cells are essential. Several attempts have been made to facilitate the HEp-2 cell classification. It is shown in the literature that the

choice of classifier does not affect the final classification result as much as the type of features selected [57, 42]. To this end, a large number of intensity-based, statistical, morphological and engineered feature vectors are extracted including Grey Level Co-occurrence Matrix (GLCM) [47], Local Binary Pattern (LBP) and its modifications [134], wavelet transform [98], Scale Invariant Feature Transform (SIFT) [39, 80], etc.

However, there are some major drawbacks to these approaches. The large number of features extracted are not necessarily representative and/or discriminative, and the possibility of obtaining redundant features is very high [44]. This leads to a need for a post processing stage e.g. Principal Component Analysis (PCA) to reduce the feature dimension or Linear Discriminant Analysis (LDA) to make the features more discriminative. Lastly, on dealing with *intermediate* level images (see section 2.3.3), where the pixel values are much lower than *positive* intensity images, the intensity based methods [199, 139] are prone to misclassification and need a preprocessing stage to obtain representative features.

The performance of feature engineering based methods [134, 161] dominates the intensity based approaches because they are specifically designed and tuned for the problem at hand. However, there is no intervening procedure between the feature extraction stage and the classifier to make the features more representative and discriminative.

Recently, there has been an increasing interest in sparse coding and dictionary learning in computer vision and image processing research for classification task [40, 104, 175, 77, 37]. The input signal in sparse coding is reconstructed by a linear combination of a few columns (atoms) of the dictionary, which is a mapping function from feature space to low/high dimensional space. [38] proposed a method, where the SIFT and SURF features are extracted as the input features to learn a dictionary followed by spatial Pyramid Matching (SPM) [97] to provide the sparse representation of the input cell images. Then an SVM is learned to classify the test images. Intensity order based features are also extracted in [157] with the SPM sparse coding procedure followed by an SVM. Additionally, in [114, 115] the same procedure is used but Locality-constrained Linear Coding (LLC) is replaced with SPM for sparse coding scheme and a variety of features such as multi-resolution Local Pattern (mLP),

SIFT, Random Projection (RP) and Intensity Histogram (IH) are exploited to increase the final classification accuracy.

For each modality of the data, we learn and update a set of basis (dictionary) that can decompose the multimodal data into multimodal sparse codes that convey the prior information that we have about the structure between modalities. The proposed optimization problem has within each modality terms and some terms to make the connection between modalities. For each modality, we need the dictionary to be reconstructive and discriminative, so that it can successfully reconstruct the data while at the same time, has a poor performance in modeling the noise. The relation between different modalities in physical space is translated as grouping between their decomposition coefficient vectors in the space of sparse codes: the sparsity pattern of the multimodal sparse coefficient vectors is enforced to convey the desired prior information (here coupling structure between modalities). Our intuition was that this way provides codes that are more distinctive between different classes and so; better classification accuracy at the end.

While calculating the sparse codes of each image patch provides the local information stored in the patches, the spatial information is also essential for classification and this is obtained by aggregating the local information. A naive approach is to concatenate the features of all patches in each image to obtain a long vector of sparse codes. However, the final feature vector size for each image would be different due to the various number of patches for each image according to the image size. To this end, we introduced a novel pooling strategy to combine the patches' sparse codes that benefit from two important properties of small size feature vector and wisely selected image regions where their patches should be aggregated. This is performed by dividing the image into three layers as in Spatial Pyramid Matching (SPM) [97] (see Section 2.3.2) including whole image, a tube around the cell boundary and the inner side of the tube. The last two layers are then divided to 4 regions and the max-pooling operator is performed to combine the information of the image patches.

2.3.2 Sparse Codes Pooling

Patch based approach of calculating features and corresponding sparse codes result in obtaining local texture features, but we also need the spatial information for each image by aggregating the information of local patches. A naive solution is to concatenate the features of all patches in each image but this results in a long vector of sparse codes, which has two main problems. Firstly, the neighboring patch information is lost and secondly, the final size of the feature vector varies depending on the number of patches for each image. We describe how this issue is addressed in the proposed method in Fig. 2.1.*d* and Fig. 2.3.

Fig. 2.2 shows the Spatial Pyramid Matching (SPM) [97] method that divides the image into 1, 4 and 16 non-overlapping regions (21 regions in total) and performs max-pooling on the sparse codes in each region to finally produce a feature vector of size $(1+4+16)\times p$, where p is the number of atoms in the dictionary. A limitation of this approach is that the image is blindly divided into different layers without taking into account the underlying information in the image. As evident from Fig 2.2, the information pertaining to the cell boundary and inside cells are totally different but the SPM combines them nevertheless. Moreover, the SPM results in a large regions (e.g. 21 regions) and concatenating them all, produces a long feature vector for classification.

To alleviate these limitations, we propose a Sparse Codes Pooling (SCP) method which is shown in Fig. 2.3. "Layer 1" is the whole cell image and the information of all the image patches are pooled. The *distance transform* is applied on the cell mask, which assigns a value to each image pixel with the Euclidean distance to the nearest non-zero pixel. These non-zero pixels are including centroid of the cell, cell boundary and the image boundary (the bounding box of the cell). As can be seen in Fig. 2.3, two boundaries are extracted from the *distance image*, which are shown in blue circles in "Layer 2" and create the tube-shape region around the cell boundary. This layer is then divided to four regions as in SPM. "Layer 3" is created by using the inner circle of the "Layer 2" and also divided to four regions. The pooling strategy is then applied on the regions and all the feature vectors concatenated.

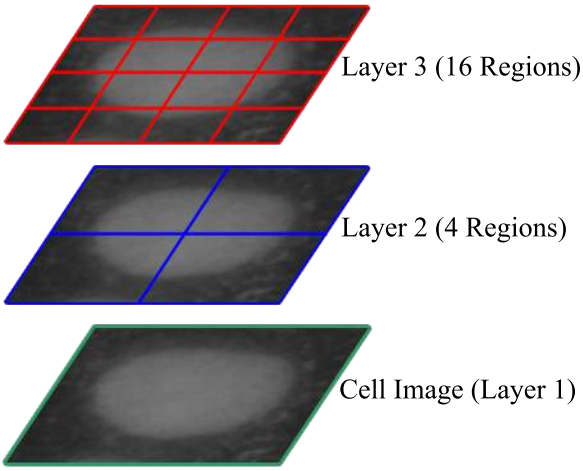


Figure 2.2: SPM method.

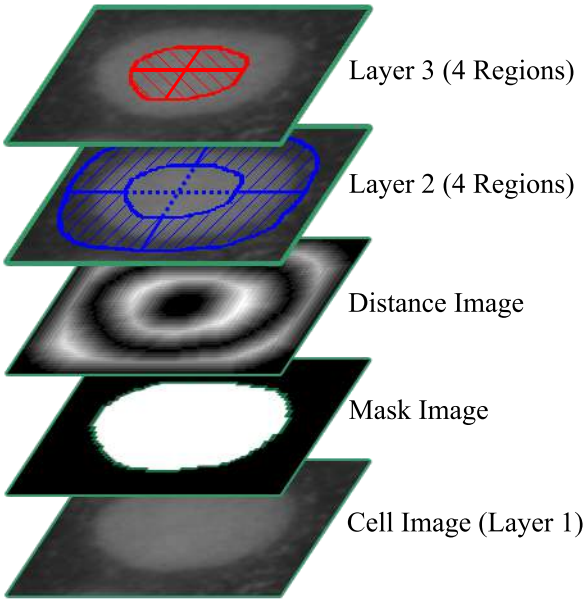


Figure 2.3: Proposed SCP method.

This approach benefits two main advantages. First, the final feature dimension vector is $9\times p$, which is around 57% lower than $21\times p$ in SPM. Second, the most informative area of the cells are near cell boundaries (e.g. Golgi and Nucleolar Membrane classes) and inner area of cells (e.g. Nucleolar and Speckled classes) as is evident in Fig. 2.4. By focusing on these two important areas, we can obtain more informative and discriminative feature vectors.

By considering the three image layers $l \in \{0, 1, 2\}$, a pooling function $\mathcal{F}$ is applied on the sparse codes $\boldsymbol{h}^l = [s_1^l, s_2^l, \cdots, s_{n_l}^l]$ in each layer, where s_i^l is the sparse codes of image patch i in layer l and n_l is the number of image patches in layer l. The final feature vector for layer l is $\boldsymbol{x}^l$.

$$\boldsymbol{x}^l = \mathcal{F}(\boldsymbol{h}^l) \tag{2.30}$$

The one-hot encoding, mean- and max- pooling functions are studied. In one-hot encoding, just one representative atom from dictionary is selected by having only one non-zero element in the final sparse code vector which is calculated as follows:

$$\boldsymbol{T}_l = \max\{\boldsymbol{h}^l\} \tag{2.31a}$$

$$\boldsymbol{x}_j^l = \begin{cases} 0 & if\ \boldsymbol{h}_{ji}^l < T_l \\ T_l & if\ \boldsymbol{h}_{ji}^l \geqslant T_l \end{cases} \quad i = \{1, 2, \cdots, n_l\} \tag{2.31b}$$

where T_l is the maximum sparse code of all patches in layer l and $\boldsymbol{x}_j^l$ is the j-th element of final feature vector.

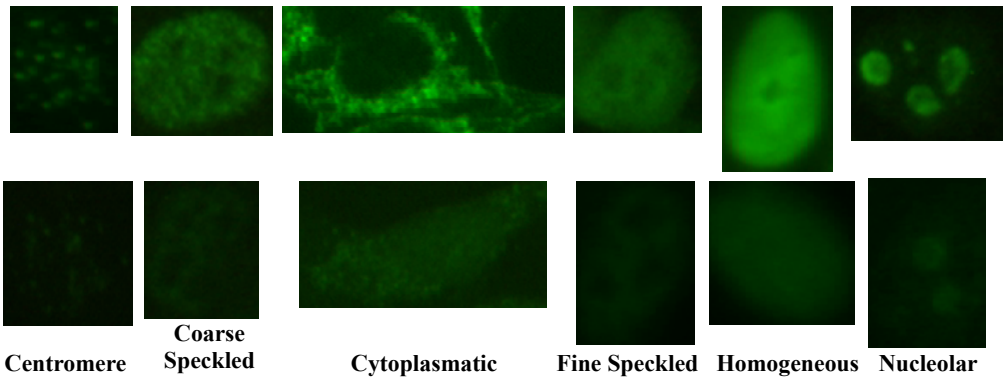
For mean- and max-pooling, the average and maximum values for each row of $\boldsymbol{h}^l$ is selected. For instance, the max-pooling function is:

$$\boldsymbol{x}_j^l = \max\{\boldsymbol{h}_{ji}^l\}, \quad i = \{1, 2, \cdots, n_l\} \tag{2.32}$$

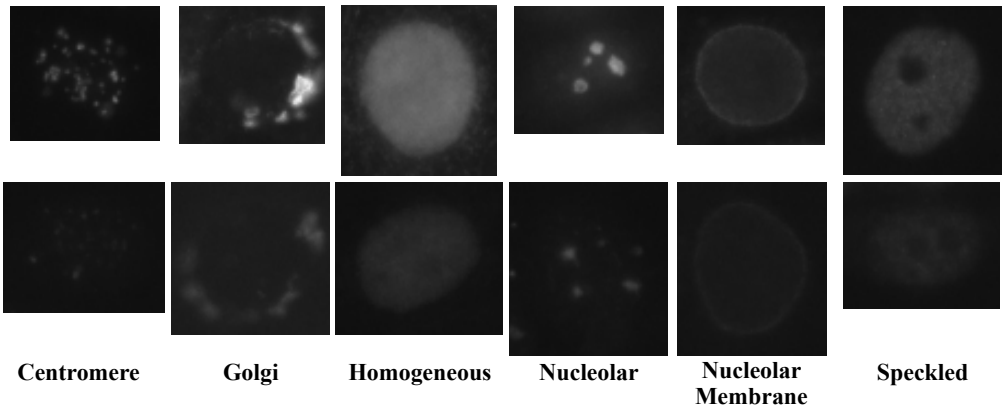
2.3.3 Dataset

For evaluation, two publicly available datasets namely ICPR2012 [43] and ICIP2013 [42] are used in the experiments. Both datasets contain many *cells* within each *specimen* image. The masks of the cells are also provided. The ICPR2012 has training and test images in six classes but in ICIP2013, the training cells and specimen images are available in six and seven classes respectively. Fig. 2.4.*a* and 2.4b show some samples of cell images for both datasets. It should be noted that cells in the ICPR2012 dataset are manually segmented but those in the ICIP2013 dataset are generated by automatic segmentation and the corresponding cell masks are prone to errors.

ICPR2012. This dataset consists of 28 HEp-2 specimen images where each image is in 1388×1038 resolution with 24-bit RGB pixels. The images are captured by using a fluorescence microscope (40-fold magnification) that is coupled with a 50W mercury vapor lamp and a digital camera. Each of the 28 images contains just one of the six staining patterns including Centromere (Ce), Coarse-speckled (Cs), Cytoplasmatic (Cy), Fine-speckled (Fs), Homogeneous (H) and Nucleolar (N) as illustrated in Fig. 2.4.*a*. The mask of each cell in each image and the labels of the cells are provided. Also, there are two levels of intensity images, which are called *intermediate* and *positive* images. In total, there are 1455 cells in



(a) ICPR2012 Dataset



(b) ICIP2013 Dataset

Figure 2.4: The *Cell Level* images of six classes for the ICPR2012 dataset in (a) and the ICIP2013 dataset in (b): First rows are the *positive* and second rows are the *intermediate* intensity level images.

the 28 images, which are divided to 721 images for training and 734 images for testing in our experiments [43].

ICIP2013. This dataset contains 419 sera of patients, which were prepared on the 18-well slide of HEP-2000 IIF assay with screening dilution 1:80. To capture the images, a monochrome high dynamic range microscopy camera is used. Approximately 100-200 cell images were extracted from each patient serum. In total, there were 68,429 cell images extracted: 13,596 cell images used for training, made available publicly, and 54,833 for testing, privately maintained by the organizers[1] [43].

[1] http://i3a2014.unisa.it/?page_id=126

Each annotated cell image contains information of cell pattern, intensity level (*positive* or *intermediate*), mask and the ID of image - which category the cell belongs to. Examples of these cell images can be seen in Fig. 2.4.*b*. Note that at the *Cell Level* the dataset has six classes including Centromere, Golgi, Homogeneous, Nucleolar, Nucleolar membrane (NuMem) and Speckled but at the *Specimen Level*, it has seven classes including one more Mitosis Spindle class is added.

Feature Extraction

We extract gradient based features of SIFT with size 128 and SURF with size 64 from each sample in an overlapping patches. The patch size is 12×12 and the distance between patches is 4 pixels. According to the size of the images, the number of patches is different. However, to train the dictionaries, we randomly select 100 patches from each image to get the balanced distribution of patch samples from all the input images.

Evaluation Strategies

The HEp-2 classification problem is divided into two categories, at the *Cell Level* and *Specimen Level*. In the *Cell Level* classification, each cell is classified solely without considering other neighboring cells. In contrast, the *Specimen Level* classification focuses on classifying whole specimen image containing many cells. As described in Section 2.3.3, two HEp-2 datasets are publicly available (ICPR2012 and ICIP2013) where the following experimental scenarios are exploited to evaluate the proposed method:

i. "Test set" evaluation, which can be done only on ICPR2012, for which the test set is publicly available but not for ICIP2013 for which a test set is not provided.

ii. "Leave-One-Specimen-Out (LOSO)", where all the cells from one specimen image are used for test and the rest of the specimen cells for training. This scenario is applied to both datasets.

iii. "HSM" evaluation method proposed by [57], 600 cells from each class (300 cells from Golgi class) are randomly selected for training and the rest of the cells are used for the test set. This strategy is only applied on ICIP2013 for comparison with other methods.

It should be noted that the cell masks for both datasets are provided but the masks are inaccurate specifically for the *Specimen Images* in ICIP2013. For instance, some masks contain non-cell areas and "touching cells" are not accurately divided. Therefore, the cell extraction method in [38] is used to get better cell masks by combining several morphological features.

To report the classification results, the Mean Class Accuracy (MCA) is used as suggested by dataset publisher [43]. In particular, MCA is defined by MCA= $\frac{1}{K}\sum_{k=1}^{K} CCR_k$, where CCR_k is the correct classification rate for class k and K is equals to the number of classes.

2.3.4 Results

The proposed JMCDL classification method is evaluated and the results are discussed in this section. We compare the proposed algorithm with the state-of-the-art HEp-2 cell classification methods that demonstrate the significant influence of enforcing different modalities to have similar sparsity pattern while learning multimodal dictionaries. We also investigate the effect of proposed SCP pooling strategy on the classification performance.

ICPR2012. Table. 2.1 and Table. 2.2 show the accuracies on ICPR2012 by using "Test set' and "LOSO" evaluation methods for both *Cell Level* and *Specimen Level* classifications, respectively.

The proposed JMCDL has two major components: 1. dictionary learning and, 2. joint sparsity regularization. We evaluate the performance of each novel component of the proposed method and the whole system on Tables 2.1 and 2.2. We express the performance of JMCDL without joint sparsity regularization to observe the effect of proposed dictionary learning. Since this scenario is equal to have only one feature modality, we call it Single-Cue Dictionary Learning (SCDL) and it includes three scenarios: surf only ("SURF"), sift only ("SIFT") and "SIFTSURF" that is made by putting together sift and surf features in

Table 2.1: The MCA accuracy on ICPR2012 dataset by using two evaluation strategies "Test set" and "Leave-One-Specimen-Out (LOSO)" for Cell Level classification (Task 1).

ICPR2012 (%)		Proposed					DL-based Methods			Other Methods	
		SCDL									
		JMCDL	JMC	SIFTSURF	SIFT	SURF	Ensafi⋆	SNPB°	Kastaniotis†	Nosaka◇	DiCataldo‡
Test set	Positive	**82**	78	76	74	72	81	**82**	70	79	60
	Intermediate	**79**	72	69	67	66	62	59	31	58	35
	Average	**80**	75	73	70	69	72	70	51	69	48
LOSO	Positive	**96**	92	90	86	82	91	92	72	80	95
	Intermediate	**84**	80	77	74	71	72	70	55	60	80
	Average	**90**	86	84	80	77	82	81	64	70	88

⋆[37] °[39] †[175] ◇[134] ‡[32]

one vector. The impact of joint sparsity regularization while dictionary is learned by [104] is reported as JMC and finally, the JMCDL reflects the performance of the whole system of joint dictionary learning and multimodal sparsity regularization by extracting SIFT and SURF features.

We compare classification accuracy of JMCDL with three state-of-the-art HEp-2 classifiers that are based on dictionary learning (DL) in "DL-based Methods" part of the Table: [37] use SIFT, [39] (SNPB) exploit both SIFT and SURF and [175] consider modified version of Local Binary Patterns (LBP) features. We also bring the performance of two state-of-the-art non-sparse based representation methods to compare with the JMCDL including the winner of the ICPR2012 contest[2] [134] and [32] that exploit LBP, morphological and textural features.

Table 2.1 shows that the proposed dictionary learning consistently outperforms other methods. Learning dictionary by elastic-net (JMC column) [218] while enforcing multimodal joint sparse regularization outperforms SCDL on average by 5% and 4% in "Test set"

[2] http://mivia.unisa.it/hep2contest/index.shtml

Table 2.2: The MCA accuracy on ICPR2012 dataset by using two evaluation strategies "Test set" and "Leave-One-Specimen-Out (LOSO)" for Specimen Level classification (Task 2).

ICPR2012 (%)	Proposed					DL-based Methods			Other Methods	
			SCDL							
	JMCDL	JMC	SIFTSURF	SIFT	SURF	Ensafi⋆	SNPB°	Kastaniotis†	Nosaka◇	DiCataldo‡
Test set	**93**	86	86	79	64	86	**93**	86	79	**93**
LOSO	**93**	88	86	79	64	79	86	79	86	**93**

⋆[37] °[39] †[175] ◇[134] ‡[32]

and "LOSO" evaluation methods. In "Test set" evaluation strategy, JMCDL improves the accuracy over SIFT and SURF by more than 10% and SIFTSURF by around 7%. Also, JMCDL shows superior results comparing to the DL-based and other methods particularly in *Cell Level*, where 80% and 90% accuracies are obtained in "Test set" and "LOSO" strategies, respectively. These results are 8% better than other DL-based methods in both evaluation strategies and 11% and 2% above the other methods.

Additionally, a significant achievement is obtained on intermediate intensity level classification, where the cell classification accuracy is improved by more than 10% in "Test set" and 4% in "LOSO" strategies.

For the *Specimen Level* classification, as shown in Table 2.2, a 93% accuracy is obtained which is similar to other best performances. The similar accuracy is mostly due to the limited number of *specimen* images (28 images only). It can be expected that the proposed method will achieve better results in comparison with other methods when the number of images increases, as observed in the ICIP2013 dataset to be discussed in the ensuing section.

The confusion matrix of Cell Level classification using LOSO evaluation is shown in Table 2.4a for ICPR2012 dataset.

ICIP2013. Comparison results for the ICIP2013 dataset is shown in Table 2.3. The "HSM" and "LOSO" evaluation strategies are used (see section. 2.3.3) for both *Cell* and

Table 2.3: The MCA accuracy on ICIP2013 dataset by using two evaluation strategies "HSM" [57] and "Leave-One-Specimen-Out (LOSO)".

ICIP2013 (%)			Proposed					DL-based Methods				Other Methods	
					SCDL								
			JMCDL	JMC	SIFTSURF	SIFT	SURF	Ensafi⋆	SNPB°	Gragnaniello†	manivannan◇	HSM‡	Larsen∇
HSM	*Cell Level*	Positive	**98.5**	96.9	96.1	92.3	84.3	95.8	96.8	-	-	95.5	-
		Intermediate	**93.2**	88.7	87.4	86.8	69.7	87.9	88.8	-	-	80.9	-
		Average	**95.9**	92.8	91.8	89.6	77	91.9	92.8	-	-	88.2	-
LOSO	*Cell Level*	Positive	**87.6**	86.8	86.1	82.8	78.2	83.4	83.8	-	-	-	-
		Intermediate	**77.5**	76.9	76.4	68.4	63.4	71.2	72	-	-	-	-
		Average	**82.6**	81.8	81.3	75.6	70.8	77.3	77.9	81.1	80.3	-	78.7
	Specimen Level		**91.6**	89.2	88	84.3	77.1	88	89.2	86.7	89.9	-	-

⋆[38] °[39] †[52] ◇[114] ‡[57] ∇[96]

Specimen Level classification tasks. For the *Cell Level* classification task, the positive and intermediate intensity level images are exploited.

The JMCDL method is compared with SCDL, DL-based and other methods. It is also compared with [114], the winner of I3A contest[3] (Pattern Recognition Techniques for Indirect Immunofluorescence Images) as hosted by International Conference on Pattern Recognition (ICPR) 2014.

The performance of proposed dictionary learning using "SIFTSURF" is promising since it performs slightly better than HSM and it can get close result to the SNPB based on HSM measurement. In addition, "SIFTSURF" outperforms all the state-of-the-art methods based on LOSO standard. Learning dictionary by elastic-net (JMC column) [218] outperforms SCDL. On the other hand, JMCDL obtains better average accuracy than "SIFTSURF" by 4.1% and 1.3% based on HAM and LOSO, respectively.

Table 2.3 also shows other DL-based methods, where JMCDL outperform the I3A contest winner [114] by 2.3% and [52] by 1.5%. JMCDL outperformed [38, 39] by 5%, which used SIFT and SURF features. These comparisons clearly show the advantage of multimodal

[3] http:\i3a2014.unisa.it

Table 2.4: The Cell Level confusion matrices by using Leave-One-Specimen-Out method.

	Ce	Cs	Fs	Cy	H	N
Ce	**94.34**	0.00	0.68	0.00	2.17	2.81
Cs	0.00	**85.24**	1.12	8.66	4.38	0.60
Fs	0.00	9.35	**85.12**	0.91	3.28	1.34
Cy	0.00	2.18	1.97	**94.19**	0.00	1.66
H	2.00	0.98	1.38	0.00	**94.32**	1.32
N	6.35	0.21	5.66	1.35	1.06	**85.37**

(a) ICPR2012

	Ce	G	H	N	NuMem	S
Ce	**87.18**	1.16	2.68	1.45	3.56	3.97
G	1.01	**78.08**	2.35	9.80	6.35	2.41
H	1.36	3.54	**77.96**	2.13	4.95	10.06
N	0.47	4.25	3.26	**87.03**	1.47	3.52
NuMem	4.74	2.36	3.89	0.53	**87.16**	1.32
S	9.65	1.74	7.36	1.84	1.20	**78.21**

(b) ICIP2013

dictionary learning and joint sparse model as applied on a large ICIP2013 dataset. For *Specimen Level* classification, the JMCDL outperforms the I3A contest winner [114] by 1.7% and [52] by 4.9%. The confusion matrix of Cell Level classification using LOSO evaluation is shown in Table 2.4b for ICIP2013 dataset.

Sparse Representation with Similar Pattern

The imposed joint sparsity model makes sparse codes more discriminative and hence produces better classification results. The similar patterns are shown in Fig. 2.5, where the first row shows cell sample of the six classes. The sparse representation of each cell class is provided for various features: SIFT, SURF and SIFTSURF. Also, the patterns of the sparse codes imposed by regularization function are presented in the last row. It is evident from Fig. 2.5 that the sparse codes patterns for different modalities are similar as imposed by the $\ell_{1,2}$ regularization term.

SCP Versus SPM

The effect of proposed SCP pooling strategy is studied and compared with SPM method for two datasets as shown in Table 2.5. The first two parts of the Table 2.5 compares the JMCDL with applying SCP and SPM, where the sparse coding and dictionary learning schemes are the same but differs in pooling method. It is evident that the max-pooling strategy outperforms others in both methods and the combination of JMCDL and SCP obtains better results than other methods. The last part of the Table. 2.5 shows the performance of sparse

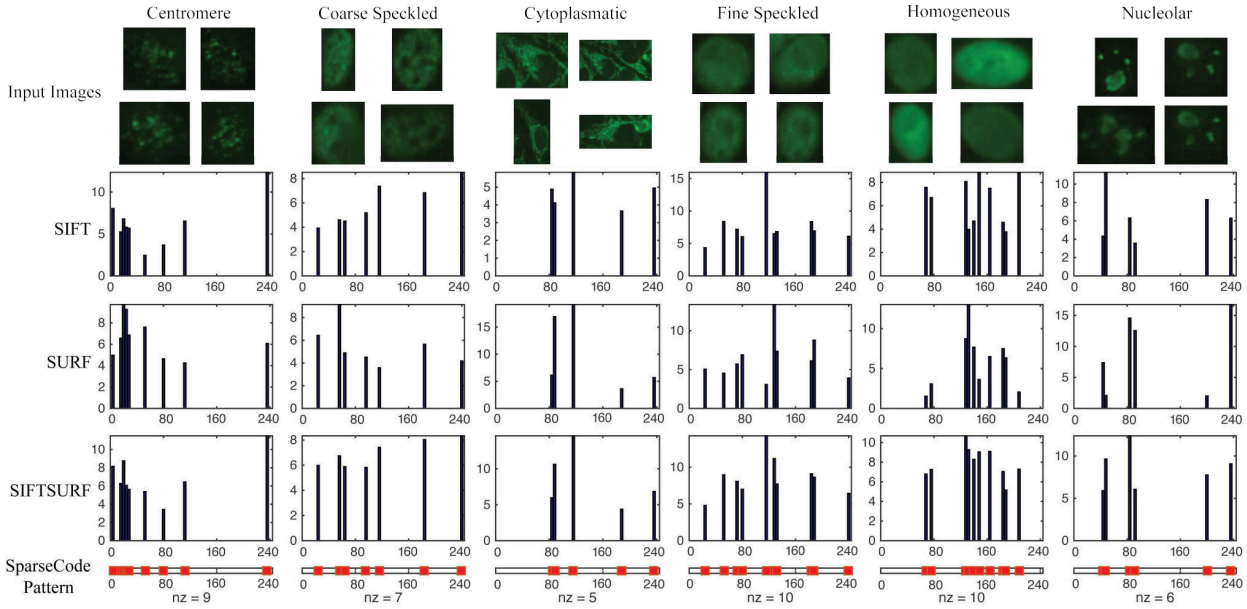


Figure 2.5: Representation coefficients generated by proposed regularization for SIFT, SURF and SIFTSURF features. There are six columns corresponding to the six classes. The x-axis is the dictionary columns and the x-axis is the sparse code values corresponding to each dictionary column. nz is the number of non-zero elements in the sparse code vector.

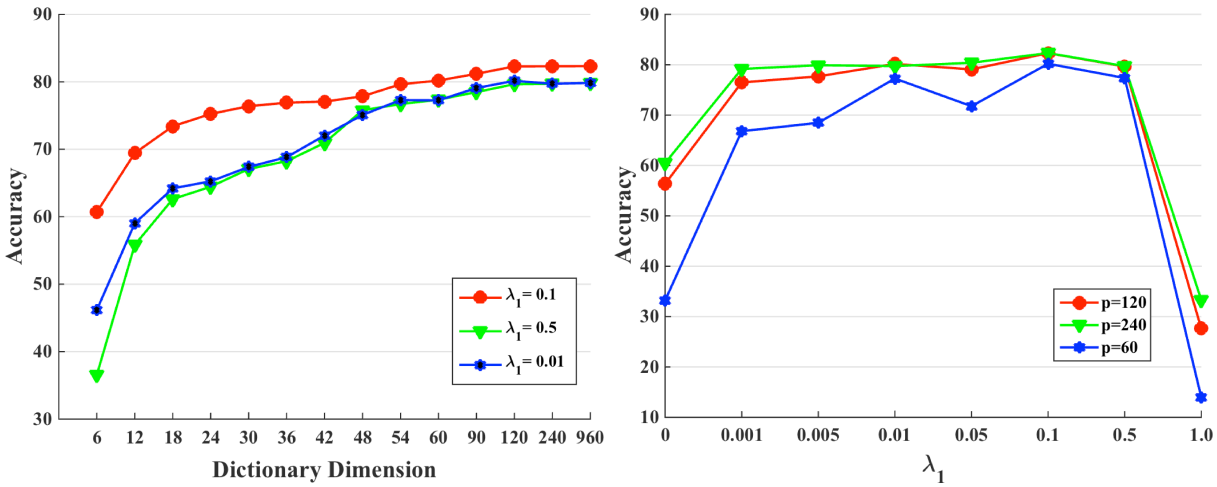
Table 2.5: The comparison of proposed SCP with SPM strategy by using different pooling functions and using LOSO evaluation method On Cell Level (Task 1).

	JMCDL+SCP			JMCDL+SPM			SPM		
	One-hot	Mean	Max	One-hot	Mean	Max	One-hot	Mean	Max
ICPR2012	66.7	84.2	**90.0**	61.3	80.2	86.7	58.1	78.6	82.1
ICIP2013	54.8	76.8	**82.6**	51.4	73.8	78.4	50.5	73.6	77.3

coding scheme combined with SPM as used in [37]. It is clear that JMCDL+SCP outperforms SPM by 7.9% and 5.3% on ICPR2012 and ICIP2013 datasets, respectively.

Parameter Study

In this section, two main parameters of the proposed method are analyzed. In particular, the dimension of the dictionary p plays a significant role where a larger number of atoms with much higher feature vector dimension creates an *over complete* dictionary. Such *over complete* dictionary is biologically inspired from human cortex and often gives better



(a) (b)

Figure 2.6: The accuracy of ICPR2012 positive test set versus different dictionary atoms in (a) and λ_1 values in (b).

classification accuracy [148]. On the other hand, calculating the *over complete* dictionaries are computationally expensive. Fig. 2.6.*a* shows the classification performance with different dictionary dimensions. It is obvious from Fig. 2.6.*a* that the performance keeps improving with the increase of the dictionary dimension until the dictionary dimension reaches 240 where the best performance is obtained.

The other most impactful parameter is regularization coefficient λ_1 in equation 2.6. Fig. 2.6.*b* shows the classification performance versus the regularization parameter. When the λ_1 is near zero, the reconstruction error influences more and provides non-sparse codes. By increasing λ_1 value, the sparsity of the weights helps increase the accuracy. However, while the λ_1 keeps increasing, the sparseness of the codes dominates the reconstruction error which reduces the classification accuracy. It can be seen that the best accuracy is obtained when $\lambda_1 = 0.1$.

2.4 Conclusion

Our main contributions in this Chapter are as follows:

- A new multimodal dictionary learning method is proposed that produces discriminative dictionaries with few atoms from many training samples, where one dictionary is trained in all-against-all fashion for each modality.

- Our goal is to show that in the presence of multimodal data where each sample is seen from highly related feature modalities with various sizes (here, SIFT and SURF), we can get better classification accuracy by encoding the a priori known correlation between feature modalities in space of sparse codes. The correlation (or relation) between different feature modalities is translated in space of sparse codes as the similarity between zero/nonzero pattern of the channels. This is done using the notion of grouping in space of sparse codes and applied with the joint sparse regularization to enforce the multimodal sparse representations of each class to share the same sparsity patterns at the column level of the corresponding dictionaries.

- The optimization problem includes two terms: 1. A data-fidelity term which is convex and continuously differentiable with Lipschitz-continuous gradient; and 2. A non-smooth norm-based regularization that models the high-order prior information of coupling between modalities. We expect a similar pattern between the sparse representation of the modalities that are grouped together: either all elements of a group contribute in decomposition, or that none of them participate. The regularization desires less number of groups to be involved in the decomposition, while data-fidelity term prone to reconstruct the multimodal signal with all groups selected.

- The HEp-2 cell classification task is studied in the sparsity scheme. The imposed joint sparsity enabled the algorithm to fuse information at feature-level by forcing their sparse codes to have similar basis. This is done using $\ell_{1,2}$ regularization that enforces high amount of correlation between different modalities of each cell class. In other words, we know a priori that the modality configuration (here, SIFT and SURF) induces a strong group structure that is encoded in the optimization using $\ell_{1,2}$

regularization (joint sparsity). JMCDL obtained better performance in comparison with other state-of-the-are results in both *Cell* and *Specimen Level* HEp-2 classification.

Chapter 3

Tree-Structured Hierarchical Coding

3.1 Introduction

As discussed in Chapter 1, making decision relying on a single source can jeopardize the decision making process. One solution is to use multiple sources of information when it is possible. Fusion of information from different sensor modalities can be more robust to single sensor failure. The information fusion is split into two broad categories: feature fusion [151] and classifier fusion [153]. In feature fusion, we have features at the input and output of the fusion process. The goal is to make or improve a new feature type from input features. The fusion system has various extracted features from each source at the input level. In classifier fusion, a classifier that is trained based on each feature type makes its decision. The fusion system combines input decisions to obtain better or new decisions.

In Chapter 2, we discussed sparse representation classification (SRC) for single modality and multiple modalities. We explained how important this framework is in the computer vision community for the both reconstructive and discriminative tasks. In SRC, the dictionary is made by concatenation of all training samples of all classes which means it does not need to be carefully designed features. But, the accuracy of classification depends strongly on a sufficient number of training samples from each class so that the distribution of

each class can be approximately obtained. We improved SRC to include learning a modality-secific dictionary as the optimization of a smooth non-convex objective function over a convex set. Specifically, within each modality, we need the dictionary to be reconstructive, so that it can successfully reconstruct the data while at the same time, remove the noise. Also, the dictionary of each modality should decompose the input data to sparse coefficients that are distinctive enough between the classes, that even a simple linear classifier that is trained over the sparse codes can generate high classification accuracy. The connection between modalities is made by applying joint sparsity regularization to generate highly discriminative multimodal codes as features so that different classes of data can be easily distinguished from multimodal standpoint.

The proposed multimodal fusion in chapter 2 has some limitations. Let us recall the proposed method in chapter 2.

Recall. The goal is to learn a reconstructive and discriminative dictionary $\boldsymbol{D}_m$ for each modality m in $\{1, \ldots, M\}$ by extending $g_N(\boldsymbol{D}) \triangleq \frac{1}{N}\sum_{i=1}^{N} \mathcal{L}_u(\boldsymbol{x}^i, \boldsymbol{D})$ to include joint sparse representation of different modalities in order to force similar pattern in different modalities:

$$\mathcal{L}_{mu}(\{\boldsymbol{x}_m^i, \boldsymbol{D}_m\}) \triangleq \underset{\boldsymbol{A}^i \in \mathbf{R}^{p\times M}}{\operatorname{argmin}} \sum_{m=1}^{M} \frac{1}{2}\|\boldsymbol{x}_m^i - \boldsymbol{D}_m\boldsymbol{\alpha}_m^i\|_2^2 + \lambda_1\Omega(\boldsymbol{A}^i) + \frac{\lambda_2}{2}\|\boldsymbol{A}^i\|_F^2 \tag{3.1}$$

where λ_1 and λ_2 are the regularizing parameters, and $\mathcal{L}_{mu}$ is the multimodal unsupervised loss function.

The fusion between observations of the sample $\{\boldsymbol{x}_m^i\}_{m=1}^M$ is enforced at the sparse coding space using joint sparsity regularization, $\Omega(\boldsymbol{A}) = \|\boldsymbol{A}\|_{\ell_{12}}$, which promotes a solution with sparse non-zero rows; hence, similar support is enforced on $\boldsymbol{A}$ at the column level of each dictionary $\boldsymbol{D}_m$.

At the same time, the dictionary in m-th modality is obtained by minimization of expected cost with respect to $\boldsymbol{D}_m$:

$$\boldsymbol{D}_m \triangleq \underset{\boldsymbol{D}_m \in \mathcal{D}_m}{\operatorname{argmin}} \mathbb{E}_{\boldsymbol{x}_m}\left[\mathcal{L}_{mu}(\{\boldsymbol{x}_m, \boldsymbol{D}_m\})\right] \tag{3.2}$$

where the loss function $\mathcal{L}_{mu}$ is defined as Eq. (3.1). The convex set of all dictionaries can be defined as: $\mathcal{D} = \{\mathcal{D}_m\}_{m=1}^{M}$; where:

$$\mathcal{D}_m \triangleq \left\{ \boldsymbol{D}_m \in \mathbf{R}^{n_m \times p} \middle| \, \forall j \in \{1, \cdots, p\}, \|\boldsymbol{d}_m^j\|_2 \leq 1 \right\} \tag{3.3}$$

Joint Sparsity Issues. The joint sparsity regularization promotes a solution with sparse non-zero rows in $\boldsymbol{A}$ and $r \in \{1, \ldots, p\}$. It is important to note that applied ℓ_2 norm inside each group does not promote sparsity. Joint sparsity is a *set-partitioning problem* that represent independent grouping between all the modalities of a signal in the space of sparse codes and relies on the idea that all views/features have highly correlated sparsity pattern. Particularly, joint sparsity is based on a strong statistical co-occurrence structure: in order to assign a sample to a class, most of its modalities/features should vote for that class, so knowing the label of one feature modality can act as a strong prior for inferring the label of others. However, this is not a valid assumption for application like visual tracking that features have different noise levels and significantly limits the performance of the method when feature modalities have perturbation or become unreliable. Plus, outlier tasks often exist that do not share a common set of features with the majority of tasks. Imposing strong correlation between all the feature modalities without considering how effective each feature was during the process, for instance, target tracking in previous frames, is suboptimal. Furthermore, since the features are originated from different spaces (*e.g.* color, edge), we can expect them to reconstruct the multimodal input $\{\boldsymbol{x}_m^i\}_{m=1}^{M}$ with different sparsity levels in the space of sparse codes.

In this chapter, we propose a new and robust multimodal fusion framework by formulating the relation between modalities in physical space to the embedded space of sparse codes as a tree-based hierarchy. This leads to a hierarchical coding that is able to capture multiple levels of cross-modality correlations while prohibiting misleading co-adaptations between data representations. We improve joint sparsity regularization to include fusion between features when multimodal dictionaries are embodied in a hierarchical tree structure and provide an exact solution to the problem. We demonstrate how powerful is our method by

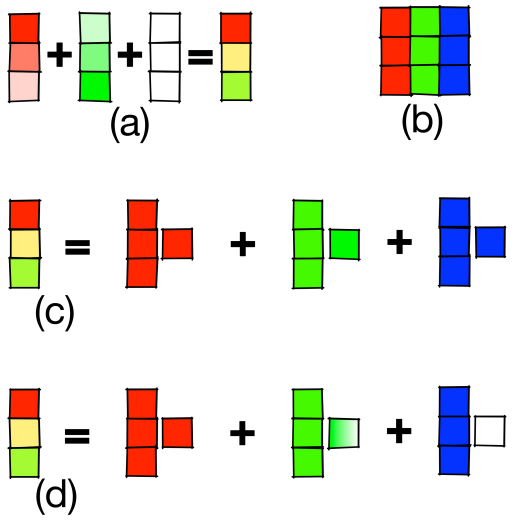


Figure 3.1: Illustration of independent vs overlapped coupling. Consider the case with $M = 3$ modalities of **red**, **green** and **blue**. (a) shows a multimodal signal $\boldsymbol{X} = \{\boldsymbol{x}_{red}, \boldsymbol{x}_{green}, \boldsymbol{x}_{blue}\}$ that has a mixture information of mostly red and a smaller amount of green. (b) a multimodal atom $\{\boldsymbol{d}\} = \boldsymbol{d}_{red}, \boldsymbol{d}_{green}, \boldsymbol{d}_{blue}$. The goal is to decompose the multimodal input $\boldsymbol{X}$ using $\{\boldsymbol{d}\}$ to multimodal coefficients $\boldsymbol{A} = [\boldsymbol{\alpha}_{red}, \boldsymbol{\alpha}_{green}, \boldsymbol{\alpha}_{blue}]$. (c) the result of independent coupling using ℓ_{12}. All three values of decompositions are equal. (d) the result of overlapping coupling. The $\boldsymbol{\alpha}_{red} \gg \boldsymbol{\alpha}_{green}$ and $\boldsymbol{\alpha}_{blue} = 0$.

comparing it to the state-of-the-art fusion techniques that are based on joint sparsity and demonstrate its performance quantitatively and qualitatively for the task of visual tacking.

Our design is based on the intuition that the higher-order prior knowledge about the structure of dictionary atoms are useful to limit the nonzero sparsity patterns. In Fig. (3.1) an intuitive representation of high-order information is given. This is due to the fact that in many applications of image processing and/or computer vision, there is a higher-order prior knowledge that models the potential relationships between the variables. For instance, the pixels of an image have a spatial relationship, or series of frames in a video are temporally connected. Enforcing the inherent structural information about the problem at hand using a norm based regularization is our desire in this dissertation.

In the presence of multimodal data, the i-th dictionary element is a multimodal feature from M modalities. We write this in either way of $\{\boldsymbol{d}_m^i\}_{m=1}^M$, or $\{\boldsymbol{d}_m^i\}_{m\in[\![1;M]\!]}$. That is to say; the atoms are partitioned into predefined groups corresponding to various types of features. One can expect a similar pattern between the sparse representation of the modalities that are

grouped together. In other words, either all elements of a group contribute in decomposition of the multimodal signal, or that none of them participate. We encode the prior information in the regularization with norms that support structural sparsity, to encourage the solutions of sparse regularized problems to promote the desired patterns of non-zero coefficients. In this chapter, we focus on hierarchical sparse coding as a particular class of structured sparsity: the multimodal atoms are considered to be configured in a directed tree $\mathcal{G}$, and the sparsity patterns are forced to make a connected and rooted subtree of $\mathcal{G}$, see Fig. (3.2).

Problem Formulation. Assume N multimodal signals as $\{\boldsymbol{X}^i\}_{i\in[\![1;N]\!]}$, which each has M modalities $\boldsymbol{X}^i = \{\boldsymbol{x}^i_{m,c}\}_{m\in[\![1;M]\!]}$, whose m-th modality has size n_m, $\boldsymbol{x}^i_{m,c} \in \mathbf{R}^{n_m}$. For each modality, there is a dictionary $\boldsymbol{D}_m$ in $\mathbf{R}^{n_m\times p}$ that has p elements, or atoms $\boldsymbol{D}_m = [\boldsymbol{d}^1_m, \ldots, \boldsymbol{d}^p_m]$. The dictionary $\boldsymbol{D}_m$ decomposes the corresponding modality of the signal $\boldsymbol{x}^i_{m,c}$ to coefficients $\boldsymbol{\alpha}^i_{m,c}$. That is, given a set of dictionaries $\{\boldsymbol{D}_m\}_{m\in[\![1;M]\!]}$, a multimodal input data $\boldsymbol{X}^i_c = \{\boldsymbol{x}^i_{m,c}\}^M_{m=1}$ is reconstructed using the multimodal coefficient vectors $\boldsymbol{A}^i_c$ in $\mathbf{R}^{p\times M}$ as $\boldsymbol{A}^i_c = [\boldsymbol{\alpha}^i_{1,c}, \ldots, \boldsymbol{\alpha}^i_{M,c}]$. The goal is to learn jointly the multimodal dictionaries and decomposition matrices $(\{\boldsymbol{D}_m, \boldsymbol{A}^i_{m,c}\}_{m\in[\![1;M]\!]})$ for all i in $\{1, \ldots, N\}$, so that we can approximately reconstruct the input from each modality as $\boldsymbol{x}^i_{c,m} \approx \boldsymbol{D}_m\boldsymbol{\alpha}^i_{c,m}$, while the non-zero coefficients of the multimodal decomposition vectors, $\boldsymbol{\alpha}^i_{c,1}, \ldots, \boldsymbol{\alpha}^i_{c,M}$, to form a connected and rooted subtree of the given tree.

3.2 Tree-Structured Hierarchical Groups

To overcome joint sparsity drawbacks, we generalize joint sparsity to a more elaborate hierarchical scheme. That is, we let features to be members of multiple groups that are overlapped and are embedded in a tree-shaped structure, as shown in Fig. (3.2). The tree-structure sparsity norm Ω is defined as

$$\Omega(\boldsymbol{A}) \triangleq \sum_{r=1}^{p}\sum_{g\in\mathcal{G}}\left(\sum_{m\in g}(\omega^{(g)}_m)^2|\boldsymbol{A}_{rm}|^2\right)^{\frac{1}{2}} \tag{3.4}$$

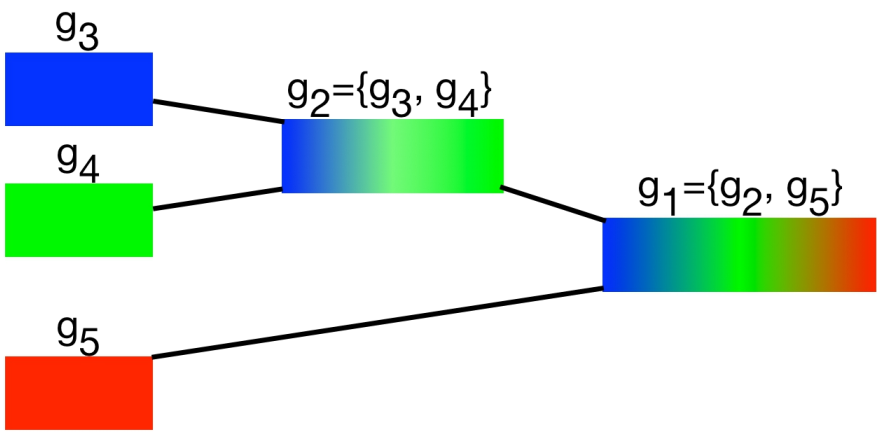


Figure 3.2: Illustration of a hierarchical structure between various modalities of input data. The tree is $\mathcal{G} = \{g_5, g_4, g_3, \{g_3, g_4\}, \{g_2, g_5\}\}$. It has three leaves of $\{g_3, g_4, g_5\}$, and g_2 enforces coupling between g_3, g_4, and the root of the tree g_1 enforces grouping between g_2 and g_5. Internal nodes near the leaves of the tree correspond to modalities that we expect highly related while the internal nodes near the root represent weakly-correlated sparse codes in its subtree. Any path from leaves to the root, is a possible solution.

The norm Ω computes the linear summation of the ℓ_2 norms of overlapping groups of sparse codes $\{\boldsymbol{A}^{1\downarrow}, \ldots, \boldsymbol{A}^{M\downarrow}\}$, with coefficients in each group being weighted by $\boldsymbol{\omega}^{(g)}$ and $g \in \mathcal{G}$: the M dimensional vector $\boldsymbol{\omega}^{(g)} = [\omega_1^{(g)}, \ldots, \omega_M^{(g)}]^\top$ is zero for indices of features that are not member of $g \in \mathcal{G}$; *i.e.* $\omega_m^{(g)} > 0$ if $m \in g$ and is zero otherwise. For instance, in Fig. (3.2), the tree structure $\mathcal{G}$ is given for coupling between $M = 3$ modalities: $\mathcal{G} = \{g_1, g_2, g_3, g_4, g_5\}$, where g_3, g_4 and g_5 are leaves of the tree that apply ℓ_1-norm sparsity on each modality. In this level, no sparsity pattern is enforced among modalities. In higher granularity, g_2 represents grouping between g_3 and g_4. Finally, g_1 is the root of the tree and seeks similar sparsity between g_2 and g_5. That is said, the size of the tree is $|\mathcal{G}| = 5$.

The set of solutions to this problem are all possible paths from the leaves to the root. The sparsity constraint guarantees that only a small number of these paths are selected in representing the input signal. This effectively allows the sparse representation to select the most relevant subset of modalities that best represent the given signal. For instance in Fig. (3.3), only case (a) that all modalities are zero, the root is not selected. In case (b), variables in g_3, g_4 are zero, the support of the solution consists g_5.

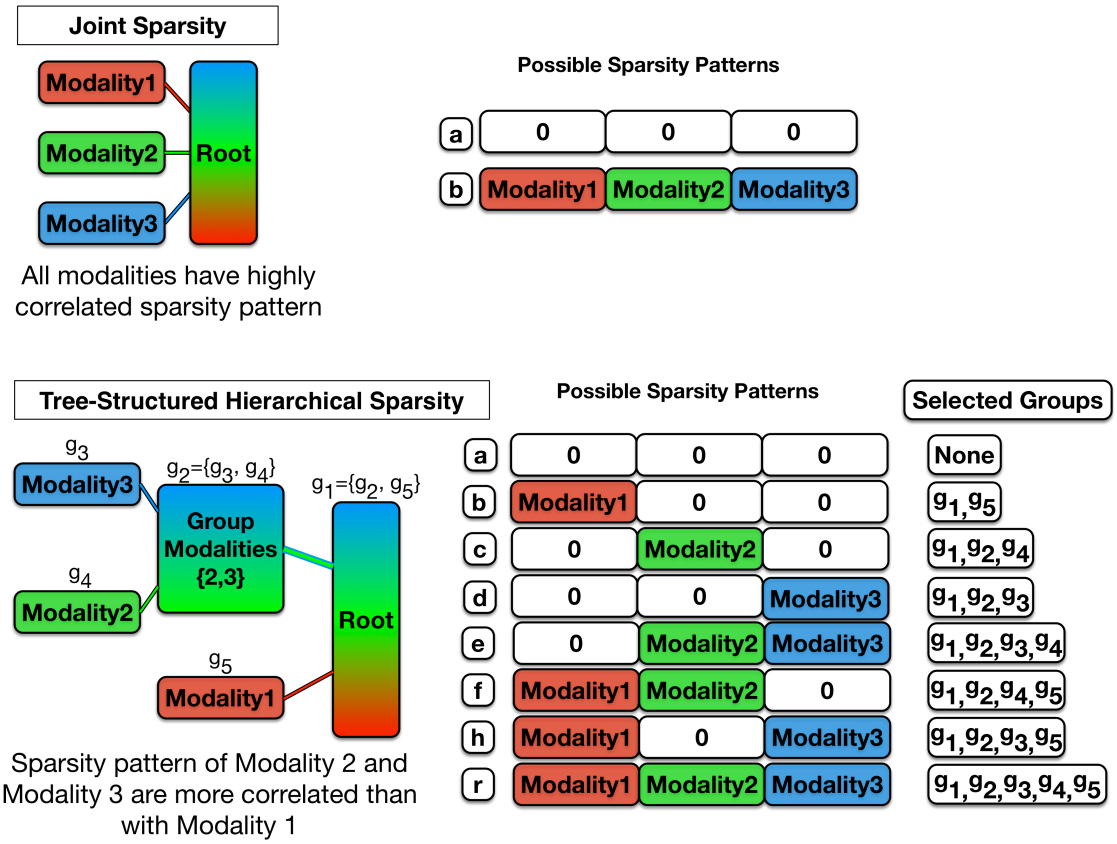


Figure 3.3: Illustration of joint sparsity vs tree-structured grouping. Consider the case with $M = 3$ modalities of **red**, **green** and **blue**. Top: ℓ_{12} all modalities are in one group. Down: The tree-structure regularization enforces hierarchical fusion between various modalities of input data in the space of sparse codes. The tree is $\mathcal{G} = \{g_5, g_4, g_3, \{g_3, g_4\}, \{g_2, g_5\}\}$. It has three leaves $\{g_3, g_4, g_5\}$, and g_2 enforces grouping between g_3, g_4, and the root of the tree g_1 enforces grouping between g_2 and g_5, and is a hierarchical grouping between red and the group of blue and green. The key here is that partially correlated coupling is not allowed in the tree structure. The groups of variables either are independent or one is subset of the other.

We extract four weak modalities (left and right periocular, nose and mouth) and one strong modality (face) which are shown in Fig. (3.4). The idea is to exploit different levels of correlation between weak and strong modalities for the task of face recognition. The tree $\mathcal{G}$ has $|\mathcal{G}| = 7$ nodes, that includes 5 leaves corresponding to the M modalities and 2 internal nodes. Each internal node encodes a possible grouping between leaves of the subtree which internal node is their root [87]. Here, one internal node represents the high correlation

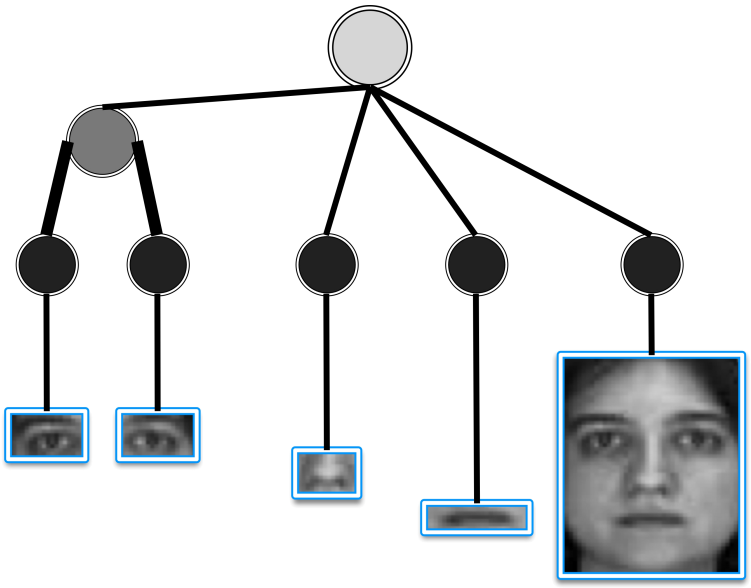


Figure 3.4: We employ the blue rectangular masks and cropping out the corresponding areas. These, along with the whole face, were taken for fusion. Simple intensity values were used as features for all of them. Tree-structure $\mathcal{G}$ corresponding to the four weak modalities of left periocular, right periocular, nose, and mouth, and a strong modality face. The tree represents a set of groups $\mathcal{G}$ between left and right periocular and all modalities at the root.

between left and right periocular and the other internal node is the root of the tree that model the grouping between nose, mouth, face and the group of eyes.

Our intuition is that leveraging this hierarchical structure will enable the multimodal subspaces to capture precisely the dependence/independence relations across modalities in the space of sparse codes. According to [72], this hierarchical structure makes following interpretation possible: the multimodal signal $\boldsymbol{X}^i = \{\boldsymbol{x}_m^i\}_{m=1}^M$ from modality m, $\boldsymbol{x}_m^i$, can be reconstructed using a subspace $\boldsymbol{d}_m^k$, (i.e., $\boldsymbol{\alpha}_m^i \neq 0$), only if all of its parent subspaces $\boldsymbol{d}_{\tilde{m}}^k$ are participating as well, where $\tilde{m}$ are the indices of the parent of the m-th node. The set of solutions to this problem are all possible paths from the leaves to the root. The sparsity constraint guarantees that only a small number of these paths are selected in representing the input signal. This effectively allows the sparse representation to select the most relevant subset of modalities that best represent the given signal.

We point out some important results:

- $\mathcal{G}$ is a subset of power set of $\{1, \ldots, M\}$, *i.e.* $\mathcal{G} \subseteq \{1, \ldots, M\}$
- $\mathcal{G}$ spans the set of modalities, *i.e.* $\cup_{g \in \mathcal{G}} = \{1, \ldots, M\}$. It ensures that the corresponding penalty of $\mathcal{G}$ is a norm, because we assume that all $\boldsymbol{\alpha}_1^i, \boldsymbol{\alpha}_2^i, \ldots, \boldsymbol{\alpha}_M^i$ belong to at least one group $g \in \mathcal{G}$ (all norms are convex).
- The solution to the corresponding regularization (shown as Υ, defined later), is sparse at the group level, in the sense that decomposition coefficients within a group are usually zero or nonzero together.
- Internal nodes near the leaves of the tree correspond to modalities that we expect highly related while the internal nodes near the root represent weakly-correlated sparse codes in its subtree. For instance, the regularization for the Fig. (3.4) is defined as

$$\begin{aligned}
\Upsilon(\boldsymbol{a}_{r\rightarrow}) = & \sum_{g \in \mathcal{G}} \|\boldsymbol{a}^{(g)}\|_{\ell_2} \\
= & (|\boldsymbol{a}_{LP}| + |\boldsymbol{a}_{RP}| + |\boldsymbol{a}_{nose}| + |\boldsymbol{a}_{mouth}| + |\boldsymbol{a}_{face}|) + && \text{(3.5a)} \\
& \sqrt{(\boldsymbol{a}_{LP}^2 + \boldsymbol{a}_{RP}^2)} + && \text{(3.5b)} \\
& \sqrt{(\boldsymbol{a}_{LP}^2 + \boldsymbol{a}_{RP}^2 + \boldsymbol{a}_{nose}^2 + \boldsymbol{a}_{mouth}^2 + \boldsymbol{a}_{face}^2)} && \text{(3.5c)}
\end{aligned}$$

  where Eq. (3.5a) is equal to the ℓ_1-norm of the leaves, Eq. (3.5b) is the squared of the ℓ_2-norm of the g_2 (grouping between left and right eyes), and Eq. (3.5c) represents the squared of the ℓ_2-norm of the root g_1. Now, it is clear that each path from leaves to root has the potential to be the solution of the regularization. If face is not occluded, it can be perfectly reconstructed using only face modality; hence, only one group out of 7 nodes are selected. However, if eyes are covered, it tries to reconstruct the image using nose, mouth and the face. If $\boldsymbol{a}^{(g_2)} = \mathbf{0}$, both eyes should have zero coefficients *i.e.* $\boldsymbol{a}^{(g_3)} = 0$ and $\boldsymbol{a}^{(g_4)} = 0$.

When $\mathcal{G} \subseteq \{1, \ldots, M\}$ is a set of indices with cardinality $|\mathcal{G}|$, the collection of M-dimensional vectors that are indexed by members of $\mathcal{G}$ is defined by the $|\mathcal{G}|$-tuple $(\boldsymbol{\psi}^{(g)})_{g \in \mathcal{G}}$.

In other words, the vector $\boldsymbol{\psi}^{(g)} \in \mathbf{R}^M = [\psi_1^{(g)}, \ldots, \psi_M^{(g)}]^\top$ contains the entries of $\boldsymbol{A}_{r\rightarrow}$ corresponding to the indices in $g \in \mathcal{G}$

$$\begin{cases} \psi_m^{(g)} = \boldsymbol{A}_{rm} & \text{if } m \in g \\ \psi_m^{(g)} = 0 & \text{if } m \notin g \end{cases} \tag{3.6}$$

The tree structure $\mathcal{G}$ is embedding the latent variables, $\boldsymbol{\alpha}_{1,c}^i, \ldots, \boldsymbol{\alpha}_{M,c}^i$, in a tree-shaped hierarchy. Let us denote for each node g in tree $\mathcal{G}$, parents$(g) \subseteq \{1, \ldots, |\mathcal{G}|\}$ as the set of nodes in the tree that node g is one of their descendent and children$(g) \subseteq \{1, \ldots, |\mathcal{G}|\}$ as the set of nodes that node g is their ancestor. The tree structure $\mathcal{G}$ enforces each row r in $\{1, \ldots, p\}$ of the multimodal sparse codes $\boldsymbol{A}_{r\rightarrow}^i \in \mathbf{R}^M$ to have the following character

$$\boldsymbol{A}_{rm} \neq 0 \Rightarrow [\boldsymbol{\psi}^{(g)} \neq \mathbf{0}, \forall g \in \text{parents}(m)] \tag{3.7}$$

Intuitively, the multimodal signal $\{\boldsymbol{x}_{c,m}^i\}_{m=1}^M$ can use the r-th atom in m-th modality, $\boldsymbol{d}_m^r$, in decomposition, only if the parents of the m-th modality (parents(m)) in the tree $\mathcal{G}$ are themselves part of the decomposition. In other words, if modality m has non-zero value, *i.e.*, $\boldsymbol{A}_{rm} \neq 0$, then, all groups g that modality m is a member, are activated. Similarly, for a node g in $\mathcal{G}$, if modalities that are members of g, have no contribution in reconstruction of the signal from corresponding modalities, *e.g.*, $\{\boldsymbol{A}_{rm}^i = 0 | m \in g\} \triangleq \{\boldsymbol{\psi}^{(g)} = \mathbf{0} | m \in g\}$, then, the nodes that belong to the children(g) should not be used either.

$$\boldsymbol{\psi}^{(g)} = \mathbf{0} \Rightarrow [\boldsymbol{\psi}^{(\tilde{g})} = \mathbf{0},\ \forall \tilde{g} \in \text{children}(g)] \tag{3.8}$$

which is equal to enforce penalty on the number of groups g that contribute in the reconstruction of the multimodal input signal $\{\boldsymbol{x}_{m,c}^i\}_{m=1}^M$. From the Eq. (3.7) and Eq. (3.8), it is clear to see that, the tree structure $\mathcal{G}$ leads to hierarchical and multimodal latent sparse codes in each row of $\boldsymbol{A}_{r\rightarrow}^i$.

To be more concrete, the tree-structured groups are defined as [72]

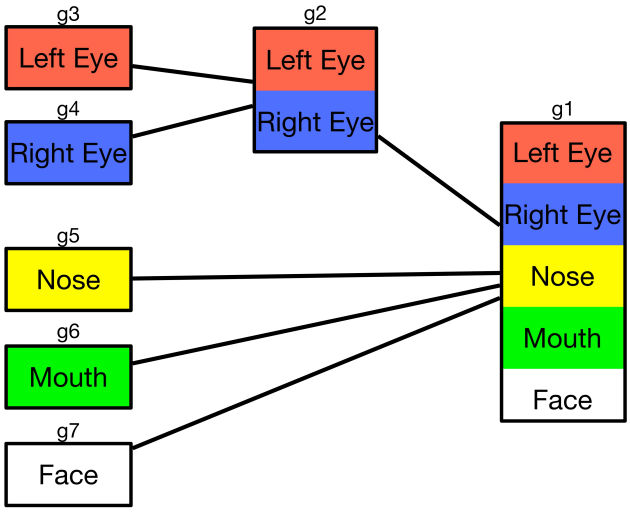


Figure 3.5: Illustration of intersection closed coupling. The tree-structure $\mathcal{G}$ corresponding to the four weak modalities of left eye, right eye, nose, and mouth, and a strong modality face, $\mathcal{G} = \{g_1, g_2, g_3, g_4, g_5, g_6, g_7\}$. If variable left eye is non-zero, *i.e.* $\boldsymbol{a}^{(g_3)} \neq 0$,then automatically, g_2 is non-zero, and also the root g_1 is selected. This path from $g_3 \rightarrow g_2 \rightarrow g_1$ is a valid solution with three activated groups out of total 7 groups. This solution gets high punishment from reconstruction part of the optimization problem. Intuitively, each interior node that is activated, here g_2, favors to see all of its members, (g_3 and g_4) to be non-zero, to get less reconstruction punishment.

Definition 3.1. ***(Tree-structured set of groups.)***

A set of groups $\mathcal{G} = \{g\}_{g \in \mathcal{G}}$ is said to be tree-structured in $\{1, \ldots, p\}$, if $\cup_{g \in \mathcal{G}} g = \{1, \ldots, p\}$ and for all $\gamma_1, \gamma_2 \in \mathcal{G}, (\gamma_1 \cap \gamma_2 \neq \emptyset) \Rightarrow (\gamma_1 \subseteq \gamma_2 \text{ or } \gamma_2 \subseteq \gamma_1)$. Then, a total order relation $\preceq$ exists between the group members such that: $\gamma_1 \preceq \gamma_2 \Rightarrow \{\gamma_1 \subseteq \gamma_2 \text{ or } \gamma_1 \cap \gamma_2 = \emptyset\}$.

The key note here is that partially correlated coupling is not allowed in the tree structure. The groups of variables either are independent or one is subset of the other.

Intersection Closed Coupling. The tree structure sparsity norm defined as (3.1) is a particular generalization of ℓ_{12} to include intersection closed coupling. setting a group to zero, makes all of its variables zero, no matter if those variables belong to other groups. In other words, this leads to groups that are not entirely selected. For instance in the case (f) of Fig. (3.3), the blue modality is shrunken to zero, *i.e.*, $\text{supp}(\boldsymbol{A}^{(g_3)}) = \mathbf{0}$. However the green modality has support in the solution, $\text{supp}(\boldsymbol{A}^{(g_4)}) \neq \mathbf{0}$. Hence, the group g_2 is not entirely selected. Many studies make a common mistake about the Lasso by considering Lasso as a

method to select variables, and consequently, group Lasso as selecting groups of variables. Concretely, the impact of norm-based grouping regularization is to set a subset of groups of variables, to zero (not to select them). Fusion in physical space is translated as grouping between different modalities of each atom. In the case that all modalities of an atom are members of a single group, then there is no overlapping between various groups. In other words, this is the same as set partitioning problem, and hence if the corresponding sparse codes of one multimodal atom is zero, the other rows in the space of sparse codes leave to be nonzero. When groups In the case of non-overlapping groups of variables, assigning zero to all members of a group can give the impression of not selecting the group. However, if the groups are overlapped, setting a group to zero, no matter if they belong to other groups. In other words, this leads to groups that are not entirely selected. This is illustrated in Fig. (3.5).

In order to demonstrate the main issue of joint sparsity and elaborate our proposed solution, we choose visual tracking task in computer vision.

3.3 Application: Visual Tracking

Visual object tracking is essential to many applications such as surveillance and robotics. Target appearance modeling is a key component in visual tracking and is challenging due to the real-world problems for instance, illumination change, scaling and pose variations, background clutter and occlusions. Various types of features have been exploited to make an accurate representation, *e.g.* color histograms [137, 29] and keypoint-based features like sift or hog [59, 138]. However, no single feature can be robust to all possible scenarios in a video sequence. It is important to construct a new and powerful appearance model which can integrate useful features and explore their mutual dependencies.

Using different feature modalities has been demonstrated to be effective for visual tracking. Many recent trackers attempt to model the appearance using various features such as color, texture or edge [95, 187, 94, 62]. Typically, information fusion in visual tracking happens at either the feature or classifier level [153]. In feature fusion different types of

features are combined to make a new feature set, while classifier fusion aggregates decisions from several classifiers which are individually trained on different features. Classifier fusion has been well-studied in visual tracking, for instance as the online multiple instance learning in [11] and as the multiple kernel boosting in [191]. Although fusion at the feature level has been demonstrated to be more effective for visual tracking, it is a less-studied problem. This is mainly due to the incompatibility of feature sets [95]. For example in [69] feature fusion using color, gradient, and texture was proposed; but the method requires all features to have the same dimension. The feature fusion in [187] concatenates all the features into one vector. The dimension of this vector can be very high relative to the number of training samples available, resulting in the classic "curse-of-dimensionality" problem. Moreover, concatenation of the feature vectors makes it impossible to model any potential correlations between the feature types. We refer to the proposed tracking scheme as multimodal tracking by tree-structured hierarchical modeling (MM-THM). The contribution is thus three-fold.

- First, MM-THM encodes the hierarchical correlation between different modality channels into a tree structure and scores them adaptively according to their representative and discriminative powers.

- Despite existing joint sparse representation based tracking algorithms that make dictionaries without training, MM-THM learns a dictionary for each feature while the tree-structure regularization is enforced in the space of sparse codes to implicitly enforce the physical connection between dictionaries.

- The performance of each feature modality directly affects its contribution to the decisions in upcoming frames. When a feature is unreliable, a larger weight (punishment) would be assigned to its decomposition in the latent space of sparse codes, which promotes the optimization to make them zero.

We test our hierarchical appearance tracker on recent online tracking benchmark data sets which evaluate the proposed method for various real-world challenges involving significant

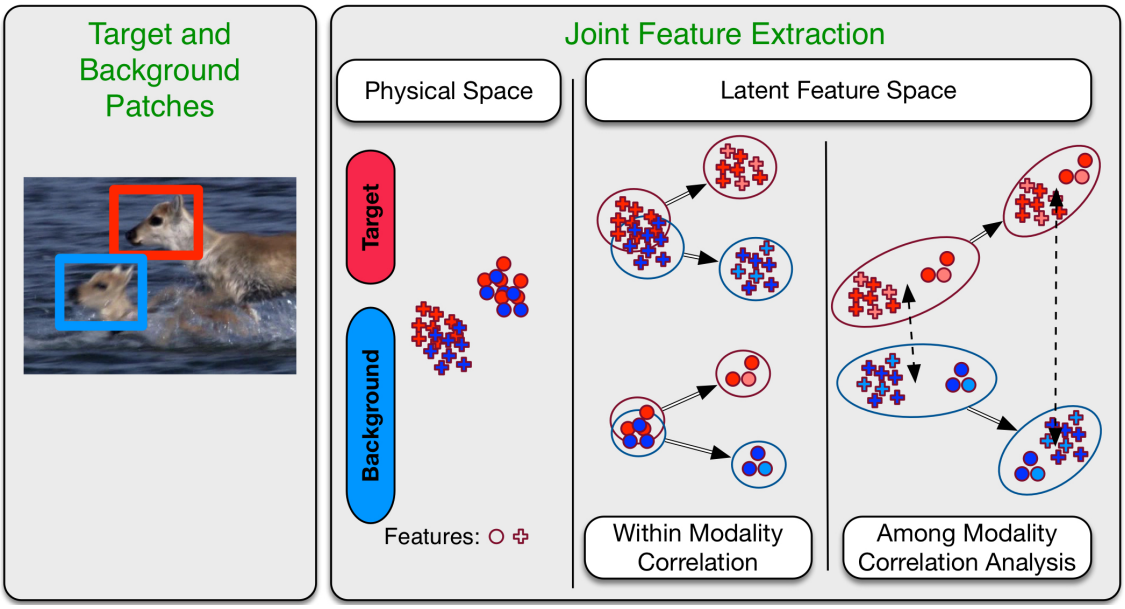


Figure 3.6: Illustration of the proposed MM-THM framework. Tracking can be seen as a binary classification of target and background. Consider each patch has M different modalities. Originally, the physical attributes are not discriminative enough to distinguish the target from the background. Our method learns a set of dictionaries to find the representation of data in latent space of sparse codes, to make the target more distinctive in each modality, and, from multimodal stand-point.

changes in appearance and pose, background clutter, and occlusions. The framework of the proposed MM-THM is illustrated in Fig. 3.6.

3.4 Related Works

Target appearance modeling is one of the key components in any visual tracking algorithms. Since the early work of Lucas and Kanade (LK) [102], holistic templates based on raw intensity values have been widely used for tracking [4]. Visual appearance modeling can be broadly categorized into generative [23] and discriminative [86] algorithms.

Generative models reformulate tracking as a search for an optimal state that yields an object appearance most similar to the target appearance model. The well-known methods in generative models are mostly either pixel-based like Gaussian mixture models [75] and color histograms [29], or global-based like subspace learning [152].

Discriminative models attempt to transfer visual information from the physical space to a latent feature space so that a simple classifier can separate target from background [5, 28].

Generative or discriminative appearance models have their own merits and limitations. The proposed MM-THM can be considered as a unified framework for robust visual tracking that has advantages of both generative and discriminative schemes in the particular signal representation of sparse coding. In MM-THM, we design a joint decision measure based on both reconstruction and classification errors, and hence the method is both generative and discriminative. In the following, we briefly go over the evolution of sparsity trackers.

3.4.1 Sparse Trackers

Sparsity-based trackers represent the target candidates as linear combinations of a set of bases or dictionary elements or atoms. The dictionary is made from trivial target templates which are selected from the tracking results in previous frames. The target and background dictionaries in each feature are obtained by horizontally concatenating the samples of each class. Given object and background dictionaries as $\boldsymbol{D}_O$ and $\boldsymbol{D}_B$, respectively, the final dictionary is obtained by putting together the target and background sub-dictionaries, $\boldsymbol{D} = [\boldsymbol{D}_O, \boldsymbol{D}_B]$ with p columns.

Inspired by sparse representation in face recognition [185], it is assumed that the test candidate, $\boldsymbol{x}^t$, is the object in this frame, if it can be represented using atoms that belong to the object dictionary $\boldsymbol{D}_O$. That is, the test candidate $\boldsymbol{x}^t$ lies in the space formed by the target template of and can be approximated using few number of atoms in object dictionary: $\boldsymbol{x}^t \approx \boldsymbol{D}\boldsymbol{\alpha}^t$, where $\boldsymbol{\alpha}^t \in \mathbf{R}^p$ is sparse codes of the test patch. The test patch is the target in current frame if $\boldsymbol{\alpha}^t$ uses $\boldsymbol{D}_O$ to reconstruct the patch and is zero otherwise: $\boldsymbol{\alpha}^t = [\boldsymbol{\alpha}_O^\top, \mathbf{0}^\top]^\top$.

The majority of sparse trackers obtain the target appearance model by optimizing an objective function that incorporates reconstruction error and sparsity regularization norm. Given dictionary as $\boldsymbol{D} = [\boldsymbol{d}^1, \ldots, \boldsymbol{d}^p]$, each target candidate is decomposed to its

corresponding coefficient vector, $\boldsymbol{\alpha}^t$

$$\underset{\boldsymbol{\alpha}^t \in \mathbf{R}^p}{\operatorname{argmin}} \frac{1}{2}\|\boldsymbol{x}^t - \boldsymbol{D}\boldsymbol{\alpha}^t\|_2^2 + \lambda\Omega(\boldsymbol{\alpha}^t) \tag{3.9}$$

where λ is the regularization parameter. Traditionally, Ω is chosen as ℓ_1-norm like [213, 210].

Sparsity Issues. It is evident from Eq. (3.9) that the performance of the sparsity-based trackers strongly depends on the dictionary to span over the recent target appearance. Majority of the methods update the dictionary by simply stacking the target templates [62, 213, 210] without "training"; which inevitably incorporates background and noise in the dictionary and may not be optimum to deal with changes in appearance and pose. The sparse coding objective function in Eq. (3.9) only cares about reconstructing the input well, and does not attempt to make sparse codes useful as input for particular task of visual tracking. We explain our contribution to solve this issue in Section 3.5 by modifying dictionary through iterative optimization to make sparse codes more useful for prediction.

On the other hand, the primary goal of standard ℓ_1-norm sparse coding is cardinality: to penalize dense vectors of parameters with a large number of nonzero coefficients. Particularly, these regularizations treat each variable individually, and they are blind to potential relationships that may exist between the features, which leads us to the design of sparsity-inducing norms capable of encoding some additional structure about the variables. Next, we briefly go over joint sparsity trackers that are proposed in an attempt to model potential relationships between various features of the target, in the space of sparse codes.

3.4.2 Joint Sparsity Trackers

In particle filter-based tracking methods, the joint sparse appearance model has been used based on the following intuition [69, 207, 208]: since particles are sampled at and around the previous location of the target, each particle shares dependencies with other particles and their corresponding images are likely to be similar. For instance in [208, 209], learning the representation of each particle is viewed as an individual task and a multi-task learning with

joint sparsity for all particles is employed. In [207], sparse representations of all particles are jointly learned by applying the low-rank sparse learning. In [62], visual tracking is formulated as the multi-task multi-view joint sparse representation where each feature is referred to as a view.

The joint decision between different modalities is made using joint sparse representation classification (JSRC) by enforcing sparse codes of the test patch from M different modalities $\boldsymbol{X}^t = \{\boldsymbol{x}_1^t, \boldsymbol{x}_2^t, \ldots, \boldsymbol{x}_M^t\}$ to have the same sparsity pattern. It means that each feature modality of the patch is reconstructed using the same set of column indices from its corresponding dictionary ($\{\boldsymbol{D}_1, \boldsymbol{D}_2, \ldots, \boldsymbol{D}_M\}$). The test sample is classified as target or background by its multimodal sparse codes $\boldsymbol{A}^t = [\boldsymbol{\alpha}_1^t, \ldots, \boldsymbol{\alpha}_M^t]$ obtained by dictionaries $\{\boldsymbol{D}_m\}_{m=1}^M$. This is formulated as multimodal joint sparse modeling and the dictionary is made from concatenation of training samples:

$$\underset{\boldsymbol{A}^t}{\operatorname{argmin}} \frac{1}{2} \sum_{m=1}^{M} \|\boldsymbol{x}_m^t - \boldsymbol{D}_m \boldsymbol{\alpha}_m^t\|_2^2 + \lambda \Omega(\boldsymbol{A}^t) \tag{3.10}$$

where the first component is the reconstruction error and $\Omega(\boldsymbol{A}^t) = \|\boldsymbol{A}^t\|_{\ell_{12}}$ is the regularization error. Collaboration between $\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_M$ is imposed by ℓ_{12} and is defined as $\|\boldsymbol{A}\|_{\ell_{12}} = \sum_{r=1}^{p} \|\boldsymbol{A}_{r\rightarrow}\|_2$; where $\boldsymbol{A}_{r\rightarrow} = [\boldsymbol{A}_{r1}, \boldsymbol{A}_{r2}, \ldots, \boldsymbol{A}_{rM}]$ is the r-th row of $\boldsymbol{A}$. The ℓ_{12} promotes solution with sparse non-zero rows; hence, similar support is enforced on $\boldsymbol{A}$ at the column level of each dictionary $\boldsymbol{D}_m$.

Other Multimodal Fusion Approaches

Bayesian inference fusion methods have been the most popular mechanisms for data fusion where the multimodal information is combined as per the rules of probability theory [100, 212]. These methods have various advantages, including increment computation of the posterior probability based on new observations, convenient incorporation of any prior knowledge, and allowing a subjective probability estimate for the *a priori* of hypotheses in the absence of empirical data. However, these advantages are also seen as the limitations

in some cases. Bayesian inference methods require the prior and conditional probabilities of the hypothesis to be well defined. In absence of a proper prior, these methods might not perform well. Many nonparametric Bayesian approaches have been proposed to tackle this problem [99, 113, 214]. However, the high computational cost of Gibbs sampling and variational learning hinders its capability for online learning.

The most recent deep learning-based fusion framework uses the deep neural network to automatically generate a set of hierarchical representations of the multimodal inputs [46, 132, 160, 163, 27]. The cross-correlation among different modalities is implicitly incorporated in the derived feature vector instead of the explicit configuration using a tree structure as in MM-THM. Another distinctive difference is that most deep learning-based multimodal fusion relies on off-line trained models obtained from very large training datasets that would incur high computational cost. In use cases with limited time-to-solution window, this might not be optimal or feasible for applications where the operational environment is of very dynamic nature that demands models to be updated in an online fashion or where the cost of acquiring large amount of training samples is extremely high if possible at all.

The proposed MM-THM solution differs from these fusion alternatives for its strong capacity in both online training (or learning) and online testing (or classification), which is particularly important for in-situ applications. MM-THM is also more computationally efficient for real-world deployment.

3.5 The Proposed Visual Tracker - MM-THM

In this section, we elaborate on our particle filtering-based tracking method that uses the tree structure-based hierarchical appearance modeling. Our key insight to robustly model the target appearance is that the patterns of non-zero coefficients in $\boldsymbol{A}$ conveys the feature-modality configuration. We encode this information in the regularization with norms that support structural sparsity, to encourage the solutions of sparse regularized problems to promote the desired patterns of non-zero coefficients.

Assuming multimodal sample i as $\{\boldsymbol{x}_m^i\}_{m=1}^M$, we propose to obtain its sparse representation matrix $\boldsymbol{A}^i = [\boldsymbol{\alpha}_1^i, \ldots, \boldsymbol{\alpha}_M^i]$ and the set of dictionaries $\boldsymbol{D}_m = [\boldsymbol{d}_m^1, \ldots, \boldsymbol{d}_m^p]$ and m in $\{1, \ldots, M\}$

$$\underset{\substack{\boldsymbol{A}^i \\ \{\boldsymbol{D}_m\}}}{\operatorname{argmin}} \sum_{m=1}^{M} \frac{1}{2}\|\boldsymbol{x}_m^i - \boldsymbol{D}_m\boldsymbol{\alpha}_m^i\|_2^2 + \lambda_1\Omega(\boldsymbol{A}^i) + \frac{\lambda_2}{2}\|\boldsymbol{A}^i\|_F^2 \tag{3.11}$$

where Ω is the tree-structure norm regularization defined in Eq. (3.4) and λ_1 and λ_2 are the regular parameters. With only one feature ($m = 1$) and $\lambda_2 > 0$, the optimization problem (3.11) reduces to elastic-net formula [217] (when $\lambda_2 = 0$, elastic-net would be the same as Lasso). Mairal et al [104] proved that elastic-net is strongly convex and leads to more stable sparse code solution than Lasso. In our experiments λ_2 is assigned a small positive value as $\lambda_2 = 10^{-3}\lambda_1$.

The optimization problem (3.11) includes two terms: a data-fidelity term which is convex with Lipschitz-continuous gradient; and, a non-smooth norm-based regularization that models the high-order prior information of coupling between feature-modalities. The regularization penalizes the number of overlapping groups that are "involved" in the decomposition, while data-fidelity term prone to reconstruct the multimodal signal with all groups selected. The regularization parameter $\lambda_1 \geq 0$ is used to adjust the tradeoff between minimizing the loss and finding a solution which is sparse at the group level.

3.5.1 Optimization

The problem (3.11) has the product of the two optimization variables as $\boldsymbol{D}\boldsymbol{\alpha}_m^i$; which implies that this problem is not joint convex in the space of multimodal coefficients and the dictionary. However, when one of the two optimization variable is fixed, the problem (3.11) is convex with respect to the other variable [108]. Hence, the problem (3.11) is solved by splitting to two sub-problems: 1. given dictionaries $\{\boldsymbol{D}_m\}_{m=1}^M$, estimate the multimodal sparse codes $\{\boldsymbol{\alpha}_m^i\}_{m=1}^M$ for all i in $\{1, \ldots, N\}$; 2. given sparse representation of samples in m-th modality, $\{\boldsymbol{\alpha}_i^m\}_{i=1}^N$, update the corresponding dictionary of m-th modality $\boldsymbol{D}_m$.

We initialize the multimodal dictionaries $\{\boldsymbol{D}_m\}_{m=1}^M$ from target and background templates similar to [95, 208, 62]. We use the alternating direction method of multipliers (ADMM) [136] to obtain optimal multimodal sparse codes $\boldsymbol{A}^{i*} \in \mathbf{R}^{p\times M}$ of i-th multimodal sample $\{\boldsymbol{x}_m^i\}_{m=1}^M$ and for all $i \in \{1,\ldots,N\}$, by solving the optimization problem (3.11).

Assume temporary variables $\boldsymbol{Z} \in \mathbf{R}^{p\times M} = [\boldsymbol{z}_1,\ldots,\boldsymbol{z}_M]$ and $\boldsymbol{U} \in \mathbf{R}^{p\times M} = [\boldsymbol{u}_1,\ldots,\boldsymbol{u}_M]$ and both initialized as zero. We denote the proximal operator associated with the norm Ω as $\mathbf{prox}_{\lambda\Omega}$ that maps its domain, vector $\mathbf{p}$, to the vector $\mathbf{q}$, both with size M as

$$\mathbf{prox}_{\lambda\Omega}(\mathbf{p}) \triangleq \underset{\mathbf{q}}{\operatorname{argmin}} \frac{1}{2}\|\mathbf{p}-\mathbf{q}\|_2^2 + \lambda\Omega(\mathbf{q}). \tag{3.12}$$

Then, the solution to the optimization 3.11 in iteration τ would be [136]:

$$\boldsymbol{A}^{(\tau+1)} = \mathbf{prox}_{\lambda_1 f}(\boldsymbol{Z}^{(\tau)} - \boldsymbol{U}^{(\tau)}) \tag{3.13a}$$

$$\boldsymbol{Z}^{(\tau+1)} = \mathbf{prox}_{\lambda_1\Omega}(\boldsymbol{A}^{(\tau+1)} + \boldsymbol{U}^{(\tau)}) \tag{3.13b}$$

$$\boldsymbol{U}^{(\tau+1)} = \boldsymbol{U}^{(\tau)} + \boldsymbol{A}^{(\tau+1)} - \boldsymbol{Z}^{(\tau+1)} \tag{3.13c}$$

where data-fidelity term $f \triangleq \sum_{m=1}^M \frac{1}{2}\|\boldsymbol{x}_m^i - \boldsymbol{D}_m\boldsymbol{\alpha}_m^i\|_2^2 + \frac{\lambda_2}{2}\|\boldsymbol{\alpha}_m^i\|_2$ is smooth and differentiable. The optimization variables $\boldsymbol{A}^{(\tau)}$ and $\boldsymbol{Z}^{(\tau)}$ are the solution of minimizing the smooth and non-smooth part of the problem (3.11) at iteration τ, respectively. After a limited number of iterations they will eventually converge, (*i.e.* $\boldsymbol{U}^{(\tau+1)} = \boldsymbol{U}^{(\tau)}$))

The proximal step of problem (3.13a) is defined for each modality independently as:

$$\begin{aligned}&\mathbf{prox}_{\lambda_1 f}(\boldsymbol{z}_m^{(\tau)} - \boldsymbol{u}_m^{(\tau)}) = \\ &\underset{\boldsymbol{\alpha}_m}{\operatorname{argmin}}\ \lambda_1 f(\boldsymbol{\alpha}_m^{(\tau)}) + \frac{1}{2}\|\boldsymbol{\alpha}_m^{(\tau)} - (\boldsymbol{z}_m^{(\tau)} - \boldsymbol{u}_m^{(\tau)})\|_2^2\end{aligned} \tag{3.14}$$

f is smooth with gradient $\nabla_{\boldsymbol{\alpha}_m} f = -\boldsymbol{D}_m^\top(\boldsymbol{x}_m - \boldsymbol{D}_m\boldsymbol{\alpha}_m) + \lambda_2\boldsymbol{\alpha}_m$, we compute the solution to problem (3.14) in iteration $\tau+1$:

$$\boldsymbol{\alpha}_m^{(\tau+1)} = \Delta^{-1}(\boldsymbol{D}_m^\top\boldsymbol{x}_m + \frac{1}{\lambda_1}(\boldsymbol{z}_m^{(\tau)} - \boldsymbol{u}_m^{(\tau)})) \tag{3.15}$$

where $\Delta = (\boldsymbol{D}_m^\top \boldsymbol{D}_m + (\frac{1}{\lambda_1} + \lambda_2)\boldsymbol{I})$. the method is designed to get high classification accuracy while $\{\boldsymbol{D}_m\}_{m=1}^M$ have small numbers of atoms; but, this may increase the chance of singularity in (3.15). However, $\lambda_1 > 0$ and $\lambda_2 > 0$ makes the denominator Δ positive definite. We solve (3.15) for each modality separately and concatenate the results to make $\boldsymbol{A}^{(\tau+1)} = [\boldsymbol{\alpha}_1^{(\tau+1)}, \ldots, \boldsymbol{\alpha}_M^{(\tau+1)}]$.

Next, we solve the proximal step over $\boldsymbol{Z}_{r\rightarrow}$ in (3.13b) for each row r of $\boldsymbol{A}$ and r in $\{1, \ldots, p\}$:

$$\begin{aligned} &\mathbf{prox}_{\lambda_1\Omega}(\boldsymbol{A}_{r\rightarrow}^{(\tau+1)} + \boldsymbol{U}_{r\rightarrow}^{(\tau)}) = \\ &\underset{\boldsymbol{Z}_{r\rightarrow}}{\operatorname{argmin}}\ \lambda_1\Omega(\boldsymbol{Z}_{r\rightarrow}^{(\tau+1)}) + \frac{1}{2}\|\boldsymbol{Z}_{r\rightarrow}^{(\tau+1)} - (\boldsymbol{A}_{r\rightarrow}^{(\tau+1)} + \boldsymbol{U}_{r\rightarrow}^{(\tau)})\|_2^2 \end{aligned} \tag{3.16}$$

Substituting Ω from Eq. (3.4) in (3.16), the final optimization problem for $\boldsymbol{Z}$ can be written as:

$$\underset{\boldsymbol{Z}}{\operatorname{argmin}} \sum_{r=1}^{p} \Big(\lambda_1 \sum_{g\in\mathcal{G}} \Big(\sum_{m\in g} (\omega_m^{(g)})^2 |\boldsymbol{A}_{rm}|^2\Big)^{\frac{1}{2}} + \frac{1}{2}\|\boldsymbol{Z}_{r\rightarrow}^{(\tau+1)} - (\boldsymbol{A}_{r\rightarrow}^{(\tau+1)} + \boldsymbol{U}_{r\rightarrow}^{(\tau)})\|_2^2\Big) \tag{3.17}$$

The optimization problem (3.17) is solved in p independent optimizations corresponding to p rows, while each optimization is done on an M-dimensional vector, $\boldsymbol{Z}_{r\rightarrow}^\top$. Since the groups are ordered, each of the p optimization can be done in one iteration using the dual form [72], which means that proximal step (3.17) can be solved with the same computational cost as joint sparsity. By extension of optimization algorithm in [72], we solve the proximal step of (3.17) for optimization variable $\boldsymbol{Z}_{r\rightarrow}^\top$ in Algorithm (2). We solve the optimization problem (3.17) using the SPArse Modeling Software [72]. After $\boldsymbol{Z}$ is obtained, this iteration would be finished by updating $\boldsymbol{U}$ according to (3.13c).

3.5.2 Multimodal Dictionary Learning

So far we obtain multimodal sparse coefficients of i-th particle, $\boldsymbol{A}^{i*} = [\boldsymbol{\alpha}_1^{i*}, \ldots, \boldsymbol{\alpha}_M^{i*}]$ by solving the optimization problem (3.11) given the set of dictionaries $\{\boldsymbol{D}_m\}_{m=1}^M$. In this

Algorithm 2 Algorithm to solve the proximal optimization step (Eq. (3.17))

Input: $\boldsymbol{V} = [\boldsymbol{V}_{1\rightarrow}^\top, \ldots, \boldsymbol{V}_{p\rightarrow}^\top] \in \mathbf{R}^{p\times M}$, ordered groups $\mathcal{G} = \{g_1, \ldots, g_{|\mathcal{G}|}\}$, weights $\omega^{(g)}$.

Output: for each r in $\{1, \ldots, p\}$, primal vector $\boldsymbol{Z}_{r\rightarrow}^\top \in \mathbf{R}^M$ and its dual solution $\boldsymbol{\Xi}_r = [\boldsymbol{\Xi}_r^{1\downarrow}, \ldots, \boldsymbol{\Xi}_r^{|\mathcal{G}|\downarrow}] \in \mathbf{R}^{M\times|\mathcal{G}|}$; whose i-th column $\boldsymbol{\Xi}_r^{i\downarrow}$ has size M.

for each row $r \in \{1, \ldots, p\}$ **do**
  Reset the dual solution $\boldsymbol{\Xi}_r^{1\downarrow} \leftarrow \mathbf{0}, \ldots, \boldsymbol{\Xi}_r^{|\mathcal{G}|\downarrow} \leftarrow \mathbf{0}$.
  for $g = g_1, g_2, \ldots \in \mathcal{G}$ **do**
    $\boldsymbol{Z}_{r\rightarrow}^\top = \boldsymbol{V}_{r\rightarrow}^\top - \sum_{h\leq g} \boldsymbol{\Xi}_r^{h\downarrow}$.
    $\boldsymbol{\Xi}_r^{g\downarrow} = \begin{cases} (1 - \frac{\lambda_1\omega^{(g)}}{\|\boldsymbol{Z}_{r\rightarrow}^{\top\ (g)}\|_2})\boldsymbol{Z}_{r\rightarrow}^{\top\ (g)} & \text{if } \|\boldsymbol{Z}_{r\rightarrow}^{\top\ (g)}\|_2 > \lambda_1\omega^{(g)} \\ \boldsymbol{Z}_{r\rightarrow}^{\top\ (g)} & \text{if } \|\boldsymbol{Z}_{r\rightarrow}^{\top\ (g)}\|_2 \leq \lambda_1\omega^{(g)} \end{cases}$
  end for
  $\boldsymbol{Z}_{r\rightarrow}^\top = \boldsymbol{V}_{r\rightarrow}^\top - \sum_{g\in\mathcal{G}} \boldsymbol{\Xi}_r^{g\downarrow}$.
end for

section, the obtained decomposition coefficients are exploited to update the dictionaries. The dictionary $\boldsymbol{D}_m = [\boldsymbol{d}_m^1 \ldots, \boldsymbol{d}_m^p]$ is updated using the sparse representation of all samples from m-th modality: $[\boldsymbol{\alpha}_m^1, \ldots, \boldsymbol{\alpha}_m^N]$. The appearance of the target in visual tracking is dynamic and changes over time; hence, the popular iterative batch dictionary learning methods (*e.g.* KSVD [2]) that access the whole training set in each iteration to minimize the cost function cannot be used. Instead of minimizing empirical cost $f_n(\boldsymbol{D}_m)$ with high precision, we design dictionary learning method based on minimizing a quadratic local surrogate of the expected cost as, $\hat{f}(\boldsymbol{D}_m) \triangleq \mathbb{E}_x[\mathcal{L}_u(\boldsymbol{x}_m, \boldsymbol{D}_m)]$ as in [19, 106] assuming that the data $\{\boldsymbol{x}_m\}$ is drawn from an (unknown) finite probability distribution $p(\boldsymbol{x}_m)$. The dictionary $\boldsymbol{D}_m$, will be updated by solving the optimization (3.11) using online stochastic approximations [106].

In attempt to utilize target and background labels of the training data in a discriminative tasks, the estimated coefficients using Eq. (3.11) from m-th feature, $[\boldsymbol{\alpha}_m^{1*}, \ldots, \boldsymbol{\alpha}_m^{N*}]$, are assumed as the latent feature representation for the training data $[\boldsymbol{x}_m^1, \ldots, \boldsymbol{x}_m^N]$ and a classifier is trained in a classical expected risk minimization [181] by adopting multivariate ridge regression model with quadratic loss and ℓ_2 norm regularization. The supervised loss function evaluates how close classifier with parameters $\boldsymbol{W}_m$ using $\boldsymbol{\alpha}_m^{i*}$ can predict label $\boldsymbol{y}^i$.

We update dictionary $\boldsymbol{D}_m$ to keep track of the appearance change, once in each $L = 15$ frames. We provide steps to learn the multimodal unsupervised dictionary and the joint sparse modeling using tree-structure regularization in Algorithm 3.

Algorithm 3 Online MM-THM

Input: $\{\boldsymbol{X}^i\}_{i=1}^N$ and $\boldsymbol{X}^i = \{\boldsymbol{x}_m^i\}_{m=1}^M$, T (number of iterations)

Initialize $\boldsymbol{D}_m$ with samples of m-th feature of both target and background.
Reset the past: $\boldsymbol{\Gamma}_m \leftarrow 0$ and $\boldsymbol{C}_m \leftarrow 0$ for all m in $\{1, \ldots, M\}$
for $\tau = 1$ to T **do**
    Fix $\{\boldsymbol{D}_m\}_{m=1}^M$ and estimate multimodal $\{\boldsymbol{A}^i\}_{i=1}^N$
    Compute $\boldsymbol{A}^i = [\boldsymbol{\alpha}_1^i, \ldots, \boldsymbol{\alpha}_M^i]$ for each i in $\{1, \ldots, N\}$ using the Eq. (3.11).
    for each feature $m = [1 \cdots M]$ **do**
        Update $\boldsymbol{\Gamma}_m = \boldsymbol{\Gamma}_m + \sum_i \boldsymbol{\alpha}_m^i \boldsymbol{\alpha}_m^{i\top}$ and $\boldsymbol{C}_m = \boldsymbol{C}_m + \sum_i \boldsymbol{x}_m^i \boldsymbol{\alpha}_m^{i\top}$.
    end for
    Fix $\{\boldsymbol{A}^i\}_{i=1}^N$ and update each dictionary $\{\boldsymbol{D}_m = [\boldsymbol{d}_m^1, \ldots, \boldsymbol{d}_m^p]\}_{m=1}^M$.
    for each modality $m = [1 \cdots M]$ **do**
        for each atom $\boldsymbol{d}_m^j$ and $j \in \{1, \ldots, p\}$ **do**
            if $\boldsymbol{\Gamma}_m(j, j) > 0$ **then**
                Update $\boldsymbol{d}_m^j$.
            end if
        end for
    end for
end for

3.5.3 Classification and Template Update

In a new frame t, we sample N target candidates, $\{\boldsymbol{X}^i\}_{i=1}^N$, which each one has same M modalities: $\boldsymbol{X}^i = \{\boldsymbol{x}_m^i\}_{m=1}^M$. We solve (3.11) to decompose each candidate $\boldsymbol{X}^i = \{\boldsymbol{x}_m^i\}_{m=1}^M$ to $\boldsymbol{B}^i = [\boldsymbol{b}_1^i, \ldots, \boldsymbol{b}_M^i]$, using the $\{\boldsymbol{D}_m\}_{m=1}^M$ that learned in training phase. Then, each candidate i is evaluated from both reconstructive and discriminative perspective: how successful $\boldsymbol{D}_m$ is to reconstruct the target candidate (lower reconstruction error) and how close the target label $[1, 0]^\top$ can be produced using the learned classifier $\boldsymbol{W}_m$. The distance of i-th candidate to the target from m-th modality is measured by $\mathcal{L}_m^i$:

$$\mathcal{L}_m^i = \|\boldsymbol{x}_m^i - \boldsymbol{D}_m \boldsymbol{b}_m^i\|_2^2 + \beta \|[0, 1]^\top - \boldsymbol{W}_m \boldsymbol{b}_m^i\|_2^2 \tag{3.18}$$

and β controls the contribution between reconstruction error $\|\boldsymbol{x}_m^i - \boldsymbol{D}_m \boldsymbol{b}_m^i\|_2^2$ and misclassification error $\|[0, 1]^\top - \boldsymbol{W}_m \boldsymbol{b}_m^i\|_2^2$. We set $\beta = 0.3$ for all the experiments. The candidate, $\{\boldsymbol{x}_m^*\}$ with minimum error $\mathcal{L}^*$ defined as the sum of errors from all modalities, *i.e.* $\text{argmin}_i(\sum_m \mathcal{L}_m^i)$, would be the target in this frame. The training set is augmented by the new sample. We exponentiate and normalize $\{\mathcal{L}_m\}$ using softmax function: $\mathcal{L}_m^* = \exp(\mathcal{L}_m^*) / \sum_m (\exp(\mathcal{L}_m^*))$.

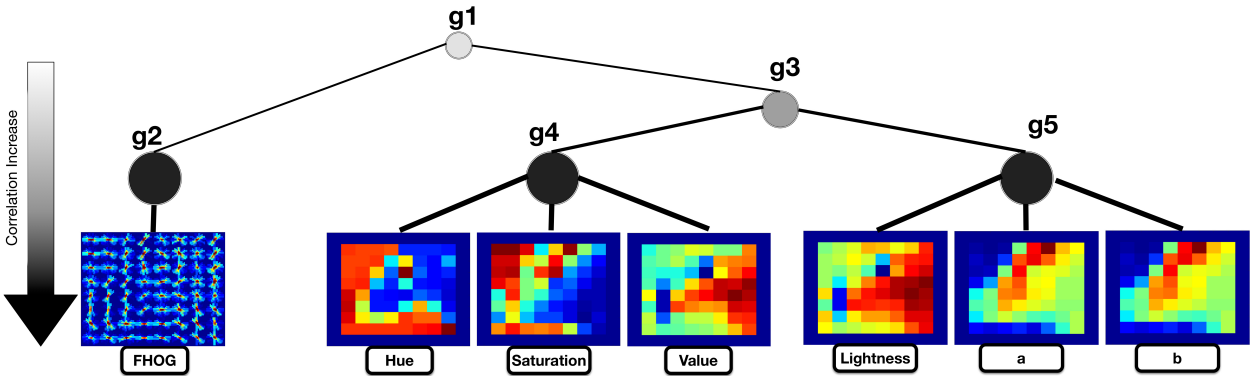


Figure 3.7: The hand-coded tree-structure norm-based regularization in space of sparse codes for MM-THM. This tree has $M = 7$ leaves (features) and 5 groups. From left to right: fhog, HSV and CIE Lab channels.

We update the group weights $\omega^{(g)}$ in the tree-structure Ω by the new sample in a moving average scheme, depends on how close the new sample is to the target: $\omega^{(g)} = \varrho\omega^{(g)} + (1 - \varrho)\sum_{m\in g}\mathcal{L}_m^* \quad \forall g \in \mathcal{G}$ with $\varrho = 0.8$ to control the length scale of the moving average.

Let us illustrate this with an example: Assume $\tilde{m}$ as a subset of M feature modalities $\tilde{m} \subseteq \{1, \ldots, M\}$ that are unreliable and produce high error in this frame based on Eq. (3.18). Also, consider $\tilde{g}$ as those groups in the tree structure $\mathcal{G}$ that have unreliable features $\tilde{m}$ as their members. Increasing the error measurement, would result in bigger weights for the $\tilde{g}$. The tree-structure regularization in (3.4) assigns the elements of the sparse codes that corresponds to the unreliable features $\tilde{m}$ to zero. This remove the unreliable features from making decision in upcoming frame. The idea of adaptive feature fusion was investigated in [156] where the quality of each modality is obtained by the sparsity concentration index. However, there is no clear relation between sparsity degree and reliability of the data.

We update $\{\boldsymbol{D}_m\}$ and $\{\boldsymbol{W}_m\}$ periodically. The set of samples to update these parameters is made by random sampling bounding boxes around the optimal location, $\{\boldsymbol{x}_m^*\}$ as positive samples, and far away from the optimal location as negative samples. We only consider those samples with small loss function $\mathcal{L}^*$. In fact, each training sample, $\boldsymbol{x}_m^*$ is weighted by $\exp(-\mathcal{L}_m^*)$.

3.6 Experiments and Results

The proposed method does feature fusion by tree-structured joint sparse modeling and decision level fusion by aggregating votes from modality-based classifiers. In this section we evaluate the proposed MM-THM.

Implementation Details The proposed tracker is implemented in MATLAB and MEX. The method does not need to update dictionary and classifier in each frame and it takes 6fps on average on Macbook Pro with 2.3 GHz processor and 16 GHz memory. The patches are preprocessed to have zero mean and unit ℓ_2 norm. In learning the multimodal dictionaries of Algorithm(3), the total number of iterations is $T = 3$. We choose λ_1 as 0.1. The regularization parameter of the Frobenius norm is chosen as $\lambda_2 = 0.01\lambda_1$ as suggested in [104].

We represent each observation using 7 features: fhog [41], hue, saturation and value channels of HSV and lightness, a and b of Lab system. The HSV and Lab features are "similar by nature". The tree-structure model of Fig.(3.7) defines 5 groups from 7 features as $\mathcal{G} = \{g_1, \ldots, g_5\}$, whose g_1 is the root, g_2 applies sparsity on fhog, g_4 models the grouping between HSV channels, g_5 models the grouping between Lab channels and g_3 models the grouping between g_4 and g_5. The grouping is enforced in space of sparse codes. This tree-structure enforces grouping at multiple granularities (levels): the g_2, g_4 and g_5 are the internal nodes near the bottom of the tree that correspond to highly correlated sparse representations, whereas the internal nodes near the root *i.e.* g_3 enforces grouping with weak correlations among the sparse codes in its subtree. The group weights are initialized one.

The template set includes boxes of size 32×32. For each feature, we experimentally set $\boldsymbol{D}_m$ to have 40 atoms, with 20 for the target class and 20 for the background. Also, we take 200 positive and 200 negative samples from location of the target in the first frame and after that 400 samples are extracted as test set in each frame. For all experiments, particle sampling is done assuming variances of affine parameters as $(0.01, 0.0005, 0.0005, 0.01, 4.0, 4.0)$.

We evaluate the proposed tracking method on the OTB-50 [188] and OTB-100 [189] that include 50 and 100 fully annotated sequences in One-Pass Evaluation (OPE) mode.

Table 3.1: The average overlap score of 5 trackers on 7 different videos. The best is shown by red and blue is the second best.

Video	CN	MUSTer	KCF	MEEM	MTMV	p1	p2	p3	p4
trellis	0.49	0.78	0.47	0.62	0.61	0.79	0.81	0.78	0.86
basketball	0.64	0.75	0.48	0.82	0.76	0.72	0.74	0.69	0.78
bolt	0.78	0.78	0.01	0.97	0.01	0.75	0.69	0.62	0.77
singer1	0.36	0.75	0.48	0.29	0.42	0.75	0.68	0.68	0.81
singer2	0.05	0.76	0.74	0.04	0.71	0.05	0.78	0.78	0.84
skating	0.49	0.50	0.13	0.40	0.51	0.58	0.56	0.36	0.69
couple	0.41	0.63	0.53	0.60	0.46	0.65	0.74	0.71	0.76

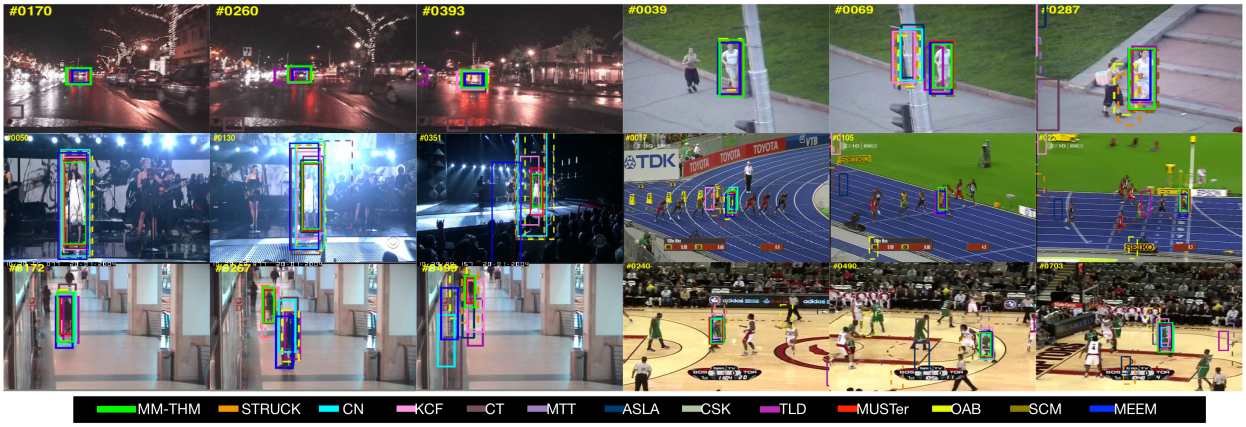
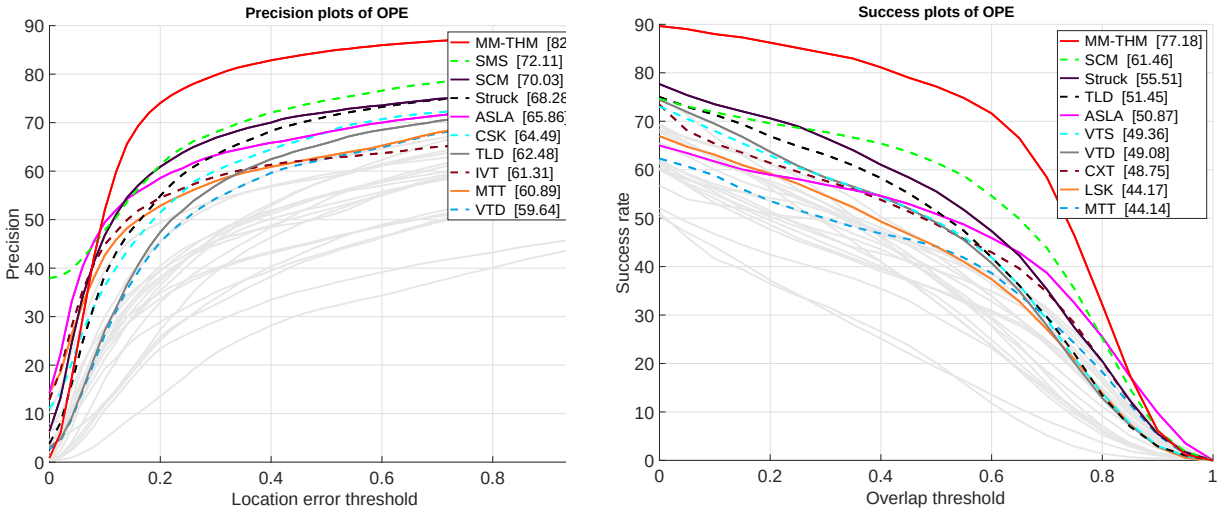


Figure 3.8: Tracking results of selected 11 trackers in representative frames. Frame indices are shown in the top left of each figure. The showing examples are from sequences carDark, Jogging, Singer1, Bolt, Walking2, Basketball, respectively.

Figure 3.8 shows a qualitative comparison with selected trackers on several representative videos/frames.

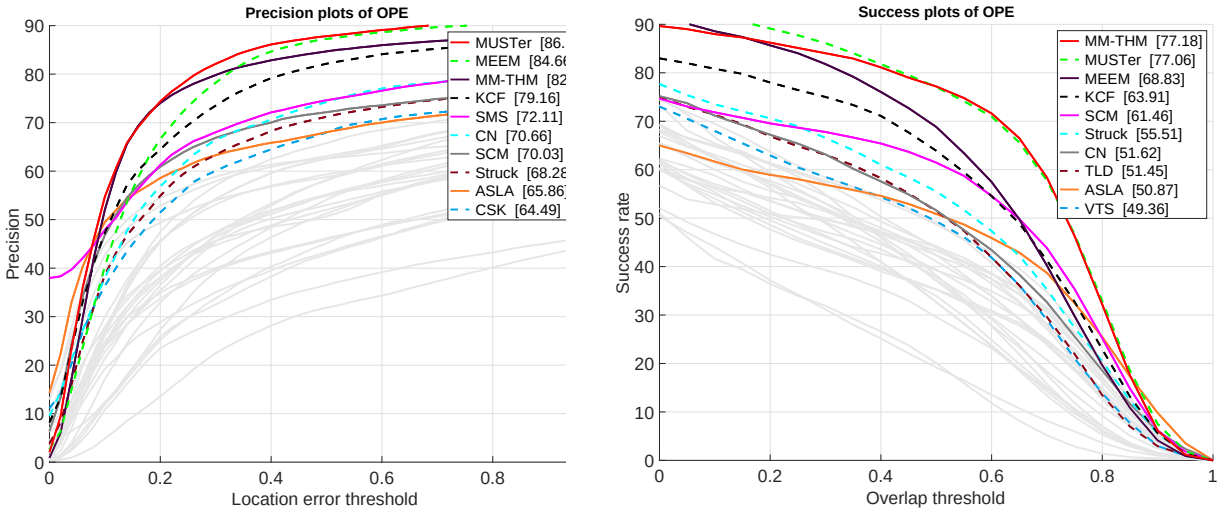
OTB evaluates the robustness of trackers based on two different metrics: the precision plot and success plot. The precision plot checks the performance of the tracker by checking the Center Location Error (CLE) to be less than a threshold (default value is 20). The success plot measures the Intersection Over Union (IOU) metrics for trackers on each frame to show the percentage of successfully tracked frames.



(a) (b)

Figure 3.9: (a) precision and (b) success plots for the 50 videos with all available trackers in the benchmark OTB-50. The proposed MM-THM achieves overall the best performance in both metrics and outperforms the second best tracker SMS and SCM more than 10% and 15%, respectively.

We compare the proposed method with all trackers whose results are available in [188] *e.g.* Struck [58], TLD [79], SCM [213], OAB [51] and CST [60]. The CT [205], LST [76], and MTT [191] are based on global, local, and joint sparse models, respectively. We add MTMV [62] that is another joint sparse tracker that exploit ℓ_{12}. We report MTMV and MTT results to show the effect of grouping by ℓ_{12} instead of tree structure. The results show that MM-THM tracker achieves performance gain of more than 18% from the other related sparse trackers [15, 76, 205, 191, 118, 119]. We also compare our method with recent trackers that show good performance in [92]: CN [30], MUSTer [61], KCF [59] and MEEM [204]. We compare quantitatively MM-THM with 34 trackers using the precision plot and success plot in Fig. (3.10), which indicates that MM-THM outperforms in both the metrics and significantly improves all the trackers of OTB benchmark including sparse based trackers like MTMV by 15% as well as other recent trackers like CN, KCF, and MEEM that are the top-performing trackers in [92]. The success plot shows that MM-THM outperforms MEEM by 8.2% and has similar performance with MUSTer and also achieves third best overall performance in precision plot with precision slightly less than



(a) (b)

Figure 3.10: (a) precision and (b) success plots for the 50 videos with all available trackers in the benchmark OTB-50 and recent high-performance trackers in [92]: CN [30], MUSTer [61], KCF [59] and MEEM [204]. The proposed MM-THM outperforms MEEM by 8.2% and has similar performance with MUSTer in success plot and also achieves third best overall performance in precision plot with precision 3.26% less than MUSTer.

MEEM and 2.5% less than MUSTer. We show precision and success rate metrics for all attributes including background clutter, occlusion, illumination and low resolution attributes in Figs. (3.13) and (3.14). We also provide the successful tracking rate (STR) in Table 3.1 as done in [15, 62, 191, 76, 205, 118, 119] on 10 image sequences. To observe the effect of each component of the proposed method, the result of three systems (p1)-(p3) are reported:

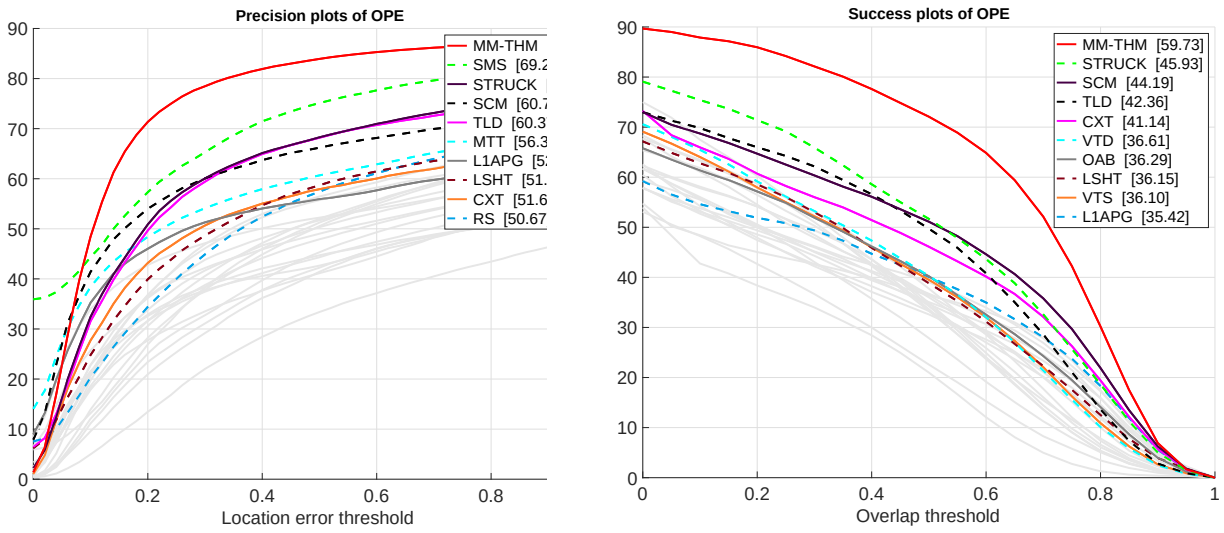
(p1) MM-THM with fixed dictionaries;

(p2) MM-THM (Eq.(3.4)) with constant group weights $\boldsymbol{\omega}$;

(p3) MM-THM without multimodal classifiers;

(p4) MM-THM, the whole system;

Overall precision and success metrics for system p2 has 5.8% and 5.4% drop with respect to p4, while these metrics for system p3 are 14.4% and 13.1% less than p4, respectively; which shows the importance of classifiers (decision-level fusion). Overall, the proposed MM-THM algorithm performs favorably against the other state-of-the-art sparse trackers on all tested sequences and show better performance than very recent trackers like MEEM, CN and KCF.



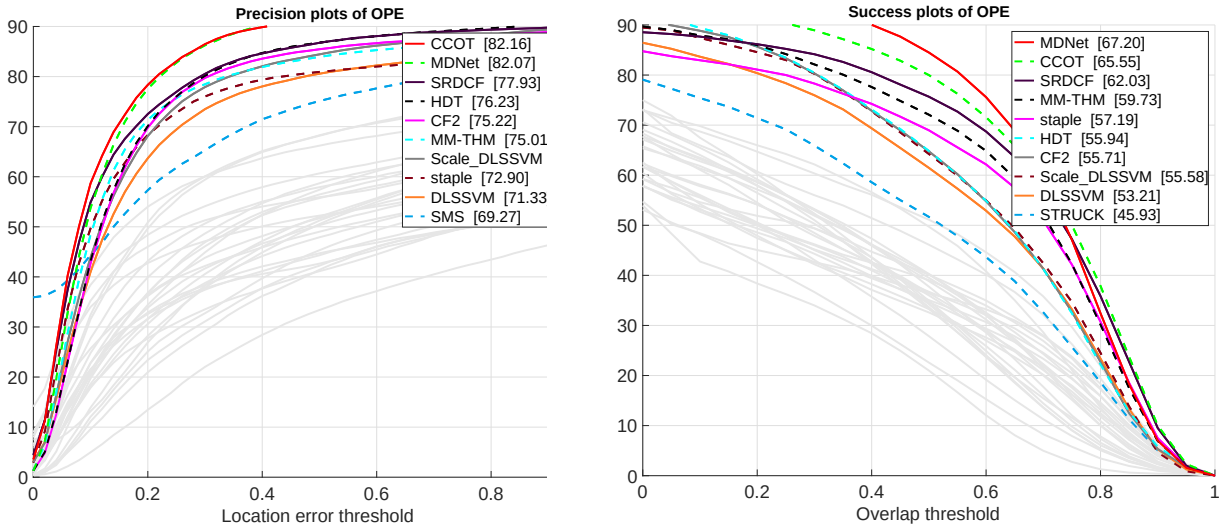
(a) (b)

Figure 3.11: (a) precision and (b) success plots for the 100 videos with all available trackers in the benchmark OTB-100. The proposed MM-THM achieves overall the best performance in both metrics and outperforms the second best tracker SMS and STRUCK more than 5% and 13%, respectively.

Table 3.2 and Table 3.3 show the average center location error and success rates for all the trackers with respect to all attributes at the average Euclidean distance between the center locations of the tracked targets and the manually labeled ground-truth as 20 pixels and overlap threshold of 0.5, respectively. The tracking algorithms are sorted by the average success rates, and the top-five methods denoted by different colors.

3.7 Conclusion

The visual tracking in the sparsity scheme was studied and a method was proposed to learn the unsupervised dictionary and classifier while obtaining multimodal sparse representation of each positive and negative patches using tree-structure sparsity model. The imposed tree-structured joint sparsity enabled the algorithm to fuse information at feature-level in different granularity by forcing their sparse codes to have similar basis within each group and at decision-level by augmenting the classifier decisions. In contrast to other tree-structured sparsity models that assign constant weights, we automatically assign them by getting



(a) (b)

Figure 3.12: Comparison with state-of-the-art deep learning trackers. (a) precision and (b) success plots trackers in the benchmark OTB-100.

feedback from their proposed reliability measure. The experimental results shows that the proposed method outperforms state-of-the-art trackers in challenging scenarios.

Our designed scheme is originated from matrix factorization. It is designed to make a connection between modalities using a latent factors space. This way, our framework extracts the common structure of all the modalities and provides the projection of one representation on an alternative space. In our latent factor model, the correlation between modalities directly depends on the modality configuration that we found before. Intuitively, we say that modalities have the same underlying semantics in the latent space. We target learning cross-modality correlations while at the same time try prohibiting false co-adaptations between data representations and ensuring robustness of the classifier to missing signals and signal corruption.

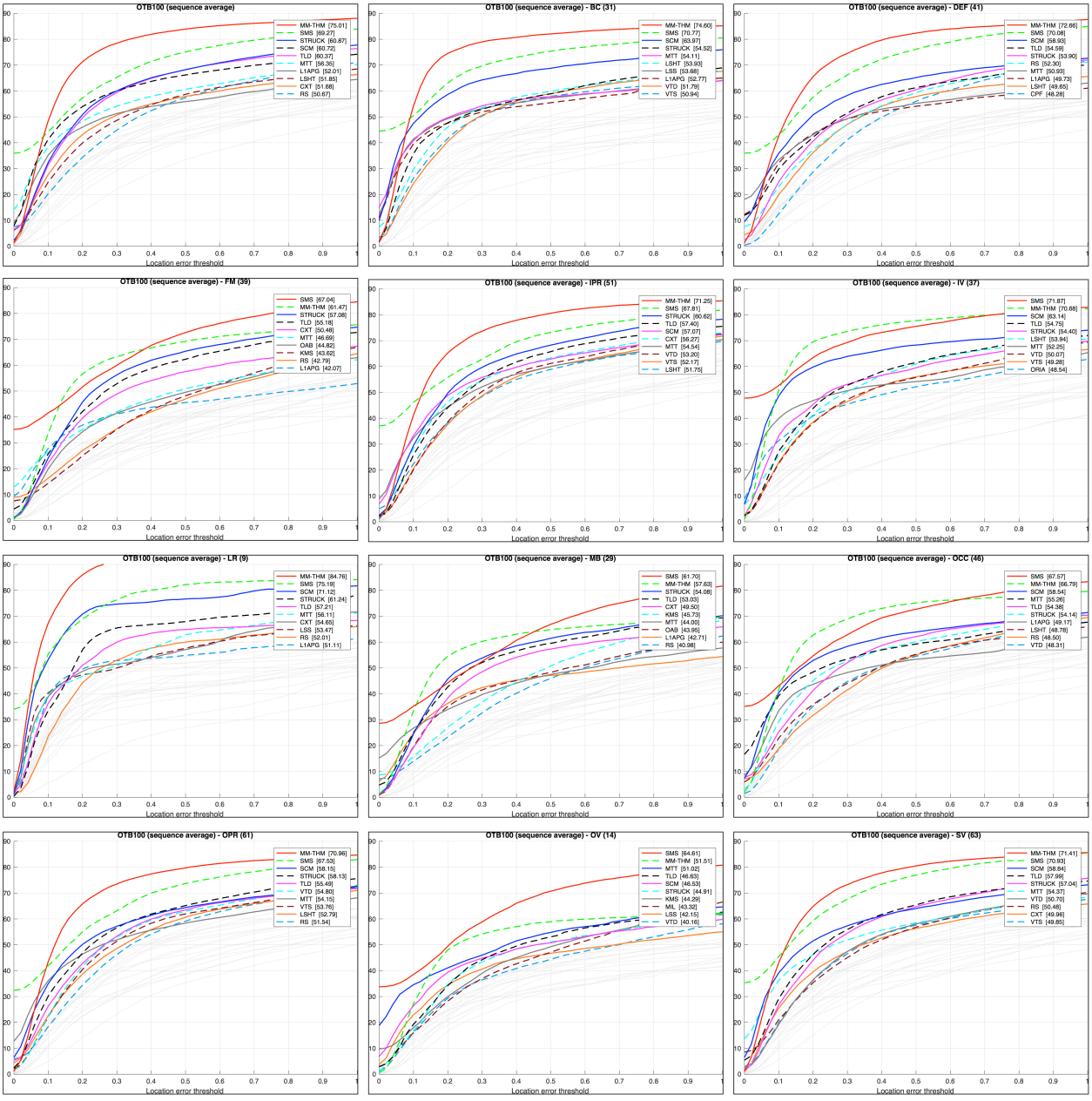


Figure 3.13: Comparing MM-THM in precision plot with trackers in OTB-100 in all attributes. The score for each tracker is shown in the legend. The top 10 trackers are presented for the sake of clarity, and the rest are shown as gray dashed curves.

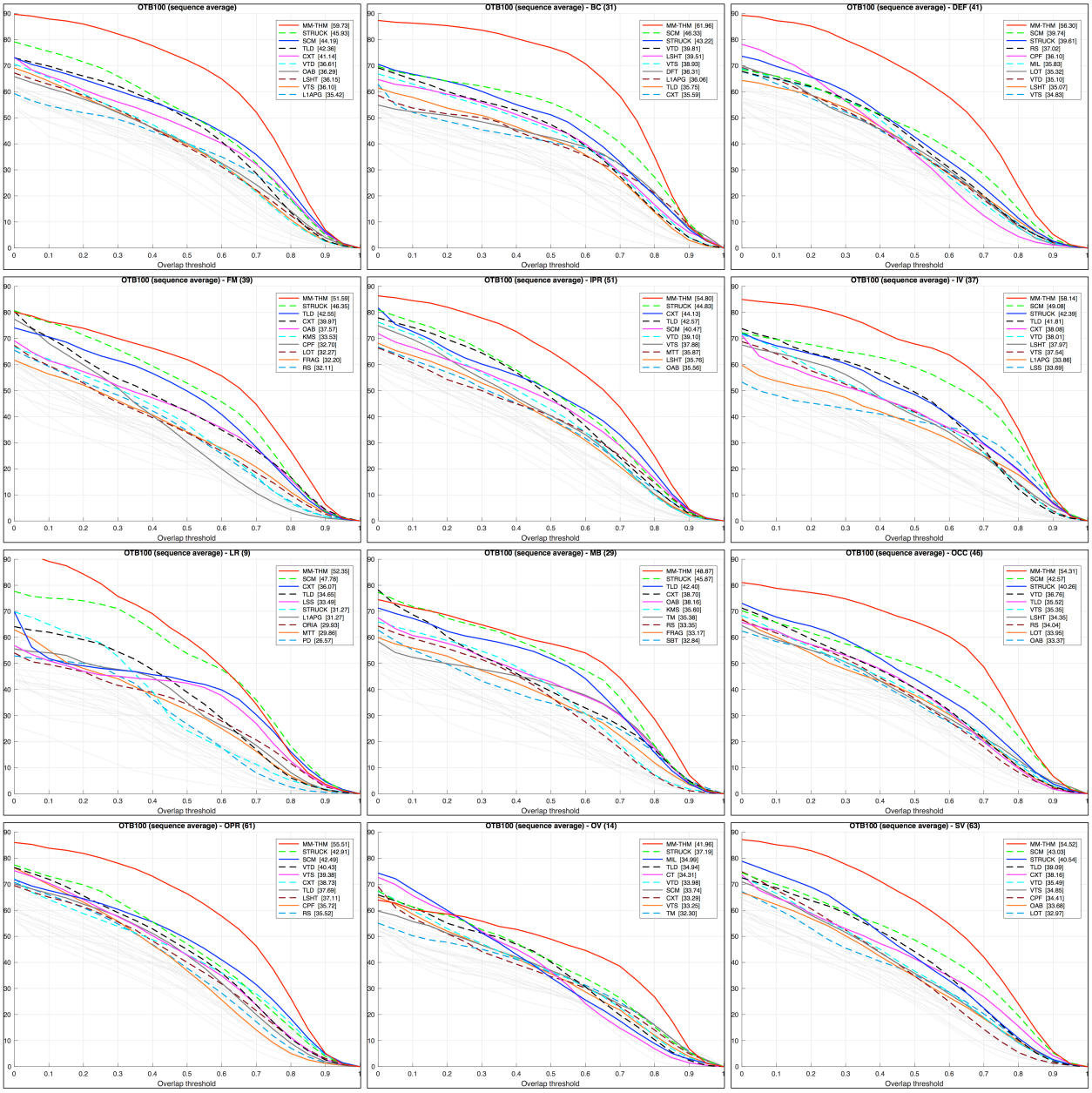


Figure 3.14: Comparing MM-THM in success plot with OTB-100 trackers in all attributes. The score for each tracker is shown in the legend. The top 10 trackers are presented for the sake of clarity, and the rest are shown as gray dashed curves.

Table 3.2: Precision (Center Location Error) in OTB-100 (sequence average). The trackers are ordered by the average overlap scores, and the top 5 methods in each attribute are denoted by different colors: **red**, **green**, **blue**, **cyan**, and **magenta**.

Attributes	All	BC	DEF	FM	IPR	IV	LR	MB	OCC	OPR	OV	SV
MM-THM	**75.0**	**74.6**	**72.7**	**61.5**	**71.2**	**70.7**	**84.8**	**57.6**	**66.8**	**71.0**	**51.5**	**71.4**
SMS	**69.3**	**70.8**	**70.1**	**67.0**	**67.8**	**71.9**	**75.2**	**61.7**	**67.6**	**67.5**	**64.6**	**70.9**
STRUCK	**60.9**	**54.5**	**53.9**	**57.1**	**60.6**	**54.4**	**61.2**	**54.1**	54.1	**58.1**	44.9	**57.0**
SCM	**60.7**	**64.0**	**58.9**	38.3	**57.1**	**63.1**	**71.1**	35.6	**58.5**	**58.1**	**46.5**	**58.8**
TLD	**60.4**	50.9	**54.6**	**55.2**	**57.4**	**54.7**	**57.2**	**53.0**	**54.4**	**55.5**	**46.6**	**58.0**
MTT	56.3	**54.1**	50.9	46.7	54.5	52.2	56.1	44.0	**55.3**	54.2	**51.0**	54.4
L1APG	52.0	52.8	49.7	42.1	51.4	46.4	51.1	42.7	49.2	48.2	34.7	46.9
LSHT	51.9	53.9	49.7	36.6	51.8	53.9	49.5	32.3	48.8	52.8	36.4	47.6
CXT	51.7	42.5	36.6	**50.5**	56.3	47.4	54.7	**49.5**	42.1	49.1	39.5	50.0
RS	50.7	49.4	52.3	42.8	48.5	43.0	52.0	41.0	48.5	51.5	39.7	50.5
VTD	49.9	51.8	47.0	35.6	53.2	50.1	47.4	31.0	48.3	54.8	40.2	50.7
VTS	48.9	50.9	46.4	35.9	52.2	49.3	45.3	29.6	46.5	53.8	39.4	49.9
LSS	48.6	53.7	44.6	32.6	47.9	47.7	53.5	31.7	47.8	49.7	42.2	46.8
OAB	46.4	39.8	38.7	44.8	45.2	41.5	41.6	44.0	42.1	42.9	35.2	45.3
MIL	46.2	45.1	47.2	37.8	48.9	40.1	49.4	31.5	46.2	49.4	43.3	45.5
KMS	45.2	41.5	47.5	43.6	44.8	39.9	41.0	45.7	45.5	46.9	44.3	46.2
ORIA	45.1	43.9	36.4	31.1	47.4	48.5	46.0	29.2	42.5	46.4	38.2	44.7
LOT	44.8	40.8	47.0	38.0	44.6	33.8	39.1	35.9	44.6	47.3	36.5	44.0
CPF	44.6	37.1	48.3	37.0	44.4	37.2	35.3	31.8	43.3	48.2	37.0	45.8
FRAG	42.7	38.7	39.8	38.9	43.1	34.6	38.3	37.6	38.9	43.5	38.3	40.8
SBT	41.1	36.2	37.3	36.3	39.7	33.3	38.5	36.7	35.6	36.2	30.1	37.5
TM	40.3	33.6	36.3	36.6	41.2	34.3	39.7	38.9	34.6	39.1	39.4	38.1
DFT	40.0	41.8	39.5	28.6	40.4	38.9	40.2	24.6	40.4	42.4	31.3	34.5
PD	39.4	31.2	38.8	34.0	38.3	32.9	46.9	35.0	38.5	37.7	32.8	40.0
BSBT	38.6	31.5	32.2	32.0	38.9	34.6	31.1	33.8	33.0	36.2	29.6	34.0
VR	36.9	32.7	36.3	30.7	37.9	24.6	41.6	32.6	35.8	36.0	29.5	36.9
CT	36.4	33.3	37.1	28.8	36.4	33.4	42.5	25.7	38.9	38.7	40.2	37.8
MS	29.3	23.9	27.6	29.6	30.2	24.5	23.6	31.0	25.2	30.6	28.2	32.3

Table 3.3: Success rate (overlap) in OTB-100 (sequence average). Each entry contains the average overlap in percentage at the overlap threshold of 0.5. The trackers are ordered by the average overlap scores, and the top 5 methods in each attribute are denoted by different colors: **red**, **green**, **blue**, **cyan**, and **magenta**.

Attributes	All	BC	DEF	FM	IPR	IV	LR	MB	OCC	OPR	OV	SV
MM-THM	**59.7**	**62.0**	**56.3**	**51.6**	**54.8**	**58.1**	**52.3**	**48.9**	**54.3**	**55.5**	**42.0**	**54.5**
STRUCK	**45.9**	**43.2**	**39.6**	**46.3**	**44.8**	**42.4**	31.3	**45.9**	**40.3**	**42.9**	**37.2**	**40.5**
SCM	**44.2**	**46.3**	**39.7**	29.4	**40.5**	**49.1**	**47.8**	27.1	**42.6**	**42.5**	33.7	**43.0**
TLD	**42.4**	35.7	32.5	**42.5**	**42.6**	**41.8**	**34.6**	**42.4**	**35.5**	37.7	**34.9**	**39.1**
CXT	**41.1**	35.6	29.3	**40.0**	**44.1**	**38.1**	**36.1**	**38.7**	33.3	38.7	33.3	**38.2**
VTD	36.6	**39.8**	35.1	27.6	39.1	38.0	26.0	26.1	**36.8**	**40.4**	34.0	35.5
OAB	36.3	31.8	30.8	**37.6**	35.6	32.3	22.6	**38.2**	33.4	33.3	31.4	33.7
LSHT	36.2	**39.5**	35.1	26.2	35.8	38.0	23.2	22.7	34.4	37.1	28.9	30.2
VTS	36.1	38.9	34.8	27.6	37.9	37.5	24.9	25.0	35.3	**39.4**	33.2	34.9
L1APG	35.4	36.1	29.6	28.5	35.1	33.9	31.3	32.7	32.0	32.2	26.9	30.6
RS	34.8	33.2	**37.0**	32.1	32.7	27.1	24.9	33.4	34.0	35.5	31.4	32.2
MTT	34.4	34.3	26.1	29.7	35.9	31.4	29.9	26.8	30.5	32.2	26.2	30.6
CPF	34.1	27.8	**36.1**	32.7	33.4	27.0	21.2	29.4	33.3	35.7	32.0	34.4
LOT	33.8	31.6	35.3	32.3	32.1	26.6	21.1	31.1	33.9	35.1	32.1	33.0
FRAG	33.4	29.0	32.0	32.2	32.0	26.2	22.5	33.2	30.4	32.8	30.0	30.0
MIL	33.2	34.6	35.8	29.3	34.1	28.5	24.8	25.3	33.2	35.1	**35.0**	31.6
TM	33.1	27.6	29.8	32.0	33.6	29.7	22.4	35.4	28.8	31.2	32.3	30.0
SBT	32.8	28.3	28.9	31.1	30.6	27.9	20.8	32.8	27.9	28.9	26.5	28.8
DFT	32.3	36.3	33.1	26.0	32.0	32.9	22.6	24.4	33.1	33.5	27.7	26.3
KMS	32.2	28.2	34.0	33.5	30.7	28.3	17.7	35.6	31.0	32.6	32.1	31.2
LSS	32.0	32.3	25.7	21.7	30.6	33.7	**33.5**	20.9	30.3	31.3	30.8	30.2
PD	31.8	26.1	32.7	30.0	29.6	26.7	26.6	32.6	30.7	30.5	26.7	30.8
BSBT	31.2	25.1	25.2	28.8	30.7	29.0	16.6	31.1	26.9	28.7	25.8	26.3
ORIA	30.8	29.7	23.6	20.0	33.1	32.8	29.9	18.1	30.4	31.2	23.1	29.2
VR	30.3	28.1	31.8	27.4	29.2	21.0	22.9	31.3	28.7	29.3	25.1	29.3
CT	28.1	27.6	30.1	24.5	26.9	27.4	18.5	23.6	30.8	29.7	**34.3**	27.9
MS	23.6	20.0	23.5	26.6	22.8	20.5	8.3	28.3	21.8	24.6	26.7	25.8
SMS	21.3	16.0	22.2	23.6	19.8	16.4	17.6	24.4	22.0	23.0	24.4	23.3

Table 3.4: Comparing the best trackers of OTB-100, and Deep Learning trackers with MM-THM using Precision rate (Center Location Error) in OTB-100 (sequence average). Each entry contains the average overlap in percentage at the overlap threshold of 0.5. The trackers are ordered by the average overlap scores, and the top 5 methods in each attribute are denoted by different colors: **red**, **green**, **blue**, **cyan**, and **magenta**.

CCOT	**82.2**	**81.9**	**77.9**	**80.3**	**79.5**	**81.6**	**90.0**	**83.1**	**82.1**	**80.3**	**79.3**	**82.9**
MDNet	**82.1**	**83.7**	**79.2**	**78.6**	**80.3**	**82.1**	**87.1**	**76.3**	**76.2**	**80.0**	**75.0**	**81.1**
SRDCF	**77.9**	**80.6**	69.3	**73.5**	73.9	**78.6**	70.3	**75.3**	**71.8**	**73.4**	**63.7**	**76.1**
HDT	**76.2**	**76.2**	**73.3**	**72.4**	**75.0**	72.3	**78.1**	**70.3**	**68.8**	**71.6**	**62.2**	**72.8**
CF2	**75.2**	**75.8**	**70.3**	**72.0**	**75.8**	**72.3**	75.2	**71.7**	68.0	71.5	60.6	**71.9**
MM-THM	75.0	74.6	**72.7**	61.5	71.2	70.7	**84.8**	57.6	66.8	71.0	51.5	71.4
Scale_DLSSVM	75.0	73.7	70.0	67.3	**74.4**	**73.2**	**87.1**	66.9	**69.5**	**71.9**	61.8	70.6
STAPLE	72.9	71.0	69.9	65.2	69.9	71.5	66.3	64.3	67.5	67.8	59.5	68.3
DLSSVM	71.3	70.3	66.1	68.4	71.0	67.9	75.5	69.7	65.4	69.7	**62.2**	67.6
Struck	60.9	54.5	53.9	57.1	60.6	54.4	61.2	54.1	54.1	58.1	44.9	57.0
SCM	60.7	64.0	58.9	38.3	57.1	63.1	71.1	35.6	58.5	58.1	46.5	58.8
MTT	56.3	54.1	50.9	46.7	54.5	52.2	56.1	44.0	55.3	54.2	51.0	54.4

Table 3.5: The best trackers of OTB-100, and Deep Learning trackers are compared with MM-THM using Success rate (overlap) in OTB-100 (sequence average). Each entry contains the average overlap in percentage at the overlap threshold of 0.5. The trackers are ordered by the average overlap scores, and the top 5 methods in each attribute are denoted by different colors: **red**, **green**, **blue**, **cyan**, and **magenta**.

MDNet	**67.2**	**67.4**	**63.5**	**66.2**	**64.5**	**67.6**	**63.0**	**66.1**	**63.2**	**65.1**	**62.3**	**65.7**
CCOT	**65.5**	**64.2**	**59.3**	**65.9**	**61.1**	**66.0**	**62.4**	**68.4**	**65.3**	**63.5**	**63.9**	**64.8**
SRDCF	**62.0**	**64.0**	**52.9**	**59.1**	**56.2**	**63.4**	**51.5**	**62.3**	**56.8**	**57.6**	**50.6**	**60.5**
MM-THM	**59.7**	**62.0**	**56.3**	51.6	**54.8**	**58.1**	**52.3**	48.9	**54.3**	**55.5**	42.0	**54.5**
STAPLE	**57.2**	55.9	**55.4**	52.6	53.7	**57.7**	39.9	52.1	**54.4**	**53.6**	**47.2**	**51.7**
HDT	55.9	57.6	52.3	**55.5**	54.5	52.2	40.1	55.7	50.8	51.9	47.0	48.4
CF2	55.7	**58.3**	51.0	**55.9**	**55.0**	52.9	38.8	**56.9**	50.7	52.1	**47.1**	48.4
Scale_DLSSVM	55.6	54.5	50.2	52.7	54.7	56.2	43.5	**56.2**	52.5	53.4	46.5	49.3
DLSSVM	53.2	51.6	49.1	52.8	52.2	51.1	37.6	55.3	48.9	51.6	46.5	46.4
Struck	45.9	43.2	39.6	46.3	44.8	42.4	31.3	45.9	40.3	42.9	37.2	40.5
SCM	44.2	46.3	39.7	29.4	40.5	49.1	**47.8**	27.1	42.6	42.5	33.7	43.0
MTT	34.4	34.3	26.1	29.7	35.9	31.4	29.9	26.8	30.5	32.2	26.2	30.6

Chapter 4

Supervised Dictionary Learning

4.1 Introduction

Dictionary learning methods can be categorized into two parts: unsupervised and supervised algorithms. In unsupervised dictionary learning the optimization formula only has reconstruction penalty, and therefore, the dictionary is adapted to the data. We elaborate unsupervised dictionary learning in Chapter 2. The unsupervised dictionary learning methods are applied to mostly reconstructive tasks like signal and image denoising [36]. Although in unsupervised approach there is no discriminative penalty, the obtained dictionary is applied for discriminative tasks like classification [198, 185, 193]. It has been shown that if the dictionary is learned to adapt to the specific task and not only to the data, the classification result will improve [104, 68]. This type of dictionary learning is supervised and sometimes called the task-driven method. The higher classification accuracy achieved because the error in supervised methods are based on misclassification and not only the reconstruction; hence, the dictionary is more suitable for the discriminative task like classification rather than to reconstruct the data [159, 65]. Some studies have investigated weighted mixture of discrimination and reconstruction errors [77, 206, 111]. Unsupervised dictionary learning is accounted as large-scale matrix factorization and solved efficiently in [105, 63]. The supervised scheme, in contrast, is more complicated to solve. Bilevel

optimization scheme especially stochastic gradient descent algorithm is used to solve the supervised dictionary learning [192], and the efficiency is tested for different discriminative tasks like compressive sensing and classification.

Most of the dictionary learning studies, unsupervised or supervised, are designed for single feature modality. Supervised dictionary learning using multiple feature modalities is studied for action recognition in [211]. Each view is considered as a modality, and a dictionary is trained for each view to capture specific information of each camera. Also, a shared dictionary is learned to model the correlation between different cameras. Although the method minimizes reconstruction error, it exploits class labels to make class-specific atoms. The method can only handle fusion between feature modalities of the same size. Constrained supervised dictionary learning is used for text and image modalities retrieval [216]. The dictionaries are learned by joint reconstruction error minimization across all modalities. However, this method only relies on reconstruction error and does not consider any discriminative error term. Also, the dictionary learning method does not exploit the valuable correlation between modalities.

One important distinction between supervised and unsupervised dictionary learning which is not proved mathematically or empirically is the relation between number of atoms and dimensionality of the signal. The unsupervised dictionary should be overcomplete: the dictionaries have more atoms than the signal dimension. In particular for image processing applications, this is reported to lead to better reconstruction [36, 108]. On the other hand, perfect reconstruction is not always required for discriminative tasks, as long as the sparse coding procedure captures discriminative latent features. That is, with supervised learning, we expect to get better classification result with small and compact dictionaries, $p < n_m$.

4.2 Single Modal

4.2.1 Estimation of Dictionary and Classifier: Independent

The framework is to learn a dictionary in training phase in an unsupervised scheme. The learned dictionary is used to extract sparse code coefficients of the test signal using Lasso or basis pursuit (1.2,1.3). In [185, 14, 67] the test signal is assigned to the class that can approximate the test signal with minimum reconstruction error. But, utilizing class labels of the data in a misclassification error is more reasonable for the classification task. Therefore, some methods use obtained sparse code $\boldsymbol{\alpha}^\star(\boldsymbol{x}, \boldsymbol{D})$ as latent features for representing the training data and learn a classifier. Some of existing sparse coding approaches train a classifier for each pair of categories or they train an independent classifier in the one-against-all scheme [193, 107, 194, 111]. Some studies try to train a universal multi-class classifier in the all-against-all scheme [107, 110, 70]. The advantage of all-against-all classifier lies in the fact that the classifier is obtained by looking at all classes at the same time, and the method can model shared features between classes.

Learning dictionary for a specific task have shown to get a better result than unsupervised methods. For example in [110, 77] the learned dictionary designed for compressed sensing and classification showed to have superior results than unsupervised learning in (2.9).

Data-Driven Dictionary Learning. The classical dictionary learning framework having already been introduced in details in the previous Chapters, we just briefly recall the formulation here to fix the notations. Assume a finite set of training samples $\boldsymbol{X} = [\boldsymbol{x}^1, \ldots, \boldsymbol{x}^N] \in \mathbf{R}^{m \times N}$. Classical dictionary learning techniques for sparse coding ([135, 2, 154]) assume a finite training signals $\boldsymbol{X} = [\boldsymbol{x}^1, \ldots, \boldsymbol{x}^N]$ in $\mathbf{R}^{m \times N}$ and minimize the empirical cost function

$$f_n(\boldsymbol{D}) \triangleq \frac{1}{N} \sum_{i=1}^{N} (\mathcal{L}_u(\boldsymbol{x}^i, \boldsymbol{D})),$$

with dictionary $\boldsymbol{D}$ in $\mathbf{R}^{m \times p}$ as optimization variable, whose each column is an atom or dictionary element. The unsupervised loss function $\mathcal{L}_u$ should be designed so that the learned dictionary $\boldsymbol{D}$ is good at representing the input data $\boldsymbol{x}$ in sparse scheme. The subscript u in

$\mathcal{L}_u$ emphasizes that the loss function is data-driven and does not use the labels. Following [154, 106], the optimal value of a sparse coding problem is defined as $\mathcal{L}_u(\boldsymbol{x}, \boldsymbol{D})$, which we use elastic-net [217] over Lasso or basis pursuit for the stability reasons

$$\mathcal{L}_u(\boldsymbol{x}, \boldsymbol{D}) \triangleq \underset{\boldsymbol{\alpha} \in \mathbf{R}^p}{\operatorname{argmin}} \frac{1}{2}\|\boldsymbol{x} - \boldsymbol{D}\boldsymbol{\alpha}\|_2^2 + \lambda_1 \|\boldsymbol{\alpha}\|_1 + \frac{\lambda_2}{2}\|\boldsymbol{\alpha}\|_2^2 \tag{4.1}$$

where regularization parameters are λ_1 and λ_2. If $\lambda_2 = 0$, we have ℓ_1 sparse decomposition problem, also known as Lasso [176], or basis pursuit [26]. With $\lambda_2 > 0$ the optimization problem (4.1) would be strongly convex. We will show in this chapter that it guarantees its solution to be unique and Lipschitz on $\boldsymbol{x}$ and $\boldsymbol{D}$ with a constant depending on λ_2. We will show in experiments that stability does not play an important role in learning a dictionary for a reconstruction task; however, it is a crucial issue for discrimination tasks.

Without any restriction on columns of the dictionary, we would get small values of sparse representation. To solve this issue, the ℓ_2 norm of each dictionary element $\{\boldsymbol{d}^i\}_{i \in [\![1;p]\!]}$ is regularized to be less than or equal to one. The convex set of all eligible dictionary candidates $\mathcal{D}$ is defined as

$$\mathcal{D} \triangleq \{\boldsymbol{D} \in \mathbf{R}^{m \times p} \text{ s.t. } \forall k \in \{1, 2, \ldots, p\}, \|\boldsymbol{d}^k\|_2^2 \leq 1\} \tag{4.2}$$

the members of the convex set $\mathcal{D}$ are matrices in $\mathbf{R}^{m \times p}$ which their columns are in the unit ball of the ℓ_2 norm. As mentioned in [19, 64] the dictionary should be obtained by minimizing expected cost $f(\boldsymbol{D})$. Minimizing empirical cost $f_n(\boldsymbol{D})$ with high precision obtains a dictionary that is sub-optimum to represent data in general. The reason lies in the fact the empirical cost is an approximation of the expected cost. In [105] an inaccurate solution but with better expected cost for $\boldsymbol{D}$ is proposed in online scheme using the expected cost

$$f(\boldsymbol{D}) \triangleq \mathbb{E}_{\boldsymbol{x}}\big[\mathcal{L}_u(\boldsymbol{x}, \boldsymbol{D})\big] = \lim_{N \to \infty} f_N(\boldsymbol{D}) \text{ a.s.} \tag{4.3}$$

the expectation is taken relative to the (unknown) probability distribution of the data $p(\boldsymbol{x})$. The coefficients $\boldsymbol{\alpha}^\star(\boldsymbol{x}, \boldsymbol{D})$ generated by a given dictionary $\boldsymbol{D}$ for decomposing the sample

$\boldsymbol{x} \in \mathcal{X}$ is estimated using elastic-net formula [217].

$$\boldsymbol{\alpha}^\star(\boldsymbol{x}, \boldsymbol{D}) \triangleq \underset{\boldsymbol{\alpha} \in \mathbf{R}^p}{\operatorname{argmin}} \frac{1}{2}\|\boldsymbol{x} - \boldsymbol{D}\boldsymbol{\alpha}\|_2^2 + \lambda_1 \|\boldsymbol{\alpha}\|_1 + \frac{\lambda_2}{2}\|\boldsymbol{\alpha}\|_2^2 \tag{4.4}$$

In this chapter, our goal is to design a sparse coding method to learn a dictionary so that the generated sparse representation is suitable for the classification task. Let us denote a finite set $\mathcal{Y}$ with C members as labels for classification tasks. A more concrete explanation is that we are looking for a variable $\boldsymbol{y}$ in $\mathcal{Y}$ from each sample $\boldsymbol{x}$ in $\mathcal{X} \in \mathbf{R}^m$. The learned unsupervised dictionary using (4.3) are largely utilized for classification tasks in two ways: 1. The estimated decomposition coefficients $\boldsymbol{\alpha}^\star(\boldsymbol{x}, \boldsymbol{D})$ is used to approximate data $\boldsymbol{x}$ as $\hat{\boldsymbol{x}} \approx \boldsymbol{D}\boldsymbol{\alpha}^\star$, and the reconstruction error $\|\boldsymbol{x} - \hat{\boldsymbol{x}}\|_2^2$ is shown to be a robust measure for classification tasks [14, 185]. 2. A classifier is trained using the generated $\boldsymbol{\alpha}^\star(\boldsymbol{x}, \boldsymbol{D})$ as a latent feature vector of data $\boldsymbol{x}$ for predicting the variable $\boldsymbol{y}$ in the classical expected risk optimization

$$\underset{\boldsymbol{W} \in \mathcal{W}}{\operatorname{argmin}} \ \mathbb{E}_{\boldsymbol{y},\boldsymbol{x}}[\mathcal{L}_s\big(\boldsymbol{y}, \boldsymbol{W}, \boldsymbol{\alpha}^\star(\boldsymbol{x}, \boldsymbol{D})\big)] + \frac{\nu}{2}\|\boldsymbol{W}\|_F^2 \tag{4.5}$$

where the loss $\mathcal{L}_s$ with subscript s is a supervised learning method. It evaluates classifier by how close it can find the label $\boldsymbol{y}$ given the latent feature $\boldsymbol{\alpha}^\star(\boldsymbol{x}, \boldsymbol{D})$. The expectation is taken with respect to the unknown probability distribution $p(\boldsymbol{y}, \boldsymbol{x})$ of the data.

The major issues of this approach are as follows:

- The dictionary is obtained in the unsupervised scheme and independent of the labels.
- The features to learn a classifier are decomposition coefficients that are produced by a dictionary that does not have any information about the labels $\mathcal{Y}$. So, the method does not fully utilize the label information.
- The dictionary $\boldsymbol{D}$ is fixed during training classifier.

4.2.2 Estimation of Dictionary and Classifier: Jointly

In supervised learning scheme, the goal is to estimate dictionary $\boldsymbol{D}$ while exploiting class labels. This can be done in two schemes: one-against-all and all-against-all. In one-against-all a class-specific dictionary $\boldsymbol{D}_c$ is trained on data of c-th class using Eq. (4.3). Then all the class-specific dictionaries are concatenated horizontally to make the final dictionary $\boldsymbol{D} = [\boldsymbol{D}_1, \boldsymbol{D}_2, \cdots, \boldsymbol{D}_C]$ [198, 66, 145]. The issue in this approach is that each sub-dictionary $\boldsymbol{D}_c$ obtained independent of other classes. Often, classes are not completely independent from each other, and they have some features in common. Since the class-specific dictionaries $\boldsymbol{D}_c$ obtained independent of each other, it is high probable that they have similar atoms which lead to similar sparse representations for samples that belong to the different classes and hence degrades classification accuracy. Since, in each iteration, most of the computational time is specified for estimation of the coefficients vectors $\{\boldsymbol{\alpha}^i\}_{i\in[\![1;N]\!]}$, having a large number of dictionary elements increases the computational time. In practice, one-vs-all dictionary learning methods require relatively large dictionaries to achieve good classification performance, leading to high computation cost. To obtain a dictionary with independent elements, the optimization formula is designed to enforce class-specific dictionaries to be uncorrelated [147, 140, 101, 141, 173, 172]; but still the dictionary learning is unsupervised and is only based on reconstruction error.

In all-against-all dictionary learning, a single dictionary is shared between all classes. The shared dictionary usually has a less number of atoms, which make the coding in the testing phase efficiently, but, there is no guarantee that each atom is representing a certain class. If an atom is adapted to multiple classes, the generated codes of that atom are not discriminative enough. The idea of estimating dictionary and classifier jointly in the all-against-all scheme while they are connected via sparse codes proposed in [104] and it outperforms other sparsity based methods.

$$\underset{\boldsymbol{W}\in\mathcal{W},\,\boldsymbol{D}\in\mathcal{D}}{\operatorname{argmin}} \quad f(\boldsymbol{D},\boldsymbol{W}) + \frac{\nu}{2}\|\boldsymbol{W}\|_F^2 \tag{4.6a}$$

$$f(\boldsymbol{D},\boldsymbol{W}) \triangleq \mathbb{E}_{\boldsymbol{y},\boldsymbol{x}}\big[\mathcal{L}_s\big(\boldsymbol{y},\boldsymbol{W},\boldsymbol{\alpha}^\star(\boldsymbol{x},\boldsymbol{D})\big)\big] \tag{4.6b}$$

where, the set of all possible choices of dictionaries $\mathcal{D}$ is defined as Eq. (4.2), $\boldsymbol{y}$ is the label of input data $\boldsymbol{x}$, and $\boldsymbol{W}$ is parameters of the classifier. The supervised loss function is mentioned as $\mathcal{L}_s$. The optimal sparse codes $\boldsymbol{\alpha}^\star(\boldsymbol{x}, \boldsymbol{D})$ is obtained from data $\boldsymbol{x}$ and dictionary $\boldsymbol{D}$ using elastic-net (4.4). The optimization problem (4.6a) is challenging because it is non-differentiable with respect to $\boldsymbol{\alpha}^\star(\boldsymbol{x}, \boldsymbol{D})$ which is the elastic-net solution of the problem (4.4). This issue is addressed in [20] by introducing a smooth function for sparsity regulation so that the gradient of loss function with respect to sparse representation can be calculated. This simplification leads to a smooth and non-sparse solution and so most of the elements of $\boldsymbol{\alpha}^\star$ are not truly zero, where $\boldsymbol{\alpha}^\star$ is shorthand for $\boldsymbol{\alpha}^\star(\boldsymbol{x}, \boldsymbol{D})$.

Explicitly Defined Dictionary. In this section, we consider two state-of-the-art supervised dictionary learning methods that are originally designed for single modality and extend them to do fusion at feature-level and compare their performance on different classification tasks. In particular, the method in Sec. (4.3) is an all-against-all scheme [172] that learns a single dictionary shared between all classes and the algorithm in Sec. (4.4) learns independent dictionaries for every class (one-against-all) [174].

4.3 Multimodal: All-Against-All

Label Consistent K-SVD

In this section, we briefly introduce LCKSVD [77] that is equivalent to JDL for the case of one single modality ($m = 1$). LCKSVD minimizes a mixture of classification and reconstruction errors and generates a discriminative and compact dictionary in an all-vs-all scheme. In order to learn dictionaries with uncorrelated atoms, LCKSVD forces each atom to represent only one class. Assuming i-th training sample $\boldsymbol{x}^i$ from the c-th class, a binary vector $\boldsymbol{q}^i \in \mathbf{R}^p$ is defined that is zero everywhere except at the indices of atoms which belong to the c-th class. This so called "label consistency constraint" is applied using $\{\boldsymbol{q}^i\}_{i=1}^N$ so that the sample from c-th class is represented using the same subset of dictionary items associated with class

c:

$$\underset{\boldsymbol{D},\boldsymbol{T},\boldsymbol{W},\boldsymbol{\alpha}^i}{\operatorname{argmin}} \sum_{i=1}^{N} \|\boldsymbol{x}^i - \boldsymbol{D}\boldsymbol{\alpha}^i\|_{\ell_2}^2 + \alpha\|\boldsymbol{q}^i - \boldsymbol{T}\boldsymbol{\alpha}^i\|_{\ell_2}^2 + \beta\|\boldsymbol{y}^i - \boldsymbol{W}\boldsymbol{\alpha}^i\|_{\ell_2}^2 + \lambda\|\boldsymbol{\alpha}^i\|_{\ell_1} \tag{4.7}$$

where $\boldsymbol{T}$ is a linear transformation matrix, $\boldsymbol{W}$ is the parameters of a linear classifier and α and β are regularization paramters of label consistency and miss-classification error, respectively. The label consistency $\|\boldsymbol{q}^i - \boldsymbol{T}\boldsymbol{\alpha}^i\|_{\ell_2}^2$ regularization enforces the linear transformed version of original sparse codes $\boldsymbol{T}\boldsymbol{\alpha}^i$ to be most discriminative in the $\mathbf{R}^p$ space.

4.3.1 Coupling Latent Feature Spaces

We intend to generalize problem (4.7) to be able to fuse information at the feature level using a bilevel optimization to exploit relationships between sparse codes across different feature spaces. The outer-level objective enforces similarity across sparse codes for all modalities within each class, subject to inner-level constraints such that for each modality, the dictionary is reconstructive and has incoherent atoms. We propose the following objective function to jointly learn multimodal dictionaries $\{\boldsymbol{D}_m\}_{m=1}^M$, classifiers $\{\boldsymbol{W}_m\}_{m=1}^M$, linear transformations $\{\boldsymbol{T}_m\}_{m=1}^M$ and multimodal sparse coefficients $\boldsymbol{A}^i = [\boldsymbol{\alpha}_1^i, \ldots, \boldsymbol{\alpha}_M^i]$:

$$\sum_{m=1}^{M} \big(\frac{1}{2}\|\boldsymbol{x}_m^i - \boldsymbol{D}_m\boldsymbol{\alpha}_m^i\|_2^2 + \alpha\|\boldsymbol{q}^i - \boldsymbol{T}_m\boldsymbol{\alpha}_m^i\|_2^2 + \beta\|\boldsymbol{y}^i - \boldsymbol{W}_m\boldsymbol{\alpha}_m^i\|_2^2\big) + \frac{\nu}{2}\|\boldsymbol{A}^i\|_F^2 + \Omega(\boldsymbol{A}^i)$$

$$\Omega(\boldsymbol{A}^i) = \lambda_1\|\boldsymbol{A}^i\|_{\ell_{12}} + \lambda_2\|\boldsymbol{A}^i\|_{\ell_{11}} \tag{4.8a}$$

where ν is a regularization parameter for the Frobenius norm. We follow [77, 104] in order to assign regularization parameters *i.e.* $\lambda_1, \lambda_2, \alpha, \beta,$ and ν. The fusion between M different features of the sample $\{\boldsymbol{x}_{i,c}^m\}_{m=1}^M$ is enforced in the space of sparse codes using ℓ_{12} regularization, $\Omega(\boldsymbol{A}) = \sum_{r=1}^p \|\boldsymbol{A}_{r\rightarrow}\|_2$; where $\boldsymbol{A}_{r\rightarrow}$ is the r-th row of $\boldsymbol{A}$ and promotes a solution with sparse non-zero rows in $\boldsymbol{A}$. Hence, similar support is enforced on $\boldsymbol{A}^i$ at the column level of each dictionary $\boldsymbol{D}_m$. Joint sparsity gives a strong statistical co-occurrence structure: if a sample belongs to the c-th class most of its modalities should have the same

label, so knowing the label of one source can act as a strong prior for inferring the label of others. However, multimodal data analysis may become trickier in following situations: 1. in the presence of modalities that either contaminated with the different noise power or 2. proper reconstruction of different modalities require different sparsity levels. In this case, imposing ℓ_{12} may lead to suboptimal results.

We exploit a combination of the ℓ_{12} and ℓ_{11} norms to let the multimodal inputs have shared and private pattern. The non-zero pattern of $\boldsymbol{A}^i = [\boldsymbol{\alpha}_1^i, \ldots, \boldsymbol{\alpha}_M^i]$ has a strong relation with the selection of λ_1 and λ_2. Hence with increasing the ratio of λ_1/λ_2, sparse representation of different modalities are more motivated to collaborate and have a similar non-zero pattern. In contrary, decreasing the ratio λ_1/λ_2, let each modality of the signal to be reconstructed independent of other modalities. If $\lambda_1 = 0$, the above optimization problem is separable across the modalities and is equal to decision fusion. When $\nu > 0$ problem (4.8) is a generalization of elastic-net optimization [217]. Mairal et al. proved that this design leads to more stable results than Lasso ($\nu = 0$) in [104]. In our experiments, ν is assigned a small positive value as $\nu = 10^{-3}\lambda_1$. We will explain briefly in Sec. 4.3.1. We rewrite the optimization problem (4.8) as:

$$\underset{\boldsymbol{A}^i}{\operatorname{argmin}} \sum_{m=1}^{M} \frac{1}{2} \left\| \begin{pmatrix} \boldsymbol{x}_m^i \\ \sqrt{\alpha}\boldsymbol{q}^i \\ \sqrt{\beta}\boldsymbol{y}^i \end{pmatrix} - \begin{pmatrix} \boldsymbol{D}_m \\ \sqrt{\alpha}\boldsymbol{T}_m \\ \sqrt{\beta}\boldsymbol{W}_m \end{pmatrix} \boldsymbol{\alpha}_m^i \right\|_{\ell_2}^2 + \Omega(\boldsymbol{A}^i) \tag{4.9}$$

Assume $\boldsymbol{\mathcal{D}}_m = [\boldsymbol{D}_m^\top, \sqrt{\alpha}\boldsymbol{T}_m^\top, \sqrt{\beta}\boldsymbol{W}_m^\top]^\top$ and $\boldsymbol{\mathcal{Y}}_m^i = [\boldsymbol{x}_m^{i\top}, \sqrt{\alpha}\boldsymbol{q}_m^{i\top}, \sqrt{\beta}\boldsymbol{y}_m^{i\top}]^\top$. We normalize both $\boldsymbol{\mathcal{D}}_m$ and $\boldsymbol{\mathcal{Y}}_m^i$. Then, optimization problem (4.9) is converted to:

$$\underset{\{\boldsymbol{\mathcal{D}}_m\}_{m=1}^M, \boldsymbol{A}^i}{\operatorname{argmin}} f(\boldsymbol{A}^i) + \Omega(\boldsymbol{A}^i) \tag{4.10}$$

where $f(\boldsymbol{A}_i) = \sum_{m=1}^{M} \frac{1}{2}\|\boldsymbol{\mathcal{Y}}_m^i - \boldsymbol{\mathcal{D}}_m\boldsymbol{\alpha}_m^i\|_{\ell_2}^2$ is smooth and differentiable and $\Omega(\boldsymbol{A}^i)$ is the convex but non-smooth joint $\ell_{12} - \ell_{11}$ regularization of (4.8a). Hence, we use the alternating

direction method of multipliers (ADMM) [136] to obtain multimodal sparse codes for each training sample $\boldsymbol{A}^i = [\boldsymbol{\alpha}_1^i, \ldots, \boldsymbol{\alpha}_M^i]$.

Optimization

We first solve optimization (4.10) for multimodal sparse codes $\{\boldsymbol{A}^i\}_{i=1}^N$ while the multimodal dictionaries $\{\boldsymbol{\mathcal{D}}_m\}_{m=1}^M$ are initialized with training samples as in [104]. This part is done using the solution in Sec. (3.5.1)

Next, we exploit $\boldsymbol{A}_m = [\boldsymbol{\alpha}_m^1, \ldots, \boldsymbol{\alpha}_m^N]$ to update corresponding dictionary $\boldsymbol{\mathcal{D}}_m$. Each atom is updated using the classical projected stochastic gradient algorithm and orthogonally projected onto the compact set of the unit-norm ball: $\{\boldsymbol{D}_m | \text{s.t.} \forall j \in \{1, \ldots, p\}, \|\boldsymbol{d}_m^j\|_{\ell_2} \leqslant 1\}$ following [104, 77].

So far we have found multimodal $\boldsymbol{\mathcal{D}}_m$ and $\{\boldsymbol{A}^i\}_{i=1}^N$. Since each column of $\boldsymbol{\mathcal{D}}_m$ is normalized, we obtain the desired optimization variables $\boldsymbol{D}_m = [\boldsymbol{d}_m^1, \ldots, \boldsymbol{d}_m^p]$, $\boldsymbol{T}_m = [\boldsymbol{t}_m^1, \ldots, \boldsymbol{t}_m^p]$ and $\boldsymbol{W}_m = [\boldsymbol{w}_m^1, \ldots, \boldsymbol{w}_m^p]$ as follows $\boldsymbol{D}_m = \{\boldsymbol{d}_m^j / \|\boldsymbol{d}_m^j\|_{\ell_2}\}$, $\boldsymbol{T}_m = \{\boldsymbol{t}_m^j / \sqrt{\alpha}\|\boldsymbol{d}_m^j\|_{\ell_2}\}$ and $\boldsymbol{W}_m = \{\boldsymbol{w}_m^j / \sqrt{\beta}\|\boldsymbol{d}_m^j\|_{\ell_2}\}$ for $1 \leq j \leq p$.

Given $\{\boldsymbol{D}_m, \boldsymbol{W}_m\}_{m=1}^M$ from the training step, for a test sample input $\boldsymbol{X}^t$ which is observed from all M modalities, $\boldsymbol{X}^t = \{\boldsymbol{x}_m^t\}_{m=1}^M$, we first compute its sparse representation $\boldsymbol{A}^t = \{\boldsymbol{\alpha}_m^t\}_{m=1}^M$ by solving $\sum_m \|\boldsymbol{x}_m^t - \boldsymbol{D}_m \boldsymbol{\alpha}_m^t\|_2^2 + \Omega(\boldsymbol{A}^t)$, where Ω is (4.8a). Then, we use the linear classifiers $\{\boldsymbol{W}_m\}_{m=1}^M$ to estimate a label vector $\hat{\boldsymbol{y}} = \sum_{m=1}^M \boldsymbol{W}_m \boldsymbol{\alpha}_m^t$. The label of $\boldsymbol{X}^t$ is the index corresponding to the largest element of $\hat{\boldsymbol{y}}$.

4.4 Multimodal: One-Against-All

Latent dictionary learning (LDL) [195] is a state-of-the-art supervised DL designed for single modality. The proposed, MWDL generalizes LDL to be able to fuse information from various sources at feature-level in order to make more discriminative sparse codes suitable for classification task.

Latent Dictionary Learning

In this section we briefly introduce LDL [195] that is similar to MWDL for the case of one single modality ($m = 1$). LDL models the relation between each atom of the dictionary with class labels using weight matrix $\boldsymbol{\Gamma} = [\boldsymbol{\gamma}_1, \boldsymbol{\gamma}_2, \ldots, \boldsymbol{\gamma}_C]$ in $\mathbf{R}^{p \times C}$. The c-th vector $\boldsymbol{\gamma}_c = [\boldsymbol{\Gamma}_{1c}, \ldots, \boldsymbol{\Gamma}_{pc}]^\top$ in $\mathbf{R}^p$ indicates how effective is each atom to represent c-th class. All elements of $\boldsymbol{\Gamma}$ are constrained to be equal or greater than zero: $\{\boldsymbol{\Gamma}_{kc}\}_{k=1}^{p} \geq 0$. When the k-th atom has no contribution to reconstruct the c-th class, then $\boldsymbol{\Gamma}_{kc} = 0$. Also, the sum of all weight elements for c-th class is $\sum_k \boldsymbol{\Gamma}_{kc} = \sigma$. This is to ensure that the dictionary has enough representation power for each class. The goal is to learn $\boldsymbol{D}$ and the $\boldsymbol{\Gamma}$, so that the data can be reconstructed in a sparse coding scheme as: $\boldsymbol{x}_{i,c} \approx \boldsymbol{D}\,\mathbf{diag}(\boldsymbol{\gamma}_c)\boldsymbol{\alpha}_{i,c}$:

$$
\begin{aligned}
&\underset{\boldsymbol{\Gamma}, \boldsymbol{D}, \boldsymbol{\alpha}}{\operatorname{argmin}} \sum_{i=1}^{N} \Bigg(\|\boldsymbol{x}_{i,c} - \boldsymbol{D}\,\mathbf{diag}(\boldsymbol{\gamma}_c)\boldsymbol{\alpha}_{i,c}\|_2^2 + \lambda_1 \|\boldsymbol{\alpha}_{i,c}\|_1 + \\
&\lambda_2 \|\boldsymbol{\alpha}_{i,c} - \mathbb{E}_i(\{\boldsymbol{\alpha}_{i,c}\})\|_2^2 + \frac{\mu}{2} \sum_{c=1}^{C} \sum_{l \neq c} \sum_{k=1}^{p} \sum_{j \neq k} \boldsymbol{\Gamma}_{jc} (\boldsymbol{d}_j^\top \boldsymbol{d}_k)^2 \boldsymbol{\Gamma}_{kl} \Bigg) \\
&\text{s.t.}\ \boldsymbol{\Gamma}_{kc} \geqslant 0 \text{ and } \sum_{k=1}^{p} \boldsymbol{\Gamma}_{kc} = \sigma,\ \forall c \in \{1, \ldots, C\}
\end{aligned}
\tag{4.11}
$$

4.4.1 Multimodal Weighted Dictionary Learning

Our intention is to generalize LDL with efficient feature-fusion algorithm so that it can achieve better classification performance in the presence of multimodal input data. Here, the sample i from c-th class is multimodal and observed from M features: $\boldsymbol{X}_{i,c} = \{\boldsymbol{x}_{i,c}^m\}_{m=1}^M$. The goal is to learn multimodal dictionaries that can reconstruct $\boldsymbol{X}_{i,c}$ with decomposition coefficients $\{\boldsymbol{\alpha}_{i,c}^m\}_{m=1}^M$ that are suitable for classification task. Our motivation is to exploit the group structure that is induced by the modality configuration of a multimodal data. However, in LDL, sparse coding is implemented (see Eq. (4.11)) using the standard ℓ_1-norm, which penalizes the cardinality of decomposition coefficients $\{\boldsymbol{\alpha}_{i,c}\}$. Particularly, this regularization treats each variable individually, and it is blind to potential group structure between different features of a sample. Joint sparsity priors are able to do fusion between

multiple features which makes them suitable to reconstruct samples originated from different sources [14, 156].

For each modality/feature, there is a weight matrix $\boldsymbol{\Gamma}^m = [\boldsymbol{\gamma}_1^m, \ldots, \boldsymbol{\gamma}_C^m]$ in $\mathbf{R}^{p \times C}$ and a dictionary $\boldsymbol{D}^m$ in $\mathbf{R}^{n_m \times p}$. Vector $\boldsymbol{\gamma}_c^m$ describes how much each of p atoms in $\boldsymbol{D}^m$ is used to reconstruct c-th class. Let us denote the weight vectors of c-th class from all modalities by $\boldsymbol{W}_c = [\boldsymbol{\gamma}_c^1, \ldots, \boldsymbol{\gamma}_c^M]$ in $\mathbf{R}^{p \times M}$ and the multimodal sparse representation of the data $\boldsymbol{X}_{i,c}$ as $\boldsymbol{A}_{i,c} = [\boldsymbol{\alpha}_{i,c}^1, \ldots, \boldsymbol{\alpha}_{i,c}^M]$. The proposed method is a bilevel optimization. The outer-level objective enforces similarity across columns of two matrices within each class: $\boldsymbol{A}_{i,c}$ and $\boldsymbol{W}_c$. The outer-level is subject to inner-level constraints such that the class-specific dictionary in each modality, $\boldsymbol{D}_c^m \approx \boldsymbol{D}^m \, \mathbf{diag}(\boldsymbol{\gamma}_c^m)$ for all m in $\{1, \ldots, M\}$ is reconstructive while at the same time to be incoherent with the dictionary of other classes.

We propose to obtain simultaneously, the multimodal sparse representation $\boldsymbol{A}_{i,c} = [\boldsymbol{\alpha}_{i,c}^1, \ldots, \boldsymbol{\alpha}_{i,c}^M]$ and the set of dictionaries $\{\boldsymbol{D}^m, \boldsymbol{\Gamma}^m\}_{m=1}^M$, for all m in $\{1, \ldots, M\}$:

$$\begin{aligned} \operatorname{argmin} \sum_{m=1}^{M} \left(\frac{1}{2} \|\boldsymbol{x}_{i,c}^m - \boldsymbol{D}^m \, \mathbf{diag}(\boldsymbol{\gamma}_c^m) \boldsymbol{\alpha}_{i,c}^m\|_2^2 + \frac{\mu}{2} \sum_{l \neq c} \sum_{k=1}^{p} \sum_{j \neq k} \boldsymbol{\Gamma}_{jc}^m (\boldsymbol{d}_j^{m\top} \boldsymbol{d}_k^m)^2 \boldsymbol{\Gamma}_{kl}^m \right) + \\ \xi \Omega(\boldsymbol{A}_{i,c}) + \frac{\lambda}{2} \|\boldsymbol{A}_{i,c}\|_F^2 + \upsilon \Omega(\boldsymbol{W}_c) \\ \text{s.t.} \, \boldsymbol{\Gamma}_{jc}^m \geqslant 0 \text{ and } \sum_{j=1}^{p} \boldsymbol{\Gamma}_{jc}^m = \sigma, \; \forall c \in \{1, \ldots, C\} \end{aligned} \tag{4.12}$$

where the fusion between M different features of the sample $\{\boldsymbol{x}_{i,c}^m\}_{m=1}^M$ is enforced in the space of sparse codes using ℓ_{12} regularization, $\Omega(\boldsymbol{A}) = \sum_{r=1}^{p} \|\boldsymbol{A}_{r\rightarrow}\|_2$; where $\boldsymbol{A}_{r\rightarrow}$ is the r-th row of $\boldsymbol{A}$ and promotes a solution with sparse non-zero rows in $\boldsymbol{A}$. Applying joint sparse representation on the multimodal sparse codes, $\Omega(\boldsymbol{A})$ promotes all modalities to share the same sparsity pattern: if k-th atom, $\boldsymbol{d}_k^m$, is selected to reconstruct the input $\boldsymbol{x}_{i,c}^m$, then all modalities of k-th atom, $\{\boldsymbol{d}_k^1, \ldots, \boldsymbol{d}_k^M\}$ should contribute to reconstruct $\{\boldsymbol{x}_{i,c}^m\}_{m=1}^M$. In the same way, if $\boldsymbol{\gamma}_c^m$, the m-th column of $\boldsymbol{W}_c$ determines a certain subset of atoms in $\boldsymbol{D}^m$ to represent c-th class, other columns of $\boldsymbol{W}_c$ should also have the same opinion.

Optimization

The optimization problem (4.12) has the product of three optimization variables $\boldsymbol{D}\,\mathbf{diag}(\boldsymbol{\gamma}_c)\boldsymbol{\alpha}_{i,c}$; which implies that this problem is not joint convex in the space of variables. However, when two of the three optimization variables are fixed, the problem (4.12) is convex with respect to the third variable [108]. Hence, the problem (4.12) is solved by splitting to three sub-problems: 1. given $\{\boldsymbol{\Gamma}^m\}_{m=1}^M$ and dictionaries $\{\boldsymbol{D}^m\}_{m=1}^M$, estimate the multimodal sparse codes $\{\boldsymbol{\alpha}_i^m\}_{m=1}^M$ for all i in $\{1,\ldots,N\}$; 2. given $\boldsymbol{\Gamma}^m$ and sparse codes $\{\boldsymbol{\alpha}_i^m\}_{i=1}^N$, update the corresponding dictionary of m-th modality $\boldsymbol{D}^m$; 3. given $\{\boldsymbol{\alpha}_i^m\}$ and $\boldsymbol{D}^m$, update $\boldsymbol{\Gamma}^m$. Step 2 is done for each m in $\{1,\ldots,M\}$ and c in $\{1,\ldots,C\}$, separately.

Step 1: Find Multimodal Sparse Codes

In this section, we fix $\{\boldsymbol{\Gamma}^m\}_{m=1}^M$ and $\{\boldsymbol{D}^m\}_{m=1}^M$ and treat them as data for the problem (4.12). We initialize the multimodal dictionaries $\{\boldsymbol{D}^m\}_{m=1}^M$ by training samples of all classes same as [77, 104]. The problem (4.12) is converted to (4.13) to find an optimal $\boldsymbol{A}_{i,c}^\star = [\boldsymbol{\alpha}_{i,c}^1,\ldots,\boldsymbol{\alpha}_{i,c}^M]$ in $\mathbf{R}^{p\times M}$ for all i in $\{1,\ldots,N\}$:

$$\underset{\boldsymbol{A}_{i,c}}{\operatorname{argmin}} \sum_{m=1}^{M} \frac{1}{2}\|\boldsymbol{x}_{i,c}^m - \boldsymbol{D}^m\,\mathbf{diag}(\boldsymbol{\gamma}_c^m)\boldsymbol{\alpha}_{i,c}^m\|_2^2 + \xi\Omega(\boldsymbol{A}_{i,c}) + \frac{\lambda}{2}\|\boldsymbol{A}_{i,c}\|_F^2 \tag{4.13}$$

where $\|.\|_F$ is the Frobenius norm. To obtain optimal multimodal sparse codes $\boldsymbol{A}_{i,c}^\star$ of i-th multimodal sample $\{\boldsymbol{x}_i^m\}_{m=1}^M$, we solve the optimization problem (4.13) for a limited number of iterations using the alternating direction method of multipliers (ADMM) [136]. This part is done using the solution in Sec. (3.5.1).

Step 2: Multimodal Dictionary Learning

In Sec. 4.4.1, we obtain multimodal sparse coefficients of i-th sample, $\boldsymbol{A}_{i,c}^\star = [\boldsymbol{\alpha}_{i,c}^{1\star},\ldots,\boldsymbol{\alpha}_{i,c}^{M\star}]$ by solving the optimization problem (4.12) given the set of dictionaries $\{\boldsymbol{D}^m\}_{m=1}^M$. In this section, the obtained coefficients $\{\boldsymbol{A}_i^\star\}_{i=1}^N$ are used to update the dictionaries. The dictionary $\boldsymbol{D}^m = [\boldsymbol{d}_1^m\ldots,\boldsymbol{d}_p^m]$ is updated using the sparse representation of all samples from

m-th modality: $[\boldsymbol{\alpha}_1^m, \ldots, \boldsymbol{\alpha}_N^m]$. Since in this step dictionary of each modality is obtained independent of other modalities, we drop superscript m. We solve following optimization problem with dictionary as variable using Iterative Projection Method [149].

$$\min_{\boldsymbol{D}} \sum_i \frac{1}{2}\|\boldsymbol{x}_{i,c} - \boldsymbol{D}\boldsymbol{\beta}_i\|_2^2 + \frac{\mu}{2}\sum_{l\neq c}\sum_{k=1}^{p}\sum_{j\neq k}\boldsymbol{\Gamma}_{jc}(\boldsymbol{d}_j^\top \boldsymbol{d}_k)^2\boldsymbol{\Gamma}_{kl} \tag{4.14}$$

where $\boldsymbol{\beta}_i = \mathbf{diag}(\boldsymbol{\gamma}_c)\boldsymbol{\alpha}_{i,c}$ in $\mathbf{R}^p$. Let us define $\boldsymbol{B} = \boldsymbol{\beta}_i\boldsymbol{\beta}_i^\top$ and $\boldsymbol{G} = \boldsymbol{x}_i\boldsymbol{\beta}_i^\top$. Also, note that the second part of the Eq. (4.14) for the k-th atom would be simpilified to $\boldsymbol{S} \triangleq \mu\sum_{j\neq k}(\boldsymbol{d}_j\boldsymbol{d}_j^\top)\sum_{l\neq c}\boldsymbol{\Gamma}_{jc}\boldsymbol{\Gamma}_{kl}$. We solve problem (4.14) to update the k-th atom, $\boldsymbol{d}_k$ following [104]:

$$\boldsymbol{d}_k \leftarrow \boldsymbol{d}_k + (\boldsymbol{B}_{kk}\boldsymbol{I} + \mathbf{diag}\,(\boldsymbol{S}))^{-1}(\boldsymbol{G}^{k\downarrow} - \boldsymbol{D}\boldsymbol{B}^{k\downarrow} - \boldsymbol{S}\boldsymbol{D}^{k\downarrow}) \tag{4.15}$$

where $\boldsymbol{B}_{kk}$ is the k-th element in the diagonal of the $\boldsymbol{B}$. Finally, the updated atom $\boldsymbol{d}_k$ is projected orthogonal to the unit-norm ball. We do the same to update each atom from any feature, $\{\boldsymbol{d}_k^m\}_{k=1}^p$ and m in $\{1, \ldots, M\}$.

Step 3: Weight Estimation

Given $\{\boldsymbol{D}^m\}_{m=1}^M$ and $\{\boldsymbol{x}_{i,c}^m\}$, the Eq. (4.12) is converted to a constrained quadratic programming and solved for each class-specific weight matrix from all modalities $\boldsymbol{W}_c = [\boldsymbol{\gamma}_c^1, \ldots, \boldsymbol{\gamma}_c^M]$ in $\mathbf{R}^{p\times M}$ separately.

$$\begin{aligned}\underset{\boldsymbol{W}_c}{\operatorname{argmin}} \sum_i \frac{1}{2}\|\boldsymbol{x}_{i,c}^m - \boldsymbol{D}^m\,\mathbf{diag}(\boldsymbol{\gamma}_c^m)\boldsymbol{\alpha}_{i,c}^m\|_2^2 + \mu\sum_{k=1}^{p}\boldsymbol{\Gamma}_{kc}^m\sum_{j\neq k}(\boldsymbol{d}_j^{m\top}\boldsymbol{d}_k^m)^2\sum_{l\neq c}\boldsymbol{\Gamma}_{jl}^m + \upsilon\Omega(\boldsymbol{W}_c)\\ \text{s.t.}\,\boldsymbol{\Gamma}_{kc} \geq 0\;\forall k \in \{1, \cdots, p\} \text{ and } \sum_{k=1}^{p}\boldsymbol{\Gamma}_{kc}^m = \sigma\end{aligned} \tag{4.16}$$

with $\boldsymbol{\Gamma}_{kc} \geq 0$. Similar to problem (4.13), the optimization problem (4.16) is made of smooth and non-smooth parts ($\Omega(\boldsymbol{W}_c)$); hence the solution methodology is similar to Sec. (4.4.1): the proximal problem (3.13a) over smooth part of (4.16) is solved similar to [195]. The proximal

(3.13b) over non-smooth part is solved same as Eq. (3.16), which enforces row-sparsity for the variable $\boldsymbol{W}_c$.

Classification Approach

Each test sample $\boldsymbol{X}_t$ is observed from the same set of M features, $\boldsymbol{X}_t = \{\boldsymbol{x}_t^m\}_{m=1}^M$. We use the learned dictionaries and the weight matrices $\{\boldsymbol{D}^m, \boldsymbol{\Gamma}^m\}_{m=1}^M$ in training phase to extract sparse codes of the test sample, $\{\boldsymbol{\alpha}_t^m\}_{m=1}^M$, as elaborated in Sec. (4.4.1). The query is assigned to the class with minimum summation of reconstruction error of all features, $\mathcal{E}_t = \sum_{m=1}^M \|\boldsymbol{x}_t^m - \boldsymbol{D}^m \operatorname{\mathbf{diag}}(\boldsymbol{\gamma}_c^m)\boldsymbol{\alpha}_t^m\|_2^2$.

4.5 Implicitly Defined Dictionary

Learning a supervised dictionary in all-vs-all scheme results in similar atoms which are shared among multiple classes because the atoms in dictionary are not required to be uncorrelated. We propose a multimodal dictionary learning method that minimizes correlation between atoms. In this section, we elaborate our bilevel sparse coding model to learn multimodal task-driven dictionaries with uncorrelated elements that is able to formulate feature-fusion as a mixed-norm structure sparsity. The method is designed to obtain dictionaries via coupling across different features of a signal in space of sparse codes.

The outer-level objective is designed to find optimization variables jointly; whose variables are the dictionary, classifier and transformation of modality m, $\{\boldsymbol{D}_m^*, \boldsymbol{W}_m^*, \boldsymbol{T}_m\}_{m\in[\![1;M]\!]}$. Multimodal sparse coefficients, $\boldsymbol{A}^{i*} \in \mathbf{R}^{p\times M}$ is parameter for outer-level, but variable for the inner-level objective. Assuming i-th training sample $\boldsymbol{X}^i$ from the c-th class, a binary vector $\boldsymbol{q}^i \in \mathbf{R}^p$ is defined that is zero everywhere except at the indices of atoms which belong to the c-th class. This so called "label consistency constraint" is applied using $\{\boldsymbol{q}_i\}_{i=1}^N$ so that the sample from c-th class is represented using the same subset of dictionary items

associated with class c. The outer-level is defined as

$$\underset{\{\boldsymbol{D}_m, \boldsymbol{W}_m\}}{\operatorname{argmin}} \; f(\{\boldsymbol{D}_m, \boldsymbol{W}_m\}) + \frac{\nu_1}{2} \sum_{m=1}^{M} \|\boldsymbol{W}_m\|_F^2 \tag{4.17a}$$

$$f(\{\boldsymbol{D}_m, \boldsymbol{W}_m\}) \triangleq \mathbb{E}_{\boldsymbol{y},\boldsymbol{x}} \left[\sum_{m=1}^{M} \mathcal{L}_s(\boldsymbol{y}, \boldsymbol{W}_m, \boldsymbol{\alpha}_m^{i*}) \right] \tag{4.17b}$$

$$\mathcal{L}_s(\boldsymbol{y}^i, \boldsymbol{W}_m, \boldsymbol{\alpha}_m^{i*}) \triangleq \|\boldsymbol{y}^i - \boldsymbol{W}_m \boldsymbol{\alpha}_m^i\|_2^2 \tag{4.17c}$$

where $\mathcal{L}_s$ is the supervised loss function for i-th sample from modality m, $\boldsymbol{W}_m$ is the parameters of a linear classifier with the regularization parameter ν_1. The loss function in (4.17a) is defined as the expectation over summation of cost from each modality in (4.17b).

The inner-level objective with i-th multimodal input $\boldsymbol{X}^i = \{\boldsymbol{x}_m^i\}_{m=1}^M$, multimodal dictionaries $\{\boldsymbol{D}_m\}_{m=1}^M$ as parameters is designed to find multimodal decomposition coefficients $\boldsymbol{A}^{i\star}$

$$\boldsymbol{A}^*\big(\{\boldsymbol{x}_m^i, \boldsymbol{D}_m\}_{m=1}^M\big) \triangleq \underset{\boldsymbol{A} \in \mathbf{R}^{p \times M}}{\operatorname{argmin}} \frac{1}{M} \sum_{m=1}^{M} \frac{1}{2} \|\boldsymbol{x}_m^i - \boldsymbol{D}_m \boldsymbol{\alpha}_m^i\|_2^2 + \lambda_1 \Upsilon(\boldsymbol{A}) + \frac{\lambda_2}{2} \|\boldsymbol{A}\|_F^2 \tag{4.18a}$$

$$\Upsilon(\boldsymbol{A}) \triangleq \sum_{r=1}^{p} \sum_{g \in \mathcal{G}} \left(\sum_{m \in g} (\omega_m^{(g)})^2 |\boldsymbol{A}_{rm}|^2 \right)^{\frac{1}{2}} = \sum_{r=1}^{p} \sum_{g \in \mathcal{G}} \|\boldsymbol{\omega}^{(g)} \circ \boldsymbol{A}_{r\rightarrow}\|_2 \tag{4.18b}$$

where $\boldsymbol{\alpha}_m^{i\star}$ is the m-th column of the multimodal sparse codes $\boldsymbol{A}^\star(\{\boldsymbol{x}_m, \boldsymbol{D}_m\})$. The Frobenius norm $\|.\|_F$ will be proved to find a unique solution to the inner-level optimization (4.18a). $(\boldsymbol{\omega}^{(g)})_{g \in \mathcal{G}}$ is a $|\mathcal{G}|$-tuple of M dimensional vectors that are zero for indices of modalities that are not member of $g \in \mathcal{G}$; *i.e.* $\omega_m^{(g)} > 0$ if $m \in g$ and is zero otherwise.

The desired pattern of nonzero elements for the r-th row of $\boldsymbol{A}_{r\rightarrow}$ is a given tree $\mathcal{G}$ with $|\mathcal{G}|$ nodes index by g in $\{1, \ldots, |\mathcal{G}|\}$. Υ penalizes sparse representations of groups of features that are embedded in a tree-shaped hierarchy. Assume $\mathcal{G}$ a subset of power set of $[\![1; M]\!]$, *i.e.* $\mathcal{G} \subseteq [\![1; M]\!]$ with the condition that the $\mathcal{G}$ span the set of modalities, *i.e.* $\cup_{g \in \mathcal{G}} = [\![1; M]\!]$. Equivalently, the solution is sparse at the group level, in the sense that coefficients within a group are usually zero or nonzero together. The regularization parameter $\lambda_1 \geq 0$ is used

to adjust the tradeoff between minimizing the loss and finding a solution which is sparse at the group level.

This tree has M leaves corresponding to the M modality-based sparse codes: $\boldsymbol{\alpha}_1^i, \ldots, \boldsymbol{\alpha}_M^i$ and some internal nodes representing different grouping between the leaves (modalities). Each internal node encodes a possible grouping between leaves of the subtree ($\{\boldsymbol{\alpha}_m^i\}_{m=1}^M$) which internal node is their root [87, 14]. Let us define the set of indices corresponding to the parents of the leaf (feature) m in $\mathcal{G}$ as parents(m). Then, the tree-structure sparsity Υ enforces the following effect: $\boldsymbol{\alpha}^m \neq 0 \Rightarrow [\boldsymbol{\alpha}^g \neq 0, \forall g \in \text{parents}(m)]$. In other words, the structure of $\mathcal{G}$ may be expresses as following: the codes of any multimodal signal $\{\boldsymbol{x}_m^i\}_{m=1}^M$ can exploit a dictionary atom from m-th modality only if the parents of that modality (parents(m)) in the tree $\mathcal{G}$ are themselves part of the decomposition.

Following [72], a tree-structure $\mathcal{G}$ associated with grouping M modalities is defined as $\mathcal{G} = \{\mathcal{G}_\upsilon | \upsilon \in \mathbb{V}\}$ that has $|\mathbb{V}|$ nodes $\mathcal{G}_\upsilon$ where $\cup_\upsilon \mathcal{G}_\upsilon = \{1, \ldots, M\}$. Each node $\mathcal{G}_\upsilon$ represents a member of the $2^{\mathcal{M}}$ set of all possible grouping structures. Also, for each pair $\mathcal{G}_i$ and $\mathcal{G}_j$ we have $(\mathcal{G}_i \cap \mathcal{G}_j \neq \emptyset) \Rightarrow ((\mathcal{G}_i \subseteq \mathcal{G}_j) \vee (\mathcal{G}_j \subseteq \mathcal{G}_i))$. Either prior information or hierarchical agglomerative clustering algorithm can be used to obtain the tree structure [87, 14].

Following, we present applications of our hierarchical task-driven dictionary learning formulations for binary and multi-class classification. Our approach is not limited to these examples.

Binary Classification.

In this setting, the set of labels y is a member of the set $\{-1, +1\}$. Logistic regression is used for supervised loss function $\mathcal{L}_s = \log(1 + \exp(-y\boldsymbol{w}_m^\top \boldsymbol{\alpha}_m^\star(\boldsymbol{x}_m, \boldsymbol{D}_m)))$. Any other twice differentiable loss function can be used, for instance, the square loss is also a reasonable choice. Given a multimodal input data $\{\boldsymbol{x}_m^i\}_{m=1}^M$, we want to learn the parameters $\boldsymbol{w}_m \in \mathbf{R}^p$ of a linear model to predict $\boldsymbol{y}$ in $\mathcal{Y}$, using the sparse representation $\boldsymbol{\alpha}_m^{i\star}$ as features, and

jointly optimize $\boldsymbol{D}_m$ and $\boldsymbol{w}_m$

$$\underset{\boldsymbol{w}_m, \boldsymbol{D}_m}{\operatorname{argmin}} \mathbb{E}_{y,\boldsymbol{x}} \left[\sum_{m=1}^{M} \log \left(1 + e^{-y \boldsymbol{w}_m^\top \boldsymbol{\alpha}_m^\star}\right) \right] + \frac{\nu_1}{2} \sum_{m=1}^{M} \|\boldsymbol{w}_m\|_2^2 \tag{4.19}$$

Equation (4.19) provides the optimal variables $\{\boldsymbol{D}_m, \boldsymbol{w}_m\}$. Then a multimodal query $\{\boldsymbol{x}_m\}$ is assigned to sign of $\sum_{m=1}^{M} \boldsymbol{w}_m^\top \boldsymbol{\alpha}_m^\star$. For simplicity the intercept is omitted, however, it can be easily added. Note that, only outer-level optimization Eq. (4.17b) needs to be changed.

Multi-class Classification.

Multi-class classification can be obtained by extending binary classification with two labels to a set of labels in $\{1, 2, \ldots, C\}$ with $C > 2$. The discriminative power of dictionary in supervised methods depend on the relation between the label of atoms and class labels in the data.

This extension can be done in two schemes: one-against-all and all-against-all. In one-against-all a class-specific dictionary is trained on its corresponding data. Then all the class-specific dictionaries are concatenated horizontally to make the final dictionary [185]. The issue in this approach is that each sub-dictionary obtained independent of other classes. Often, classes are not completely independent from each other, and they have some features in common, hence it is high probable that they have similar atoms which lead to similar sparse representations for samples that belong to the different classes which degrades classification accuracy. In all-against-all dictionary learning, a single dictionary is shared between all classes. The shared dictionary usually has a less number of atoms, which make the coding in the testing phase efficiently, but, there is no guarantee that each atom is representing a certain class. If an atom is adapted to multiple classes, the generated codes of that atom are not discriminative enough. The idea of estimating dictionary and classifier jointly in the all-against-all scheme while they are connected via sparse codes proposed in [104]. In practice, one-against-all DL methods lead to large dictionaries. In all-against-all setting, the dictionary is shared between classes. This results in a dictionary with fewer atoms

but the discriminative power suffers from the fact that each atom may represent multiple classes [77, 104, 128].

Multi-class classification in all-against-all scheme can be modeled using the softmax regression loss function

$$\mathcal{L}_s(y, \boldsymbol{W}, \boldsymbol{\alpha}^\star) = \sum_{c=1}^{C} 1_{\{y=c\}} \log \left(\frac{e^{\boldsymbol{w}_c^\top \boldsymbol{\alpha}^\star}}{\sum_{c=1}^{C} e^{\boldsymbol{w}_c^\top \boldsymbol{\alpha}^\star}} \right) \tag{4.20}$$

where $1_{\{y=c\}}$ is the indicator function for class c, and $\boldsymbol{W} = [\boldsymbol{w}_1, \ldots, \boldsymbol{w}_C]$ in $\mathbf{R}C \times p$. Equation (4.20) obtains $\{\boldsymbol{D}_m, \boldsymbol{W}_m\}$. then, a new query $\{\boldsymbol{x}_m\}$ is classified as

$$\underset{c \in \{1,\ldots,C\}}{\operatorname{argmax}} \sum_{m=1}^{M} \left(\frac{e^{\boldsymbol{w}_{m,c}^\top \boldsymbol{\alpha}_m^\star}}{\sum_{c=1}^{C} e^{\boldsymbol{w}_{m,c}^\top \boldsymbol{\alpha}_m^\star}} \right) \tag{4.21}$$

4.5.1 Extension

We now extend the proposed algorithm with a more discriminative structure on the sparse codes. We enforce each atom to represent a particular class in all modalities; which enables the multimodal dictionaries to be more discriminative.

We change the outer-level objective in Eq. 4.17b. Assuming i-th training sample $\boldsymbol{X}^i$ from the c-th class, a binary vector $\boldsymbol{q}^i \in \mathbf{R}^p$ is defined that is zero everywhere except at the indices of atoms which belong to the c-th class. This so called "label consistency constraint" is applied using $\{\boldsymbol{q}_i\}_{i=1}^N$ so that the sample from c-th class is represented using the same subset of dictionary items associated with class c. The outer-level would be

$$\begin{aligned}
&\underset{\{\boldsymbol{D}_m, \boldsymbol{W}_m, \boldsymbol{T}_m\}}{\operatorname{argmin}} \quad f(\{\boldsymbol{D}_m, \boldsymbol{W}_m, \boldsymbol{T}_m\}) + \frac{\nu_1}{2} \sum_{m=1}^{M} \|\boldsymbol{W}_m\|_F^2 + \frac{\nu_2}{2} \|\boldsymbol{T}_m\|_F^2 \\
&f(\{\boldsymbol{D}_m, \boldsymbol{W}_m, \boldsymbol{T}_m\}) \triangleq \mathbb{E}_{\boldsymbol{y},\boldsymbol{x}} \left[\sum_{m=1}^{M} \mathcal{L}_s(\boldsymbol{y}, \boldsymbol{W}_m, \boldsymbol{T}_m, \boldsymbol{\alpha}_m^{i*}) \right] \qquad (4.22a) \\
&\mathcal{L}_s(\boldsymbol{y}^i, \boldsymbol{W}_m, \boldsymbol{T}_m, \boldsymbol{\alpha}_m^{i*}) \triangleq \mu \|\boldsymbol{q}^i - \boldsymbol{T}_m \boldsymbol{\alpha}_m^i\|_2 + (1-\mu) \|\boldsymbol{y}^i - \boldsymbol{W}_m \boldsymbol{\alpha}_m^i\|_2
\end{aligned}$$

where $\mathcal{L}_s$ is the supervised loss function for i-th sample from modality m and $\boldsymbol{T}_m$ is a linear transformation matrix, $\boldsymbol{W}_m$ is the parameters of a linear classifier and ν_1 and ν_2 are regularization paramters. The label consistency $\|\boldsymbol{q}^i - \boldsymbol{T}_m \boldsymbol{\alpha}_m^i\|_{\ell_2}^2$ regularization enforces the linear transformed version of original sparse codes $\boldsymbol{T}_m \boldsymbol{x}_m^i$ to be most discriminative in the $\mathbf{R}^p$ space. The inner-level is the same as Eq. 4.18.

4.6 Optimization

The main difficulty is to calculate the partial differential of cost function f in optimization problem (4.17) with respect to dictionary $\boldsymbol{D}_m$. Because dictionary is not explicitly defined in optimization problem (4.17); but, it is defined implicitly in inner-level problem (4.18). The other challenge is that the optimization problem (4.17) is not differentiable with respect to $\boldsymbol{A}^*(\{\boldsymbol{x}_m^i, \boldsymbol{D}_m\}_{m=1}^M)$. However, we will show that the function $f(\{\boldsymbol{D}_m, \boldsymbol{W}_m, \boldsymbol{T}_m\})$ defined in (4.17b) is differentiable on space of $\mathcal{D}_1 \times \cdots \times \mathcal{D}_M \times \mathcal{W}_1 \times \cdots \times \mathcal{W}_M \times \mathcal{T}_1 \times \cdots \times \mathcal{T}_M$; hence its gradient can be computed. We use chain rule to compute the step direction of gradient descent algorithm for optimization variable $\boldsymbol{D}_m$. We use $\boldsymbol{A}^*$ as shorthand for $\boldsymbol{A}^*(\{\boldsymbol{x}_m^i, \boldsymbol{D}_m\}_{m=1}^M)$.

Assumptions

The optimal condition of Eq. 4.18b is one way to show that $\boldsymbol{A}^*$ is differentiable everywhere except rows with all zero elements. Also, the proposed optimization method belongs to the class of online methods based on stochastic approximations and uses a mini-batch of training set on each iteration to update the variables and sequentially minimizes a quadratic local surrogate of the expected cost. In an attempt to prove the differentiability of function f, one can generalize required assumptions for the case with only single feature in [104] and come up with following

(A) The joint probability density $p(\boldsymbol{X}, \boldsymbol{y})$ of the multimodal data in image and video processing and its corresponding variable $\boldsymbol{y}, (\boldsymbol{X} = \{\boldsymbol{x}_m\}_{m=1}^M)$ is compact. This lies in the fact that sensors in the image and video data acquisition generate bounded values.

(B) For classification task of finite number of classes, $c \in \{1, \dots, C\}$, for any label $\boldsymbol{y}$, the distribution $p(\boldsymbol{y}, .)$ is continuous and the supervised loss function $\mathcal{L}_s(\boldsymbol{y}, .)$ is twice continuously differentiable.

We propose to solve problem (4.17) using projected first-order stochastic gradient descent algorithm. But first we need to define *active set* to state the main proposition

Definition 4.1. *(Active set:) The active set Λ with the cardinality $\pi = |\Lambda|$ of the multimodal sparse representation $\boldsymbol{A}^* \in \mathbf{R}^{p \times M}$ with p rows of $\{\boldsymbol{A}^\star_{r\rightarrow} \in \mathbf{R}^M\}_{r \in [\![1;p]\!]}$ is defined as*

$$\Lambda = \left\{ r \in \{1, \dots, p\} : \sum_{g \in \mathcal{G}} \|\boldsymbol{A}^{(g)}_{r\rightarrow}\|_2 \neq 0 \right\} \tag{4.23}$$

where $\boldsymbol{A}^{(g)}_{r\rightarrow}$ is the vector of size M whose coordinates are equal to those of $\boldsymbol{A}_{j\rightarrow}$ for indices in the set g, and 0 otherwise. For the rows that belong to the active set, $\boldsymbol{A}^*_\Lambda$ we calculate the partial derivative of the problem (4.18b) with respect to members of active set. Consider $\mathcal{A} \in \mathbf{R}^{\pi \times M}$ to only include π rows of $\boldsymbol{A}$ that are members of active set: $\mathcal{A} = \{\boldsymbol{A}_{i\rightarrow}\}_{i \in \Lambda}$. We show the i-th row of $\mathcal{A}$ as $\mathcal{A}_{i\rightarrow}$, the j-th column as $\mathcal{A}^j$ and the element of i-th row and j-th column with $\mathcal{A}(i, j)$.

Proposition 4.2. *(Differentiability and gradients of f):*
Assume $\lambda_2 > 0$ in Eq. 4.18a and the assumptions (A) and (B) holds. Let us denote $\tilde{\boldsymbol{d}}^i_m \in \mathbf{R}^n$ as extended version of atom $\boldsymbol{d}^i_m \in \mathbf{R}^{n_m}$ with zeros, where $n = \sum_{m=1}^M n_m$. Then, for the i-th atom that i is a member of active set, we concatenate horizontally the atom from all features as $\tilde{\boldsymbol{\Delta}}_i \in \mathbf{R}^{n \times M} = [\tilde{\boldsymbol{d}}^i_1, \dots, \tilde{\boldsymbol{d}}^i_M]$. Furthermore, a block diagonal matrix $\mathfrak{D}$ has cross-correlation of each active atom from all features in its diagonal

$$\mathfrak{D} \in \mathbf{R}^{\pi M \times \pi M} = \begin{bmatrix} \tilde{\boldsymbol{\Delta}}_1{}^\top \tilde{\boldsymbol{\Delta}}_1 & \dots & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \tilde{\boldsymbol{\Delta}}_\pi{}^\top \tilde{\boldsymbol{\Delta}}_\pi \end{bmatrix} \tag{4.24}$$

Also, for each active atom i in Λ, we have a square matrix $\boldsymbol{\Xi}_i$ in $\mathbf{R}^{M\times M}$ as $\boldsymbol{\Xi}_i = [\boldsymbol{\xi}_i^1, \ldots, \boldsymbol{\xi}_i^M]$; whose m-th column is denoted as $\boldsymbol{\xi}_i^m$ in $\mathbf{R}^M$ is defined as

$$\boldsymbol{\xi}_i^m = \frac{1}{\|\mathcal{A}_{i\rightarrow}\|_{\ell_2}}\left(\boldsymbol{I}^m - \frac{\mathcal{A}_{i\rightarrow}^\top \mathcal{A}(i,m)}{\left(\|\mathcal{A}_{i\rightarrow}\|_{\ell_2}\right)^2}\right) \tag{4.25}$$

where $\boldsymbol{I}^m$ is the m-th column of identity matrix $\boldsymbol{I}$. We use $\{\boldsymbol{\Xi}_i\}_{i\in\Lambda}$ to make a block diagonal matrix $\mathfrak{X}$ with $\boldsymbol{\Xi}_i$ in its i-th diagonal element

$$\mathfrak{X} \in \mathbf{R}^{\pi M\times \pi M} = \begin{bmatrix} \boldsymbol{\Xi}_1 & \ldots & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \boldsymbol{\Xi}_\pi \end{bmatrix} \tag{4.26}$$

Given $\mathfrak{D}$ and $\mathfrak{X}$, the square matrix $\mathcal{O}$ in $\mathbf{R}^{\pi M\times \pi M}$ with πM columns $\{\mathcal{O}^i\}_{i=1}^{\pi M}$ in $\mathbf{R}^{\pi M}$ is defined as

$$\mathcal{O} = \left(\mathfrak{D}^\intercal\mathfrak{D} + \lambda_1\mathfrak{X} + \lambda_2\boldsymbol{I}\right)^{-1} \tag{4.27}$$

where $\boldsymbol{I}$ is the identity matrix. Note that the columns of $\mathcal{O}$ that corresponds to dictionary of the m-th modality is: $\mathfrak{m} = \{m, m+M, m+2M, \ldots, m+(\pi-1)M\}$. Now, we denote matrix $\mathcal{Z}_m$ made by horizontally concatenating columns of $\mathcal{O}$ with indices of $\mathfrak{m}$ to make $\mathcal{Z}_m \in \mathbf{R}^{\pi M\times \pi} = \{\mathcal{O}^i\}_{i\in\mathfrak{m}}$.

The loss function $\mathcal{L}_s$ in Eq. 4.17c can be divided to two parts: $\mathcal{L}_s = \mu\mathcal{L}_s^1 + (1-\mu)\mathcal{L}_s^2$. We show $\boldsymbol{\beta}^{1*}$ and $\boldsymbol{\beta}^{2*}$ corresponding to $\mathcal{L}_s^1$ and $\mathcal{L}_s^2$, respectively. With k in the set $\{1,2\}$, we have $\boldsymbol{\beta}^{k*} \in \mathbf{R}^{1\times p}$ where $\boldsymbol{\beta}_{\Lambda^c}^{k*} = 0$ and $\boldsymbol{\beta}_{\Lambda}^{k*} = \text{vec}\left(\frac{\partial\mathcal{L}_s^k}{\partial\boldsymbol{A}^\intercal}\right)^\top\mathcal{Z}_m$ and $\text{vec}(.)$ as vectorization operator. The vector $\boldsymbol{\beta}^{k*}$ in $\mathbf{R}^{1\times p}$ contain values of $\text{vec}\left(\frac{\partial\mathcal{L}_s^k}{\partial\boldsymbol{A}^\intercal}\right)^\top\mathcal{Z}_m$ corresponding to the set Λ and zero, otherwise. Using the chain rule we obtain

$$\frac{\partial\mathcal{L}_s^k}{\partial\boldsymbol{D}_m} = E\left[\left(\boldsymbol{x}_m - \boldsymbol{D}_m\boldsymbol{A}^m\right)\boldsymbol{\beta}_m^{k*} - \boldsymbol{D}_m\boldsymbol{\beta}_m^{k*\top}\boldsymbol{A}^{m\top}\right] \tag{4.28}$$

and Finally, we have

$$\nabla_{\boldsymbol{D}_m} f = \mu\mathbb{E}\Big[(\boldsymbol{x}_m - \boldsymbol{D}_m\boldsymbol{A}^{m*})\boldsymbol{\beta}_m^{1^*\top} - \boldsymbol{D}_m\boldsymbol{\beta}_m^{1^*}\boldsymbol{A}^{m*\top}\Big] + (1-\mu)\mathbb{E}\Big[(\boldsymbol{x}_m - \boldsymbol{D}_m\boldsymbol{A}^{m^*})\boldsymbol{\beta}_m^{2^*\top} - \boldsymbol{D}_m\boldsymbol{\beta}_m^{2^*}\boldsymbol{A}^{m*\top}\Big] \tag{4.29}$$

The gradients of problem (4.17b) with respect to $\boldsymbol{T}_m$ and $\boldsymbol{W}_m$ is obtained by

$$\nabla_{\boldsymbol{T}_m} f = \mathbb{E}\big[\mu(\boldsymbol{T}_m\boldsymbol{\alpha}_m^{i*} - \boldsymbol{q}^i) + \nu_2\boldsymbol{T}_m\big] \tag{4.30a}$$

$$\nabla_{\boldsymbol{W}_m} f = \mathbb{E}\big[(1-\mu)(\boldsymbol{W}_m\boldsymbol{\alpha}_m^{i*} - \boldsymbol{y}^i) + \nu_1\boldsymbol{W}_m\big] \tag{4.30b}$$

The details of this proposition is given in the Appendix. Algorithm 4 describes the stochastic gradient descent algorithm to obtain optimal multimodal dictionaries and classifiers, $\{\boldsymbol{D}_m^*, \boldsymbol{W}_m^*, \boldsymbol{T}_m^*\}_{m=1}^M$.

4.6.1 Algorithm

Typically the optimization problems with the form of (4.29) and (4.30) are minimized using stochastic gradient descent algorithms. It has been shown that these methods can converge to a stationary point even for non-convex optimization problems assuming three-times differentiability which is slightly stricter than the assumptions in this dissertation [19, 18]. To speed-up the dictionary learning method, instead of accessing the whole training set at each iteration in order to minimize a cost function, inspired by [104], we chose a small batch of training set in each iteration to update the optimization variables of the problem (4.17).

With assumption (A) hold, the training set is made of i.i.d. samples of a distribution $p(\boldsymbol{y}, \{\boldsymbol{x}_m^i\}_{m=1}^M)$. As in stochastic gradient descent, in each iteration a mini-batch is drawn from the probability distribution $p(\boldsymbol{y}, \{\boldsymbol{x}_m^i\}_{m=1}^M))$. The algorithm alternates between estimation of the multimodal decomposition coefficients of each sample in the current mini-batch $\boldsymbol{A}^* = [\boldsymbol{\alpha}_1^{i*}, \ldots, \boldsymbol{\alpha}_M^{i*}]$ of the i-th input $\boldsymbol{X}^i = \{\boldsymbol{x}_m^i\}_{m=1}^M$ over the dictionaries $\{\boldsymbol{D}_m\}_{m=1}^M$

Algorithm 4 Stochastic gradient descent algorithm for multimodal task-driven dictionary learning.

Input: $\lambda_1, \lambda_2, \nu_1, \nu_2, \mu$ (regularization parameters), T (number of iterations), ρ, t_0 (learning-rate parameters), $\{\boldsymbol{D}_m \in \mathcal{D}_m\}_{m=1}^M$ (initial multimodal dictionaries), $\{\boldsymbol{W}_m \in \mathcal{W}_m\}_{m=1}^M$ (initial classifier parameters), $\{\boldsymbol{T}_m \in \mathcal{T}_m\}_{m=1}^M$ (initial linear transformations).

for $t = 1, \ldots, T$ **do**

Draw $\{\boldsymbol{y}_t, (\boldsymbol{X}_t = \{\boldsymbol{x}_m\}_{m=1}^M)\}$ from $p(\boldsymbol{y}, \boldsymbol{X})$.

multimodal sparse coding: Find $\boldsymbol{A}^* = [\boldsymbol{\alpha}_1^*, \ldots, \boldsymbol{\alpha}_M^*]$ in $\mathbf{R}^{p \times M}$.

$$\underset{\boldsymbol{A}}{\operatorname{argmin}} \frac{1}{M} \sum_{m=1}^{M} \frac{1}{2} \|\boldsymbol{x}_m - \boldsymbol{D}_m \boldsymbol{\alpha}_m\|_2^2 + \lambda_1 \Upsilon(\boldsymbol{A}) + \frac{\lambda_2}{2} \|\boldsymbol{A}\|_F^2$$

Find rows of $\boldsymbol{A}^*$ that satisfy Eq. 4.23: active set.

Find $\mathfrak{D}$ from 4.24 and $\mathfrak{X}$ from 4.26.

Find $\mathcal{O}$ in $\mathbf{R}^{\pi M \times \pi M}$ using 4.27.

Compute $\mathcal{Z}_m$ for all $m \in \{1, \ldots, M\}$ from $\mathcal{O}$.

Compute $\boldsymbol{\beta}_m^{1*}$ and $\boldsymbol{\beta}_m^{2*}$ in $\mathbf{R}^{1 \times p}$ $\forall m \in \{1, \ldots, M\}$.

Choose the learning rate $\rho_t \leftarrow \min(\rho, \rho \frac{t_0}{t})$.

Update the parameters by a projected first-order gradient step:

$$\begin{aligned}
&\boldsymbol{W}_m \leftarrow \Pi_{\mathcal{W}_m}[\boldsymbol{W}_m - \rho_t(\nabla_{\boldsymbol{W}_m} \mathcal{L}_s + \nu_1 \boldsymbol{W}_m)] \\
&\boldsymbol{T}_m \leftarrow \Pi_{\mathcal{T}_m}[\boldsymbol{T}_m - \rho_t(\nabla_{\boldsymbol{T}_m} \mathcal{L}_s + \nu_2 \boldsymbol{T}_m)] \\
&\left. + (1-\mu)\left[(\boldsymbol{x}_m - \boldsymbol{D}_m \boldsymbol{\alpha}_m^*)\boldsymbol{\beta}_m^{2*\top} - \boldsymbol{D}_m \boldsymbol{\beta}_m^{2*} \boldsymbol{\alpha}_m^{*\top}\right]\right)\Bigg],
\end{aligned}$$

where $\Pi_{\mathcal{W}_m}$, $\Pi_{\mathcal{T}_m}$ and $\Pi_{\mathcal{D}_m}$ are orthogonal projections on the sets $\mathcal{W}_m$, $\mathcal{T}_m$ and $\mathcal{D}_m$, respectively.

end for

end for

return $\{\boldsymbol{W}_m, \boldsymbol{T}_m, \boldsymbol{D}_m\}_{m=1}^M$

obtained at the previous iteration solving 4.18, with learning the new dictionaries using the gradient step of Eq. 4.29 over the convex sets $\{\mathcal{D}_m\}_{m=1}^{M}$. The advantage of this implementation is that for each mini-batch, we only need to find $\boldsymbol{\beta}^{k*}$ once. One may have concern about singularity of $\mathcal{Z}_m$, especially when number of atoms is small. However, $\lambda_1 \geq 0$ and $\lambda_2 > 0$ makes $\left(\mathfrak{D}^\intercal\mathfrak{D} + \lambda_1\mathfrak{X} + \lambda_2\boldsymbol{I}\right)^{-1}$ positive definite and hence a unique solution for the linear equations of $\boldsymbol{\beta}^{1*}$ and $\boldsymbol{\beta}^{2*}$ is guaranteed. But, in practice with λ_1 enough large, the $\mathfrak{D}$ becomes full column rank, the Eq. 4.27 is stable and accept the Cholesky decomposition and therefore there is no need to Frobenius norm ($\lambda_2 = 0$). This is equal to have matrix $\mathfrak{D}$ with atoms that do not have high correlation or simply assuming the summation of the smallest eigenvalue of $\mathfrak{D}^\intercal\mathfrak{D}$ and λ_1 to be greater than zero, which is common assumption in literature [35, 105, 13]. Despite the fact that our method like any other non-convex optimization method in the literature cannot guarantee to find the global optimum of the optimization problem and may end up with a stationary points; we will demonstrate in the experiment that these stationary points are acceptable for practical purposes. This is to some extent depends on the "good" initialization of the optimization variables. Similar to [105, 13] the dictionaries $\{\boldsymbol{D}_m\}_{m=1}^{M}$ are initialized by solution of the multimodal and data-driven dictionary learning in (4.18). We exploit the generated sparse codes of m-th feature as features to train modality-based classifiers, $\boldsymbol{W}_m$, by solving (4.5) with adopting multivariate ridge regression model [49] with quadratic loss and ℓ_2 norm regularization: $\boldsymbol{W}_m = \boldsymbol{H}\boldsymbol{X}_m^\top(\boldsymbol{X}_m\boldsymbol{X}_m^\top + \nu\boldsymbol{I})^{-1}$. The same is done to initialize transformation matrix $\boldsymbol{T}_m = \boldsymbol{Q}\boldsymbol{X}_m^\top(\boldsymbol{X}_m\boldsymbol{X}_m^\top + \nu\boldsymbol{I})^{-1}$.

The learning rate ρ_t is chosen based on the heuristic rule proposed in [105], *i.e.* $\rho_t = \min(\rho, \rho t_0/t)$, whose ρ and t_0 are constant parameters. The result of this form of learning rate would be a constant learning rate ρ in first t_0 iterations, and an annealing strategy of $1/t$ for the upcoming iterations, $t > t_0$. We experimentally find $t_0 = T/10$ to work well for all our experiments, where T is the total number of iterations.Then, for first few iterations, we examine various values for ρ and the one that lead to lowest error on a small validation set is kept. The size of the mini-batch is chosen to be 100 for all experiments.

4.7 Proof

Given the structure of feature grouping, $\mathcal{G}$ and multimodal dictionaries $\{\boldsymbol{D}_m\}_{m=1}^M$, we propose to obtain multimodal sparse representation of the candidate patch, $\boldsymbol{X}$, while imposing the tree-structured joint sparsity model over $\{\boldsymbol{x}_m\}_{m=1}^M$:

4.7.1 Case : M=1

Definition 4.3. *(Active set:) The active set* Λ *of the sparse representation* $\boldsymbol{\alpha}^\star \in \mathbf{R}^p$ *with* p *elements of* $\boldsymbol{\alpha}_j$ *is defined as*

$$\Lambda = \{j \in \{1, \ldots, p\} : \boldsymbol{\alpha}_j \neq 0\}, |\Lambda| = \pi$$

where π is the size of active set $|\Lambda| = \pi$ and $\pi \leq p$.

The inner-level problem (4.18) is converted to elastic-net

$$\underset{\boldsymbol{W}\in\mathcal{W},\, \boldsymbol{D}\in\mathcal{D}}{\operatorname{argmin}} \quad f(\boldsymbol{D}, \boldsymbol{W}) + \frac{\nu}{2}\|\boldsymbol{W}\|_F^2 \tag{4.31a}$$

$$f(\boldsymbol{D}, \boldsymbol{W}) \triangleq \mathbb{E}_{y,x}\big[\mathcal{L}_s\big(\boldsymbol{y}, \boldsymbol{W}, \boldsymbol{\alpha}^*(\boldsymbol{x}, \boldsymbol{D})\big)\big] \tag{4.31b}$$

$$\boldsymbol{\alpha}^\star(\boldsymbol{x}, \boldsymbol{D}) \triangleq \underset{\boldsymbol{\alpha}\in\mathbf{R}^p}{\operatorname{argmin}} \frac{1}{2}\|\boldsymbol{x} - \boldsymbol{D}\boldsymbol{\alpha}\|_2^2 + \lambda_1\|\boldsymbol{\alpha}\|_1 + \frac{\lambda_2}{2}\|\boldsymbol{\alpha}\|_2^2 \tag{4.31c}$$

where the dictionary $\boldsymbol{D}$ is not defined explicitly in optimization problem (4.31a) but defined implicitly in the inner-level of the bi-level optimization (4.31c). The main challenge is to compute the gradients of the sparse code $\boldsymbol{\alpha}$ with respect to dictionary $\boldsymbol{D}$. We use chain rule to find gradient of cost function $f(\boldsymbol{D}, \boldsymbol{W})$ with respect to $\boldsymbol{D}$. For the non-zero elements of sparse codes, $\boldsymbol{\alpha}_\Lambda^\star$ we calculate the partial derivative of the problem (4.31c) with respect to members of active set

$$\begin{aligned} &\mathbf{0} \in -\boldsymbol{D}_\Lambda^\top(\boldsymbol{x} - \boldsymbol{D}\boldsymbol{\alpha}^\star) + \lambda_1 \operatorname{\mathbf{sign}}(\boldsymbol{\alpha}_\Lambda^\star) + \lambda_2\boldsymbol{\alpha}_\Lambda^\star \Rightarrow \\ &\boldsymbol{D}_\Lambda^\top(\boldsymbol{D}\boldsymbol{\alpha}^\star - \boldsymbol{x}) + \lambda_2\boldsymbol{\alpha}_\Lambda^\star = -\lambda_1 \operatorname{\mathbf{sign}}(\boldsymbol{\alpha}_\Lambda^\star) \end{aligned} \tag{4.32}$$

For simplicity, we drop Λ and $\star$ symbols. Then, we compute the partial derivative of both sides in (4.32) with respect to the element of the dictionary in i-th row and j-th column $\boldsymbol{D}_{ij}$

$$\frac{\partial\Big(\boldsymbol{D}_\Lambda^\top(\boldsymbol{D}\boldsymbol{\alpha}^\star-\boldsymbol{x})+\lambda_2\boldsymbol{\alpha}_\Lambda^\star+\lambda_1\,\mathbf{sign}\,(\boldsymbol{\alpha}_\Lambda^\star)\Big)}{\partial\boldsymbol{D}_{ij}}=0 \tag{4.33}$$

$$\mathcal{I}_{ij}{}^\top\boldsymbol{x}-\mathcal{I}_{ij}{}^\top\boldsymbol{D}\boldsymbol{\alpha}-\boldsymbol{D}^\top\mathcal{I}_{ij}\boldsymbol{\alpha}-\boldsymbol{D}^\top\boldsymbol{D}\frac{\partial\boldsymbol{\alpha}}{\partial\boldsymbol{D}_{ij}}-\lambda_2\frac{\partial\boldsymbol{\alpha}}{\partial\boldsymbol{D}_{ij}}=0\Rightarrow \tag{4.34}$$

$$\frac{\partial\boldsymbol{\alpha}}{\partial\boldsymbol{D}_{ij}}=(\boldsymbol{D}^\top\boldsymbol{D}+\lambda_2\boldsymbol{I})^{-1}\Big(\mathcal{I}_{ij}{}^\top(\boldsymbol{x}-\boldsymbol{D}\boldsymbol{\alpha})-\boldsymbol{D}^\top\mathcal{I}_{ij}\boldsymbol{\alpha}\Big) \tag{4.35}$$

$$\frac{\partial\boldsymbol{\alpha}}{\partial\boldsymbol{D}_{ij}}=(\boldsymbol{D}^\top\boldsymbol{D}+\lambda_2\boldsymbol{I})^{-1}\Big(\mathcal{I}_{ij}{}^\top(\boldsymbol{x}-\boldsymbol{D}\boldsymbol{\alpha})-\boldsymbol{D}_{i\rightarrow}{}^\top\boldsymbol{\alpha}_j\Big) \tag{4.36}$$

where $\boldsymbol{\alpha}_j$ is j-th element of $\boldsymbol{\alpha}$, $\boldsymbol{D}_{i\rightarrow}$ is the i-th row of the dictionary and $\mathcal{I}$ is a binary matrix with n rows and p columns and $\mathcal{I}_{ij}$ means that only i-th row and j-th column is one and other indices are zero. Using the chain rule

$$\frac{\partial f}{\partial\boldsymbol{D}}=\frac{\partial f}{\partial\boldsymbol{\alpha}}\frac{\partial\boldsymbol{\alpha}}{\partial\boldsymbol{D}} \tag{4.37}$$

$$\frac{\partial f}{\partial\boldsymbol{\alpha}}=\frac{\partial\mathcal{L}_s\big(\boldsymbol{y},\boldsymbol{W},\boldsymbol{\alpha}^*(\boldsymbol{x},\boldsymbol{D})\big)}{\partial\boldsymbol{\alpha}_\Lambda} \tag{4.38}$$

$$\boldsymbol{\beta}_\Lambda^\star=\Big(\boldsymbol{D}_\Lambda{}^\top\boldsymbol{D}_\Lambda+\lambda_2\boldsymbol{I}\Big)^{-1}\frac{\partial\mathcal{L}_s\big(\boldsymbol{y},\boldsymbol{W},\boldsymbol{\alpha}^*(\boldsymbol{x},\boldsymbol{D})\big)}{\partial\boldsymbol{\alpha}_\Lambda}\quad\text{and} \tag{4.39}$$

$$\boldsymbol{\beta}_{j\notin\Lambda}^\star=0$$

$$\frac{\partial f}{\partial\boldsymbol{D}_\Lambda}=\begin{bmatrix}\boldsymbol{\beta}_\Lambda^\star{}^\top\Big(\mathcal{I}_{11}^\top(\boldsymbol{x}-\boldsymbol{D}\boldsymbol{\alpha})-\boldsymbol{D}_{1\rightarrow}^\top\boldsymbol{\alpha}_1\Big) & \dots & \boldsymbol{\beta}_\Lambda^\star{}^\top\Big(\mathcal{I}_{1\pi}^\top(\boldsymbol{x}-\boldsymbol{D}\boldsymbol{\alpha})-\boldsymbol{D}_{1\rightarrow}^\top\boldsymbol{\alpha}_\pi\Big)\\ \vdots & \dots & \vdots\\ \boldsymbol{\beta}_\Lambda^\star{}^\top\Big(\mathcal{I}_{n1}^\top(\boldsymbol{x}-\boldsymbol{D}\boldsymbol{\alpha})-\boldsymbol{D}_{n\rightarrow}^\top\boldsymbol{\alpha}_1\Big) & \dots & \boldsymbol{\beta}_\Lambda^\star{}^\top\Big(\mathcal{I}_{n\pi}^\top(\boldsymbol{x}-\boldsymbol{D}\boldsymbol{\alpha})-\boldsymbol{D}_{n\rightarrow}^\top\boldsymbol{\alpha}_\pi\Big)\end{bmatrix} \tag{4.40}$$

and finally

$$\frac{\partial f}{\partial \boldsymbol{D}} = (\boldsymbol{x} - \boldsymbol{D}\boldsymbol{\alpha})\boldsymbol{\beta}^{\star\top} - \boldsymbol{D}\boldsymbol{\beta}^{\star}\boldsymbol{\alpha}^{\top} \tag{4.41}$$

4.7.2 Case: Multimodal with Joint Sparsity

In this section, the i-th sample $\boldsymbol{X}^i$ is observed from M features, $\boldsymbol{X}^i = \{\boldsymbol{x}_m^i\}_{m=1}^M$. We intend to estimate corresponding multimodal decomposition coefficients $\boldsymbol{A}^\star(\{\boldsymbol{x}_m^i, \boldsymbol{D}_m\}_{m=1}^M) = [\boldsymbol{\alpha}_1^{i\star}, \ldots, \boldsymbol{\alpha}_M^{i\star}]$. We consider following bi-level optimization problem

$$\underset{\{\boldsymbol{D}_m \in \mathcal{D}_m, \boldsymbol{W}_m \in \mathcal{W}_m\}_{m=1}^M}{\operatorname{argmin}} \quad f(\{\boldsymbol{D}_m, \boldsymbol{W}_m\}) + \frac{\nu}{2}\sum_{m=1}^{M} \|\boldsymbol{W}_m\|_F^2 \tag{4.42a}$$

$$f(\{\boldsymbol{D}_m, \boldsymbol{W}_m\}) \triangleq \mathbb{E}_{y, \{\boldsymbol{x}_m^i\}_{m=1}^M}\left[\sum_{m=1}^{M} \mathcal{L}_s(y, \boldsymbol{W}_m, \boldsymbol{\alpha}_m^{i\star}(\boldsymbol{x}_m^i, \boldsymbol{D}_m))\right] \tag{4.42b}$$

$$\boldsymbol{A}^\star(\{\boldsymbol{x}_m^i, \boldsymbol{D}_m\}_{m=1}^M) \triangleq \underset{\boldsymbol{A} \in \mathbf{R}^{p \times M}}{\operatorname{argmin}} \frac{1}{2}\sum_{m=1}^{M} \|\boldsymbol{x}_m^i - \boldsymbol{D}_m\boldsymbol{\alpha}_m^i\|_2^2 + \lambda_1\|\boldsymbol{A}\|_{1,2} + \frac{\lambda_2}{2}\|\boldsymbol{A}\|_F^2 \tag{4.42c}$$

for simplicity we drop the symbol $\star$. The main difficulty is to calculate the partial differential of cost function f in optimization problem (4.42a) with respect to dictionary $\boldsymbol{D}_m$. Because dictionary is not explicitly defined in optimization problem (4.42a); but, it is defined implicitly in inner-level problem. The other challenge is that the optimization problem (4.42a) is not differentiable with respect to $\boldsymbol{A}^\star(\{\boldsymbol{x}_m^i, \boldsymbol{D}_m\}_{m=1}^M)$. However, we will show that the function $f(\{\boldsymbol{D}_m, \boldsymbol{W}_m\})$ defined in (4.42b) is differentiable on space of $\mathcal{D}_1 \times \cdots \times \mathcal{D}_M \times \mathcal{W}_1 \times \cdots \times \mathcal{W}_M$. We use chain rule to compute the step direction of gradient descent algorithm for optimization variable $\boldsymbol{D}_m$.

Definition 4.4. *(Active set:) The active set Λ of the multimodal sparse representation $\boldsymbol{A}^\star \in \mathbf{R}^{p \times M}$ with p rows of $\{\boldsymbol{A}_{j\rightarrow}^\star \in \mathbf{R}^M\}_{j=1}^p$ is defined as*

$$\Lambda = \{j \in \{1, \ldots, p\} : \|\boldsymbol{A}_{j\rightarrow}\|_{\ell_2} \neq 0\}, |\Lambda| = \pi$$

where π is the size of active set $|\Lambda| = \pi$ and $\pi \leq p$.

For the rows that belong to the active set, $\boldsymbol{A}^{\star}_{\Lambda}$ we calculate the partial derivative of the problem (4.42c) with respect to members of active set. Consider $\mathcal{A} \in \mathbf{R}^{\pi \times M}$ to only include π rows of $\boldsymbol{A}$ that are members of active set: $\mathcal{A} = \{\boldsymbol{A}_{i\rightarrow}\}_{i\in\Lambda}$. We show the i-th row of $\mathcal{A}$ as $\mathcal{A}_{i\rightarrow}$, the j-th column as $\mathcal{A}^{j}$ and the element of i-th row and j-th column with $\mathcal{A}_{ij}$. As a reminder the dictionary in the modality m is represented as $\boldsymbol{D}_m = [\boldsymbol{d}^1_m, \dots, \boldsymbol{d}^p_m]$.

$$\frac{\partial}{\partial \mathcal{A}} = \begin{bmatrix} -\boldsymbol{d}_1^{1\top}(\boldsymbol{x}_1 - \boldsymbol{D}_1\mathcal{A}^1) + \lambda_1 \operatorname{\mathbf{sign}}(\mathcal{A}_{11}) + \lambda_2\mathcal{A}_{11} & \dots & -\boldsymbol{d}_M^{1\top}(\boldsymbol{x}_M - \boldsymbol{D}_M\mathcal{A}^M) + \lambda_1 \operatorname{\mathbf{sign}}(\mathcal{A}_{1M}) + \lambda_2\mathcal{A}_{1M} \\ \vdots & \ddots & \vdots \\ -\boldsymbol{d}_1^{\pi\top}(\boldsymbol{x}_1 - \boldsymbol{D}_1\mathcal{A}^1) + \lambda_1 \operatorname{\mathbf{sign}}(\mathcal{A}_{\pi 1}) + \lambda_2\mathcal{A}_{\pi 1} & \dots & -\boldsymbol{d}_M^{\pi\top}(\boldsymbol{x}_M - \boldsymbol{D}_M\mathcal{A}^M) + \lambda_1 \operatorname{\mathbf{sign}}(\mathcal{A}_{\pi M}) + \lambda_2\mathcal{A}_{\pi M} \end{bmatrix} \tag{4.43}$$

where row j of (4.43) can be written as

$$\mathbf{0} = \Bigg[-{\boldsymbol{d}_1^j}^{\top}(\boldsymbol{x}_1 - \boldsymbol{D}_1\mathcal{A}^1) + \lambda_1 \frac{\mathcal{A}_{j1}}{\|\mathcal{A}_{j\rightarrow}\|_{\ell_2}} + \lambda_2\mathcal{A}_{j1}, \dots, \\ -{\boldsymbol{d}_M^j}^{\top}(\boldsymbol{x}_M - \boldsymbol{D}_M\mathcal{A}^M) + \lambda_1 \frac{\mathcal{A}_{jM}}{\|\mathcal{A}_{j\rightarrow}\|_{\ell_2}} + \lambda_2\mathcal{A}_{jM} \Bigg] \tag{4.44}$$

$$\tag{4.45}$$

we can further simplify it to

$$\frac{\partial f}{\partial \mathcal{A}_{j\rightarrow}} = \Bigg[-{\boldsymbol{d}_1^j}^{\top}(\boldsymbol{x}_1 - \boldsymbol{D}_1\mathcal{A}^1), \dots, -{\boldsymbol{d}_M^j}^{\top}(\boldsymbol{x}_M - \boldsymbol{D}_M\mathcal{A}^M) \Bigg] + \\ \lambda_1 \Bigg[\frac{\mathcal{A}_{j1}}{\|\mathcal{A}_{j\rightarrow}\|_{\ell_2}}, \dots, \frac{\mathcal{A}_{jM}}{\|\mathcal{A}_{j\rightarrow}\|_{\ell_2}} \Bigg] \lambda_2 \Big[\mathcal{A}_{j1}, \dots, \mathcal{A}_{jM} \Big] \tag{4.46}$$

Then, to calculate the step direction for gradient descent minimization over the dictionary, we can compute the partial derivation of (4.43) with respect to each element of the dictionary. As a reminder, for m-th feature, we show i-th row of the dictionary as $\boldsymbol{D}_{m,i\rightarrow}$, the j-th column as $\boldsymbol{D}^j_m$ and the element in i-th row and j-th column as $\boldsymbol{D}_m(i,j)$. It worth to mention that since the multimodal coefficients $\boldsymbol{A}(\{\boldsymbol{x}_m, \boldsymbol{D}_m\}_{m=1}^M)$ is a function of the set of feature-specific dictionaries $\{\boldsymbol{D}_m\}_{m=1}^M$, the partial derivation of each column of multimodal sparse codes $\boldsymbol{A}$, should be calculated with respect to all the members of $\{\boldsymbol{D}_m\}_{m=1}^M$; in other words, $\frac{\partial \boldsymbol{A}^{\acute{m}}}{\partial \boldsymbol{D}_m}$ and

$m \neq \acute{m}$ is not zero by default.

$$\frac{\partial \mathcal{A}_{j\rightarrow}}{\partial \boldsymbol{D}_m(i,j)} = \left[0, \ldots, -\mathcal{I}_i^{j\top}(\boldsymbol{x}_m - \boldsymbol{D}_m\mathcal{A}^m) + \boldsymbol{d}_m^{j\top}\mathcal{I}_{ij}\mathcal{A}^m + \boldsymbol{d}_m^{j\top}\boldsymbol{D}_m\frac{\partial \mathcal{A}^m}{\partial \boldsymbol{D}_m(i,j)}, \ldots\right] + \lambda_1 \frac{\partial \left[\frac{\mathcal{A}_{j1}}{\|\mathcal{A}_{j\rightarrow}\|_{\ell_2}}, \ldots, \frac{\mathcal{A}_{jM}}{\|\mathcal{A}_{j\rightarrow}\|_{\ell_2}}\right]}{\partial \boldsymbol{D}_m(i,j)} + \lambda_2 \frac{\partial \left[\mathcal{A}_{j1}, \ldots, \mathcal{A}_{jM}\right]}{\partial \boldsymbol{D}_m(i,j)} \tag{4.47}$$

where $\mathcal{I}_{ij}$ is an indicator function with n_m rows and p columns and it is zero everywhere except in row i and column j and $\mathcal{I}_i^j$ is a vector with size n_m that is zero everywhere except in i-th row. We can further simplify second term of (4.47)

$$\frac{\partial \left[\frac{\mathcal{A}_{j1}}{\|\mathcal{A}_{j\rightarrow}\|_{\ell_2}}, \ldots, \frac{\mathcal{A}_{jM}}{\|\mathcal{A}_{j\rightarrow}\|_{\ell_2}}\right]}{\partial \boldsymbol{D}_m(i,j)} = \frac{\partial \frac{\mathcal{A}_{j\rightarrow}}{\|\mathcal{A}_{j\rightarrow}\|_{\ell_2}}}{\partial \boldsymbol{D}_m(i,j)} = \frac{\|\mathcal{A}_{j\rightarrow}\|_{\ell_2}\frac{\partial \mathcal{A}_{j\rightarrow}}{\partial \boldsymbol{D}_m(i,j)} - \mathcal{A}_{j\rightarrow}\frac{\partial \|\mathcal{A}_{j\rightarrow}\|_{\ell_2}}{\partial \boldsymbol{D}_m(i,j)}}{(\|\mathcal{A}_{j\rightarrow}\|_{\ell_2})^2} =$$

$$\frac{\partial \mathcal{A}_{j\rightarrow}}{\partial \boldsymbol{D}_m(i,j)}\frac{1}{\|\mathcal{A}_{j\rightarrow}\|_{\ell_2}}\left(\boldsymbol{I} - \frac{\Theta_{jj}}{\left(\|\mathcal{A}_{j\rightarrow}\|_{\ell_2}\right)^2}\right) \in \mathbf{R}^{1\times M} \tag{4.48}$$

$$\Theta_{jj} \in \mathbf{R}^{M\times M} = \mathcal{A}_{j\rightarrow}^\top \mathcal{A}_{j\rightarrow} \text{ and } \Xi_j \in \mathbf{R}^{M\times M} = [\boldsymbol{\xi}^1, \ldots, \boldsymbol{\xi}^M] \tag{4.49}$$

$$\boldsymbol{\xi}^m \in \mathbf{R}^{M\times 1} = \frac{1}{\|\mathcal{A}_{j\rightarrow}\|_{\ell_2}}\left(\boldsymbol{I}^m - \frac{\mathcal{A}_{j\rightarrow}^\top \mathcal{A}_{jm}}{\left(\|\mathcal{A}_{j\rightarrow}\|_{\ell_2}\right)^2}\right)$$

where $\boldsymbol{I}^m$ is the m-th column of identity matrix $\boldsymbol{I}$. So, the problem (4.47) can be written as

$$\frac{\partial \mathcal{A}_{j\rightarrow}^\top}{\partial \boldsymbol{D}_m(i,j)} = \left[0, \ldots, -\mathcal{I}_i^{j\top}(\boldsymbol{x}_m - \boldsymbol{D}_m\mathcal{A}^m) + \boldsymbol{d}_m^{j\top}\mathcal{I}_{ij}\mathcal{A}^m + \boldsymbol{d}_m^{j\top}\boldsymbol{D}_m\frac{\partial \mathcal{A}^m}{\partial \boldsymbol{D}_m(i,j)}, \ldots\right]^\top + \lambda_1 \frac{1}{\|\mathcal{A}_{j\rightarrow}\|_{\ell_2}}\left(\boldsymbol{I} - \frac{\Theta_{jj}}{\left(\|\mathcal{A}_{j\rightarrow}\|_{\ell_2}\right)^2}\right)\frac{\partial \mathcal{A}_{j\rightarrow}^\top}{\partial \boldsymbol{D}_m(i,j)} + \lambda_2 \frac{\partial \mathcal{A}_{j\rightarrow}^\top}{\partial \boldsymbol{D}_m(i,j)} \tag{4.50}$$

now, we generalize problem (4.50)

$$\Big[\mathbf{0}_{\pi\times 1}, -\mathcal{I}_{ij}^{\top}(\boldsymbol{x}_m - \boldsymbol{D}_m\mathcal{A}^m) + \boldsymbol{D}_m^{\top}\mathcal{I}_{ij}\mathcal{A}^m + \boldsymbol{D}_m^{\top}\boldsymbol{D}_m\tfrac{\partial\mathcal{A}^m}{\partial\boldsymbol{D}_m(ij)}, \mathbf{0}_{\pi\times 1}\Big] + \tag{4.51}$$
$$\lambda_1 \begin{bmatrix} \left[\frac{\partial\mathcal{A}_{11}}{\partial\boldsymbol{D}_m(ij)} \quad \cdots \quad \frac{\partial\mathcal{A}_{1M}}{\partial\boldsymbol{D}_m(ij)}\right]\boldsymbol{\Xi}_1 \\ \vdots \quad \ddots \quad \vdots \\ \left[\frac{\partial\mathcal{A}_{\pi 1}}{\partial\boldsymbol{D}_m(ij)} \quad \cdots \quad \frac{\partial\mathcal{A}_{\pi M}}{\partial\boldsymbol{D}_m(ij)}\right]\boldsymbol{\Xi}_\pi \end{bmatrix} + \lambda_2\frac{\partial\mathcal{A}}{\partial\boldsymbol{D}_m(ij)} = \mathbf{0}_{\pi\times M}$$
$$\forall j \in \Lambda,\ \boldsymbol{\Xi}_j = [\boldsymbol{\xi}^1, \ldots, \boldsymbol{\xi}^M] \text{ and } \boldsymbol{\xi}^m = \frac{1}{\|\mathcal{A}_{j\rightarrow}\|_{\ell_2}}\left(\boldsymbol{I}^m - \frac{\mathcal{A}_{j\rightarrow}^{\top}\mathcal{A}_{jm}}{\left(\|\mathcal{A}_{j\rightarrow}\|_{\ell_2}\right)^2}\right)$$

where $\mathbf{0}_{\pi\times 1}$ is a zero vector of size π. To factor the partial derivation with respect to multimodal sparse codes, we need to define more variables: Consider $\tilde{\boldsymbol{d}}_m^i \in \mathbf{R}^n$ as extended version of atom $\boldsymbol{d}_m^i \in \mathbf{R}^{n_m}$ with zeros, where $n = \sum_{m=1}^{M} n_m$. Then, we concatenate horizontally atom $j \in \Lambda$ from all features as $\tilde{\boldsymbol{\Delta}}_j \in \mathbf{R}^{n\times M} = [\boldsymbol{d}_1^j, \ldots, \boldsymbol{d}_M^j]$. The block diagonal matrix $\mathfrak{D}$ is made as

$$\mathfrak{D} \in \mathbf{R}^{pM\times pM} = \begin{bmatrix} \tilde{\boldsymbol{\Delta}}_1{}^T\tilde{\boldsymbol{\Delta}}_1 & \ldots & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \tilde{\boldsymbol{\Delta}}_p{}^T\tilde{\boldsymbol{\Delta}}_p \end{bmatrix}$$

Similarly, we make block diagonal matrix $\mathfrak{X}$ using the $\boldsymbol{\Xi}_j$ and $j \in \Lambda$ as elements of diagonal

$$\mathfrak{X} \in \mathbf{R}^{\pi M\times \pi M} = \begin{bmatrix} \boldsymbol{\Xi}_1 & \ldots & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \boldsymbol{\Xi}_\pi \end{bmatrix}$$

where $\{\boldsymbol{\Xi}_j\}_{j\in[\![1;p]\!]} \in \mathbf{R}^{M\times M}$ is defined in (4.51).

$$\frac{\partial\mathcal{A}}{\partial\boldsymbol{D}_m(ij)} = \left(\mathfrak{D}^{\top}\mathfrak{D} + \lambda_1\mathfrak{X} + \lambda_2\boldsymbol{I}\right)^{-1}$$
$$\Big[\mathbf{0}_{\pi}^{\top}, \ldots, \iota_{ij}^{1\top}(\boldsymbol{x}_m - \boldsymbol{D}_m\mathcal{A}^m) - \boldsymbol{d}_m^{1\top}\mathcal{I}_{ij}\mathcal{A}^m, \ldots, \iota_{ij}^{\pi\top}(\boldsymbol{x}_m - \boldsymbol{D}_m\mathcal{A}^m) - \boldsymbol{d}_m^{\pi\top}\mathcal{I}_{ij}\mathcal{A}^m \ldots, \mathbf{0}_{\pi}^{\top}\Big] \tag{4.52}$$

where $\{\iota_{ij}^{c} \in \mathbf{R}^{\pi}\}_{c\in[\![1;\pi]\!]}$ is the c-th column of $\mathcal{I}_{ij}$. We define the first term in (4.52) as $\mathcal{O} \in \mathbf{R}^{\pi M\times\pi M} = \left(\mathfrak{D}^{\top}\mathfrak{D} + \lambda_1\mathfrak{X} + \lambda_2\boldsymbol{I}\right)^{-1}$. By vectorizing the both sides of (4.52)

$$\text{vec}\left(\frac{\partial\mathcal{A}^{\top}}{\partial\boldsymbol{D}_m(ij)}\right) = \mathcal{Z}\mathcal{I}_{ij}^{\top}(\boldsymbol{x}_m - \boldsymbol{D}_m\mathcal{A}^m) - \mathcal{A}_{jm}\mathcal{Z}\boldsymbol{D}_{m,i\rightarrow}^{\top} \tag{4.53}$$

where $\mathcal{Z} \in \mathbf{R}^{\pi M\times\pi}$ is made by putting together horizontally those columns of $\mathcal{O}$ that correspond to dictionary of m feature: $\{m, m+M, m+2M, \ldots, m+(p-1)M\}$. The $\mathcal{A}_{jm}$ is the j-th element of the vector $\mathcal{A}^m$. Finally, using the chain rule we obtain

$$\frac{\partial f}{\partial\boldsymbol{D}_m(ij)} = \text{vec}\left(\frac{\partial f}{\partial\boldsymbol{A}^{\top}}\right)^{\top}\text{vec}\left(\frac{\partial\boldsymbol{A}^{\top}}{\partial\boldsymbol{D}_m(ij)}\right) \tag{4.54}$$

The obtained formula in (4.54) is the partial derivative of the loss function $f = \sum_{m=1}^{M}\mathcal{L}_s(\sum_{m=1}^{M}\mathcal{L}_s(y, \boldsymbol{W}_m, \boldsymbol{\alpha}_m^{i\star}(\boldsymbol{x}_m^i, \boldsymbol{D}_m))$ with respect to the element of i-th row and j-th column of the dictionary in m-th feature. Assuming $\mathfrak{b} \in \mathbf{R}^{1\times\pi} = \text{vec}\left(\frac{\partial f}{\partial\boldsymbol{A}^{\top}}\right)^{\top}\mathcal{Z}$ we generalize Eq.(4.54) for the dictionary as

$$\begin{aligned}
&\frac{\partial f}{\partial\boldsymbol{D}_m} \in \mathbf{R}^{n_m\times\pi} = \\
&E\left(\begin{bmatrix} \mathfrak{b}\left(\mathcal{I}_{11}^{\top}(\boldsymbol{x}_m - \boldsymbol{D}_m\boldsymbol{A}^m) - \boldsymbol{A}_{1m}\boldsymbol{D}_{1\rightarrow}^{\top}\right) & \ldots & \mathfrak{b}\left(\mathcal{I}_{1\pi}^{\top}(\boldsymbol{x}_m - \boldsymbol{D}_m\boldsymbol{A}^m) - \boldsymbol{A}_{\pi m}\boldsymbol{D}_{1\rightarrow}^{\top}\right) \\ \vdots & \ddots & \vdots \\ \mathfrak{b}\left(\mathcal{I}_{n_m1}^{\top}(\boldsymbol{x}_m - \boldsymbol{D}_m\boldsymbol{A}^m) - \boldsymbol{A}_{1m}\boldsymbol{D}_{n_m\rightarrow}^{\top}\right) & \ldots & \mathfrak{b}\left(\mathcal{I}_{n_m\pi}^{\top}(\boldsymbol{x}_m - \boldsymbol{D}_m\boldsymbol{A}^m) - \boldsymbol{A}_{\pi m}\boldsymbol{D}_{n_m\rightarrow}^{\top}\right) \end{bmatrix}\right) = \\
&E\left[(\boldsymbol{x}_m - \boldsymbol{D}_m\boldsymbol{A}^m)\mathfrak{b} - \boldsymbol{D}_m\mathfrak{b}^{\top}\mathcal{A}^{m\top}\right]
\end{aligned} \tag{4.55}$$

4.7.3 case : Multimodal with M features with Tree-Structure

In this section, the i-th sample $\boldsymbol{X}^i$ is observed from M features, $\boldsymbol{X}^i = \{\boldsymbol{x}_m^i\}_{m=1}^{M}$. We intend to estimate corresponding multimodal decomposition coefficients $\boldsymbol{A}^{\star}(\{\boldsymbol{x}_m^i, \boldsymbol{D}_m\}_{m=1}^{M}) =$

$[\boldsymbol{\alpha}_1^{i\star}, \ldots, \boldsymbol{\alpha}_M^{i\star}]$. We consider following bi-level optimization problem

$$\underset{\{\boldsymbol{D}_m \in \mathcal{D}_m, \boldsymbol{W}_m \in \mathcal{W}_m\}_{m \in [\![1;M]\!]}}{\operatorname{argmin}} f(\{\boldsymbol{D}_m, \boldsymbol{w}_m\}) + \frac{\nu}{2} \sum_{m=1}^{M} \|\boldsymbol{W}_m\|_F^2 \tag{4.56a}$$

$$f(\{\boldsymbol{D}_m, \boldsymbol{W}_m\}) \triangleq \mathbb{E}_{y, \{\boldsymbol{x}_m^i\}_{m=1}^M} \left[\sum_{m=1}^{M} \mathcal{L}_s(y, \boldsymbol{W}_m, \boldsymbol{\alpha}_m^{i\star}(\boldsymbol{x}_m^i, \boldsymbol{D}_m)) \right] \tag{4.56b}$$

$$\boldsymbol{A}^\star(\{\boldsymbol{x}_m^i, \boldsymbol{D}_m\}_{m=1}^M) \triangleq \underset{\boldsymbol{A} \in \mathbf{R}^{p \times M}}{\operatorname{argmin}} \frac{1}{M} \sum_{m=1}^{M} \left[\frac{1}{2} \|\boldsymbol{x}_m^i - \boldsymbol{D}_m \boldsymbol{\alpha}_m^i\|_2^2 \right] + \lambda_1 \Upsilon(\boldsymbol{A}) + \frac{\lambda_2}{2} \|\boldsymbol{A}\|_F^2 \tag{4.56c}$$

$$\Upsilon(\boldsymbol{A}) \triangleq \sum_{d=1}^{p} \sum_{\mathcal{G}_v \in \mathcal{G}} \omega_{\mathcal{G}_v} \left\| \boldsymbol{A}_{d\rightarrow}^{(\mathcal{G}_v)} \right\|_{\ell_2} \tag{4.56d}$$

4.7.4 Multi-Task Learning of Hierarchical Structures

$$\underset{\{\boldsymbol{D}_m \in \mathcal{D}_m, \boldsymbol{W}_m \in \mathcal{W}_m\}_{m \in [\![1;M]\!]}}{\operatorname{argmin}} f(\{\boldsymbol{D}_m, \boldsymbol{W}_m\}) + \frac{\nu}{2} \sum_{m=1}^{M} \|\boldsymbol{W}_m\|_F^2 \tag{4.57a}$$

$$f(\{\boldsymbol{D}_m, \boldsymbol{W}_m\}) \triangleq \mathbb{E}_{y, \{\boldsymbol{x}_m^i\}_{m=1}^M} \left[\sum_{m=1}^{M} \mathcal{L}_s(y, \boldsymbol{W}_m, \boldsymbol{\alpha}_m^{i\star}(\boldsymbol{x}_m^i, \boldsymbol{D}_m)) \right]$$

$$\{\boldsymbol{A}^{i\star}(\{\boldsymbol{x}_m^i, \boldsymbol{D}_m\}_{m=1}^M)\}_{i=1}^N \triangleq$$

$$\underset{\boldsymbol{A} \in \mathbf{R}^{p \times M}}{\operatorname{argmin}} \frac{1}{N} \sum_{i=1}^{N} \left[\frac{1}{M} \sum_{m=1}^{M} \frac{1}{2} \left[\|\boldsymbol{x}_m^i - \boldsymbol{D}_m \boldsymbol{A}^m\|_2^2 + \lambda_1 \Upsilon\big(\boldsymbol{A}\big) \right] + \lambda_2 \Omega(\mathcal{A}) \right] \tag{4.57b}$$

$$\Upsilon(\boldsymbol{A}) \triangleq \sum_{d=1}^{p} \sum_{\mathcal{G}_v \in \mathcal{G}} \omega_{\mathcal{G}_v} \left\| \boldsymbol{A}_{d\rightarrow}^{(\mathcal{G}_v)} \right\|_{\ell_2} \tag{4.57c}$$

where the $\Upsilon(\boldsymbol{A})$ is defined as (4.57) and $\mathcal{A}$ is the matrix in $\mathbf{R}^{p \times MN}$. It is made by concatenation of multimodal decomposition coefficients of all signals, $\mathcal{A} = [\boldsymbol{A}^1, \ldots, \boldsymbol{A}^N]$; whose elements in the d-th row correspond to decomposition coefficients produced by the d-th atom for all signals from every feature. The last term in (4.57b) applies ℓ_{12} on the rows of $\boldsymbol{A}$, $\Omega(\boldsymbol{A}) = \sum_{d=1}^{p} \|\boldsymbol{A}_{d\rightarrow}\|_2^2$. The penalty in optimization problem (4.57b) is a combination of Υ and Ω on multi-feature sparse representations, is in fact an instance of general overlapping groups.

The optimization problem (4.57) is solved by alternating between optimization variables $\boldsymbol{D}$ and $\boldsymbol{A}$, while one variable is optimizing, the other one is fixed [108, 154, 72]. It is worth to mention that if $\lambda_2 = 0$ and the tree-structure sparsity Υ changed to ℓ_{11}, then the problem (4.57b) will changed to standard sparse coding problem. We solve the approximated version of the problem (4.57b) as follows: we consider the current tree-structure fixed by assigning λ_2 to zero and find the multi-feature sparse codes $\{\boldsymbol{A}^i\}_{i=1}^N$. Then, we prune the dictionaries from atoms that do not contribute well by applying the joint sparsity regularization on each atom: $\|\boldsymbol{A}\|_{\ell_{12}}$.

Appendix

We present the proof of Proposition 4.2 as following. The function f is differentiable with respect to $\boldsymbol{W}_m$ and $\boldsymbol{T}_m$ is because the $\mathcal{Y}$ and $\mathcal{X}$ are assumed to be compact and the $\mathcal{L}_s$ is twice differentiable. Despite the fact that the function $\boldsymbol{\alpha}_m^*$ is not differentiable everywhere, we now show that the function f is differentiable with respect to $\boldsymbol{D}_m$.

We know from optimality conditions from sub-gradient calculus that $\mathbf{0} \in \nabla f + \lambda \partial \Omega$

$$
\begin{aligned}
&[\boldsymbol{D}_1{}^\top(\boldsymbol{x}_1 - \boldsymbol{D}_1\boldsymbol{\alpha}_1^*) + \lambda_2\boldsymbol{\alpha}_1^*, \ldots, \boldsymbol{D}_M{}^\top(\boldsymbol{x}_M - \boldsymbol{D}_M\boldsymbol{\alpha}_M^*) + \lambda_2\boldsymbol{\alpha}_M^*] + \\
&\lambda_1 \partial\Upsilon(\boldsymbol{A}^{*\top}) = \mathbf{0} \qquad (4.58a)\\
&\partial\Upsilon = \Big[\partial\big(\sum_{g\in\mathcal{G}} \boldsymbol{\omega}^{(g)} \circ \boldsymbol{A}^*_{1\rightarrow}\big)^\top, \ldots, \partial\big(\sum_{g\in\mathcal{G}} \boldsymbol{\omega}^{(g)} \circ \boldsymbol{A}^*_{\pi\rightarrow}\big)^\top\Big]^\top \qquad (4.58b)
\end{aligned}
$$

For the rows that belong to the active set, $\boldsymbol{A}^*_\Lambda$ we calculate the partial derivative of the problem (4.58a) with respect to members of active set. Consider $\mathcal{A} \in \mathbf{R}^{\pi\times M}$ to only include π rows of $\boldsymbol{A}$ that are members of active set: $\mathcal{A} = \{\boldsymbol{A}_{i\rightarrow}\}_{i\in\Lambda}$. We show the i-th row of $\mathcal{A}$ as $\mathcal{A}_{i\rightarrow}$, the j-th column as $\mathcal{A}^j$ and the element of i-th row and j-th column with $\mathcal{A}_{ij}$. As a reminder the dictionary in the modality m is represented as $\boldsymbol{D}_m = [\boldsymbol{d}_m^1, \ldots, \boldsymbol{d}_m^p]$.

For the set of groups $\mathcal{G}$ which has $|\mathcal{G}|$ groups of modalities, let us denote $(\boldsymbol{\phi}^{(g)})_{g\in\mathcal{G}} \in \mathbf{R}^{M\times\pi}$ as a $|\mathcal{G}|$-tuple of M dimensional vectors that are zero for indices of modalities that are not

member of $g \in \mathcal{G}$; *i.e.* $\phi_m^{(g)} > 0$ if $m \in g$ and is zero otherwise. Then, for the j-th row of $\boldsymbol{A}_\Lambda$, we define a matrix $\boldsymbol{\Phi} \in \mathbf{R}^{M \times \pi}$ where the j-th column $\boldsymbol{\Phi}^j = (\boldsymbol{\phi}^{(g)})_{g \in \mathcal{G}} \circ \mathcal{A}_{j\rightarrow}^\top$.

The rational behind introducing matrix $\boldsymbol{\Phi}$ with $|\Lambda|M$ variables instead of $\mathcal{A}_{j\rightarrow}$ with M variables is to consider an equivalent problem to (4.18b) that removes the issue of overlapping groups at the cost of a larger number of variables. The partial differential of j-th row of $\mathcal{A}$ with respect to the element of $\boldsymbol{D}_m$ in i-th row and j-th column would be

$$\frac{\partial \Upsilon(\boldsymbol{A}_{j\rightarrow})}{\partial \boldsymbol{D}_m(i,j)} = \boldsymbol{\Xi}_j \frac{\partial \boldsymbol{A}_{j\rightarrow}^\top}{\partial \boldsymbol{D}_m(i,j)} \tag{4.59a}$$

$$\boldsymbol{\Xi}_j \in \mathbf{R}^{M \times M} = \sum_{k=1}^{|\Lambda|} \frac{1}{\|\boldsymbol{\Phi}^k\|_{\ell_2}} \left(\boldsymbol{I} - \frac{\boldsymbol{\Phi}^k \boldsymbol{\Phi}^{k\top}}{\left(\|\boldsymbol{\Phi}^k\|_{\ell_2}\right)^2} \right) \tag{4.59b}$$

4.8 Experiment

In this section we evaluate the performance of HTLDL in four different applications: multi-view object recognition using Berkeley Multiview Wireless (BMW) database [129], multiview face recognition using UMIST [53], multimodal face recognition AR face dataset [116], multiview action recognition using IXMAS [183].

For all the experiments, we choose $\mathcal{L}_s$ same as (4.17c). Samples are normalized to have zero mean and unit ℓ_2 norm. The regularization λ_1 and ν are selected in the set $\{0.0001, 0.0005, 0.001, 0.005, 0.01, 0.05\}$ by cross-validation and $\{10^{-1}, 10^{-2}, \ldots, 10^{-9}\}$. The regularization parameter of the Frobenius norm is chosen as $\lambda_2 = 0.01\lambda_1$.

To compare with the performance of unimodal dictionary learning algorithms, we learn independent dictionaries and classifiers for each modality and then combine the individual scores for a fused decision. This is equivalent to applying ℓ_{11}-norm on $\boldsymbol{A}$ instead of ℓ_{12}-norm in problem (4.18) as $\Omega(\boldsymbol{A}) = \sum |\boldsymbol{A}_{ij}|$ in Eq. (4.18) [156, 14]. The ℓ_{11} does not enforce correlation between the features in space of sparse codes.

4.8.1 Gender Classification

Gender classification is an important task in social activities and communications. In fact, automatically identifying gender is useful for many applications, *e.g.* security surveillance and statistics about customers in places such as movie theaters, building entrances and restaurants.

Most of the published work in gender classification is based on facial images. Moghaddam *et al.* [127] used Support Vector Machines (SVMs) for gender classification from facial images. They used low resolution thumbnail face images (21×12 pixels). Wu *et al.* [186] presented a real time gender classification system using a Look-Up-Table Adaboost algorithm. They extracted demographic information from human faces. Face-based gender classification is still an attractive research area and there is room for developing novel algorithms that are more robust, more accurate and fast.

Similar to [195, 196], we consider the first 25 males and 25 females, 14 images per subject, for training, and testing is done on the rest. We extract three features from each sample and treat them as modalities: raw pixels, quantized gradient [33] and fhog with 9 orientations and 8 bins [41]. Note that, we are not the first one to consider features as modalities; *e.g.* in [90] face recognition is done using edges and raw image intensities as modalities, and in [69] color, gradient, and texture are extracted for feature fusion. We compare JDL, JTLDL and MWDL with recent dictionary learning methods like SRC [185], DLSI [147], DKSVD [206], LC-KSVD [77], FDDL [196], LDL [195], COPAR [89], DLSI [147], and JDL [215].

This experiment is a two-class classification problem with huge variations in each class and large number of training samples. We report the performances for dictionary size of $p = 250$ in Table (4.1) and with $p = 25$ in Table (4.2). When number of atoms are large, $p = 250$, DL methods based on all-vs-all scheme like DKSVD and LCKSVD have less classification accuracy comparing to the class-specific (one-vs-all) DL methods like LDL, FDDL and DLSI. MWDL outperforms others with more than 3%. JDL and JTLDL enhance LDL and LC-KSVD with 0.4%, 5.7% and 0.8%, 6% respectively in Table (4.1).

Table 4.1: The gender classification accuracy (%) with $p = 250$.

Methods	Accuracy	Methods	Accuracy
SRC [185]	93.0	JDL [215]	90.8
Yang et al. [194]	94.5	DLSI [147]	93.2
DKSVD [206]	85.6	FDDL [196]	94.1
LCKSVD [77]	89.5	LDL [195]	94.8
JDL	95.2	**MWDL**	97.9
COPAR [89]	93.4	**JTLDL**	95.6

Table 4.2: Gender classification rates obtained with $p = 25$ atoms.

DLSI	JDL	FDDL	LDL	COPAR	**JDL**	**MWDL**	**JTLDL**
93.7	91.0	92.1	92.4	93.0	92.9	97.1	94.2

To visualize the fact that class-specific DL methods need a large number of atoms, we reduce the number of atoms from $p = 250$ to $p = 25$ and report the one-vs-all DL methods performances in Table (4.2). As we expected, the one-vs-all methods have poor performance with small number of atoms. Although with small p, the accuracy of all methods are reduced, MWDL is more discriminative and outperforms other methods including LDL for more than 4.0%. The accuracy of JTLDL and JDL have reduced by 1.4% and 2.3%. However, MWDL only has 0.8% drop in performance.

4.8.2 Multimodal Face Recognition

The AR database. consists of faces under different poses, illumination and expressions, captured in two sessions [116]. A set of 2,600 images 100 users (50 males and 50 females) are used, each consisting of seven images from the first session as the training samples and seven images from the second session as test samples (Fig. 4.1). We chose randomly 50 out of 700 of the training set as the validation set for optimizing the design parameters. Each face image, with dimension 165×120 pixels, is PCA-transformed and then normalized to have zero mean with unit ℓ_2-norm. We studied the effect of fusion of face as the strong modality along with the four weak modalities. Intensity values are used from each modality.



Figure 4.1: Samples of male and female with extracted modalities in AR dataset.

Table 4.3: Face recognition accuracy with the whole face modality

SVM	SRC	LDL	UTDL	STDL	FDDL	MKL
86.43	88.86	84.56	89.58	90.57	**91.90**	82.86

We crop out and resize the respective rectangular masks of the weak modalities, as shown in Fig. (4.2).

We briefly introduced this setup in Fig. (3.4) in Sec. (2) to illustrate tree-structure relation between modalities. Our intuition is that leveraging different levels of correlation between weak and strong modalities as the hierarchical structure in the space of sparse codes to enhance face recognition performance. The tree $\mathcal{G}$ has $|\mathcal{G}| = 7$ nodes, that includes 5 leaves corresponding to the $M = 5$ modalities and 2 internal nodes. Each internal node encodes a possible grouping between leaves of the subtree which internal node is their root [87]. Here, one internal node represents the high correlation between left and right periocular and the other internal node is the root of the tree that model the grouping between nose, mouth, face and the group of eyes.

Case I: Only Face. We report the face recognition performance of sparse representation classification (SRC) [185], linear support vector machine (SVM) [16], multiple kernel learning (MKL) [146] using linear, polynomial, and RBF kernels, supervised dictionary learning (STDL) [104], latent dictionary leaning (LDL) [195], and fisher discrimination dictionary learning (FDDL) [196] using only the face in Table (4.3). STDL and FDDL outperform other state-of-the-art in face recognition for AR dataset. Also, to have a better idea about each modality, the performance of using single modalities using SVM and SRC algorithms

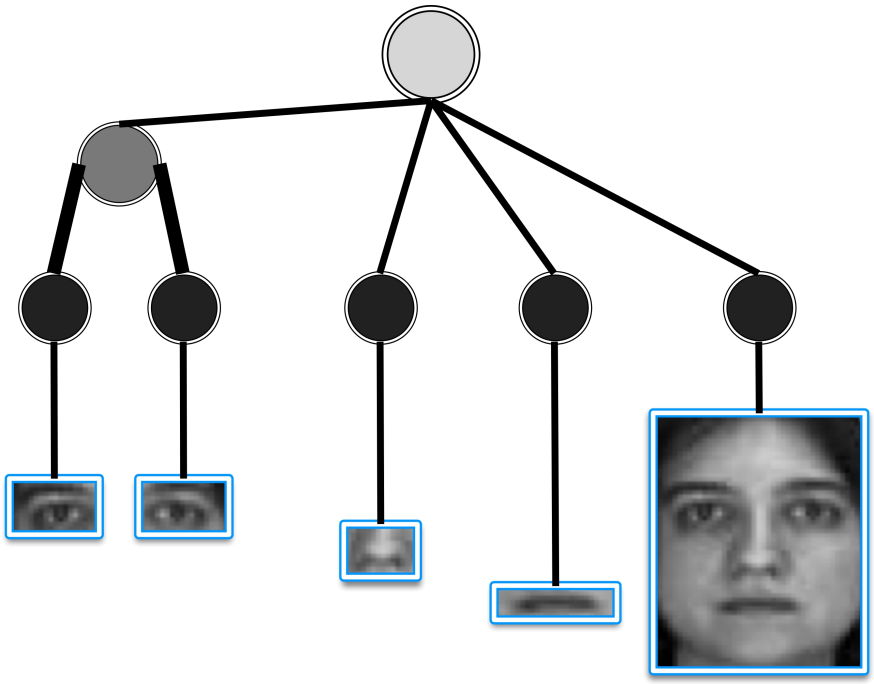


Figure 4.2: We employ the blue rectangular masks and cropping out the corresponding areas. These, along with the whole face, were taken for fusion. Simple intensity values were used as features for all of them. Tree-structure $\mathcal{G}$ corresponding to the four weak modalities of left periocular, right periocular, nose, and mouth, and a strong modality face.

Table 4.4: Recognition performance of each single modality in AR database. Modalities include left periocular, right periocular, nose, mouth, and face.

	Left periocular	Right periocular	Nose	Mouth	Face
SVM	71.00	74.00	44.00	44.29	86.86
SRC	79.29	78.29	63.43	64.14	93.71

are shown in Table 4.4. We did not report the proposed methods in Table 4.4 and Table 4.3 due to the fact that they need multiple sources and in the presence of only one source (only face) they are similar to LC-KSVD or LDL.

Case II: Sparse Regularization Evaluation: ℓ_{11} vs ℓ_{12} vs tree-structure

we learn independent dictionaries and classifiers for each modality and then combine the individual scores for a fused decision. This is equivalent to applying ℓ_{11}-norm on $\boldsymbol{A}$ instead of structural norm (*e.g.* ℓ_{12} or Υ) in problem (4.18).

Table 4.5: Modalities include 1. left periocular, 2. right periocular, 3. nose, 4. mouth, and 5. face.

Modalities	$\{1,2\}$	$\{1,2,3\}$	$\{1,2,3,4\}$	$\{1,2,3,4,5\}$
STDL$_{\ell_{11}}$	83.86	87.86	92.42	95.86
LDL$_{\ell_{11}}$	82.52	87.46	90.16	92.29
LC-KSVD$_{\ell_{11}}$	83.86	89.86	92.42	95.86
MWDL	86.36	87.24	93.16	97.63
JTLDL	**86.43**	89.86	93.57	96.86
HTLDL		87.24	**94.16**	**98.04**

Table 4.6: Multimodal face recognition results for the AR dataset

SVM$_{sum}$	SVM$_{mv}$	STDL$_{\ell_{11}}$	LDL$_{\ell_{11}}$	JSRC	JDSRC	MTSRC	MKL	**MWDL**	**JTLDL**	**HTLDL**
92.14	85.57	95.86	90.29	96.14	96.14	97.14	91.14	97.63	96.86	98.04

To better observe the effect of each component of the proposed method, the result of four systems are reported in Table (4.5):

(MWDL) unsupervised HTLDL with ℓ_{12}-norm in Eq. (4.18) (without task-driven part);

(LC-KSVD$_{\ell_{11}}$) HTLDL with ℓ_{11}-norm in Eq. (4.18);

(JTLDL) HTLDL with ℓ_{12}-norm in Eq. (4.18);

(HTLDL), HTLDL with Υ-norm in Eq. (4.18);

Table (4.5) demonstrates that the proposed framework with ℓ_{12}-norm achieves better accuracy comparing to the one with ℓ_{11}-norm. Concisely, sHTLDL$_{\ell_{12}}$ outperforms sHTLDL$_{\ell_{11}}$ with approximately 3% for fusion between left and right periocular, and with more than 1% for fusion between left periocular, right periocular, nose, and mouth and for the fusion between all modalities. Note that, we do not expect to have a significant correlation between nose and eyes, that is why, in most cases the fusion between nose, left and right periocular does not show any noticeable improvement. However, performance improves when mouth is added to the set of available modalities, and acts like a connection between nose and eyes for the task of face recognition. Also, HTLDL achieves the best result, since it can embed the information that we have about the task, here, the connection between each part of the face and itself.

Case III: Comparison with state-of-the-art fusion methods. To compare our proposed method with classification methods (like SVM) that originally are designed for single modality, we combine the classification results of individual modalities in the decision-fusion scheme. That is, the label of multi-modal data is determined either by the outcome of summation of modality-specific scores or in a majority voting scheme among the independent decisions from each modality. The former is shown with subscript *sum* and the latter by subscript *mv*, respectively in Table 4.6. Note that, decision-fusion by summation has the same effect as to change structural sparsity regularization Υ in Eq. (4.18) with ℓ_{11}-norm on multi-modal sparse codes $\boldsymbol{A}$. We know that ℓ_1-norm is blind to see any relation between variables. Similarly, ℓ_{11}-norm look at each modality independent of others. We report the classification accuracy for STDL and LDL in multimodal case when fusion is done using ℓ_{11} as $\text{STDL}_{\ell_{11}}$ and $\text{LDL}_{\ell_{11}}$ in Table (4.6).

Three other feature-fusion algorithms, the joint sparse representation classifier (JSRC) [156], joint dynamic sparse representation classifier (JDSRC) [202] and multimodal tree-structured sparse representation classification (MTSRC) [14] in Table (4.6). The dictionary for JSRC, JDSRC and MTSRC is fixed without training and they include all the training samples with all the training samples. JSRC [156] applies ℓ_{12} to enforce similar sparsity pattern among all different modalities at the space of sparse codes. JDSRC relaxes each multimodal input data to have the same sparsity pattern and lets it be reconstructed using different training samples. It applies joint sparsity on data of each class separately. MTSRC enforces a more generalized joint sparsity using a hierarchical structure regularization on each multimodal data.

Comparing Tables (4.3) and (4.6), we can see that decision-fusion by ℓ_{11} enhances the performance of LDL, and STDL with approximately 6% and 3%. However, MWDL, JTLDL and HTLDL that can do fusion at both feature-level and classifier-level outperforms decision-fusion competing methods with ℓ_{11}-norm. This outperformance is more significant for fusion of left and right periocular (around 3%) in Table (4.5). The reason lies in the fact that these modalities are highly correlated, and HTLDL learns multimodal dictionaries jointly, which

Table 4.7: Multiview face recognition results for the UMIST datasets

Views	JSRC	JDSRC	MTSRC	**MWDL**	**HTLDL**
2 Views	87.77	86.52	88.42	93.59	91.59
3 Views	99.51	98.96	99.63	100.0	100.0

results in a high recognition accuracy. The proposed HTLDL with 400 atoms achieves better performance than JSRC and JDSRC with 700 atoms. This superior results demonstrate that the dictionary learning in HTLDL is able to make discriminative and reconstructive dictionaries that can generate more discriminative sparse codes with less number of atoms.

This superior results demonstrate that the dictionary learning in HTLDL is able to make discriminative and reconstructive dictionaries that can generate more discriminative sparse codes with less number of atoms. Also, it is interesting that MWDL in general outperfoms JTLDL. This is due to applying multimodal weights with ℓ_{12} regularization on multimodal class-specific weights, $\|\boldsymbol{W}_c\|_{\ell_{12}}$ through optimization problem (4.16).

4.8.3 Multi-View Face Recognition on UMIST Dataset

UMIST face database consists of 564 cropped images of 20 persons with mixed race and gender [53]. Each person has different poses from profile to frontal views. The setup is unconstrained and faces may have pose variations within each view-ranges. We run multiview face recognition using UMIST by segmenting views of each person to M different view-range with equal number of images. In Fig. (4.3), the poses of a subject from UMIST is divided in $M = 3$ view-ranges. We report the performance of the MWDL for 2 and 3 views. Table (4.7) has the the results of 10-fold cross validation. The corresponding dictionary of each view has one normalized image from each subject in that view, $p = 20$.

We expect a higher correlation between view ranges that are close to each other. Hence, the design of the tree structure in HTLDL models the fusion and group characteristic among close views. HTLDL and MWDL achieve higher accuracy, with more than 4%, 5% in 2-views and around 1% for 3-views. Note that HTLDL for 2-view scenario converts to JTLDL.



Figure 4.3: Illustration of 3 view-range (modalities) in UMIST. Different poses of a subject from UMIST database. Each row is a view-range or modality for the subject.

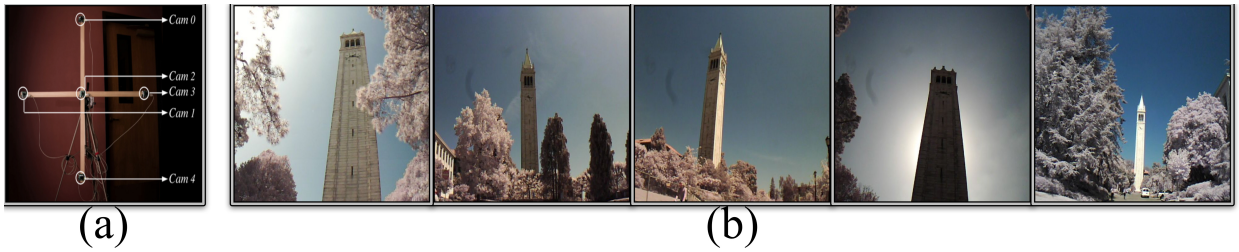


Figure 4.4: (a) Apparatus which instruments five camera sensors [129]. (b) Five "large baseline" images captured at different vantage points.

4.8.4 Multi-View Object Recognition

The BMW database consists of multiple-view images of 20 landmark buildings on the campus of University of California, Berkeley. For each building, 16 different vantage points have been selected to measure the 3-D appearance of the building. The apparatus for image acquisition incorporates five low-power CITRIC camera sensors [25] on a tripod, which can be triggered simultaneously. Figure 4.4 shows the configuration of the camera apparatus. The cameras on the periphery of the cross with a counter-clockwise naming convention are named Cam 0, Cam 1, Cam 4, Cam 3, and the center camera is called Cam 2. The BMW database has a total of 960 images.

First, we split the database into training and testing set. As the vantage points of each object are named numerically from 0 to 15, training set includes images from all the even number locations, and the testing set has the ones from the odd number locations.

Table 4.8: The recognition rate obtained for the "large-baseline" evaluation of BMW.

	LC-KSVD [77]				SDL [78]				sPCA [129]				TDL [104]			
BMW%	sift	surf	hog	ℓ_{11}	sift	surf	hog	ℓ_{11}	sift	surf	hog	ℓ_{11}	sift	surf	hog	ℓ_{11}
1 Cam	89.02	90.04	90.62	92.23	91.02	91.14	91.65	92.81	71.25	80.62	81.88	84.37	92.75	92.26	93.35	94.72
2 Cam	91.08	91.73	92.02	94.64	92.45	93.12	93.76	94.76	76.88	88.13	93.75	94.58	93.25	93.25	94.68	95.58

Table 4.9: Evaluation of MWDL, JTLDL and HTLDL for the recognition rate on "large-baseline" evaluation of BMW

$\ell_{11}\ell_{12}/\ell_{12}$	**MWDL**	**JTLDL**	**HTLDL**
1 Cam	**97.03/95.79**	**97.10/96.73**	98.24
2 Cam	**98.93/97.41**	**99.14/97.04**	100

Furthermore, since the main purpose of the experiment is to validate the recognition performance of using multiple-view testing images, we do not include the redundant multiple views in the training set.

The BMW set also provides three feature modalities: SIFT, SURF and CHoG [24] for all images. SURF and CHoG are variants of SIFT, which are more suitable for deployment on mobile camera platforms. These features are robust to the viewpoint variance, clutter and occlusion.

Like [130, 129] we use 8 images from all the even vantage points from the central camera for training model variables and test the method on the other cameras. Same as [129], we evaluate the recognition performance using one camera (*i.e.*, Cam 2) and two cameras (*i.e.*, Cam 1 and Cam 2). We compare JTLDL and HTLDL with state-of-the-art DL methods like SDL [78], LC-KSVD [77], TDL [104] and sparse PCA [129] in Table 4.8 and Table 4.9. We assign 8 atoms per class that leads to $p = 160$ atoms in all the settings. We report performance of other methods for single feature (SIFT, SURF and CHoG) and for multimodal setting under the ℓ_{11}. We report JTLDL and MWDL when all three are available under $\ell_{11}\ell_{12}/\ell_{12}$ which corresponds to $\lambda_2 > 0/\lambda_2 = 0$. We did not report JTLDL or HTLDL for

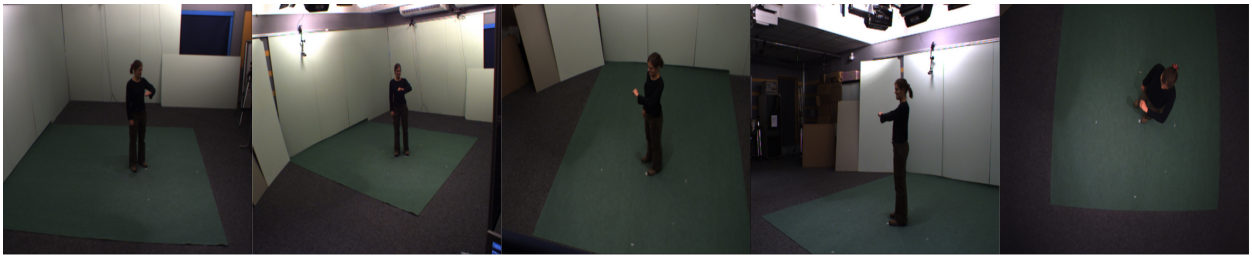


Figure 4.5: Check watch action sample from the IXMAS dataset [183]. Each action is viewed by 5 cameras ($M = 5$).

one feature because, it exploits correlation between different sources and when only one source is available it is equivalent to LC-KSVD.

The result shows that HTLDL, MWDL outperform other methods in one camera and two cameras setup by more than 2.3% and 3.35%, respectively. Also, SDL and LCKSVD cannot generate discriminative dictionaries when extra information from multiple cameras is present with the same number of atoms, and they need more atoms when the amount of training data increases. However, MWDL and JTLDL successfully generates discriminative dictionaries of the same size when more data is made available. Moreover, robust fusion using $\ell_{11}\ell_{12}$ achieves superior result and increases the accuracy more than 1.4% in both scenarios. For all the methods in Table 4.8, decision-level fusion using ℓ_{11} enhances the classification accuracy. However, our proposed fusion, which is designed to exploit feature-level fusion, outperforms decision-level fusion with other methods. This demonstrates the superiority of fusion at the feature level over fusion at the decision level.

4.8.5 Multi-View Action Recognition

The same action may seem quite different if viewed by various cameras from different angles. That is why action recognition across Camera views is a challenging task and an active area of research in computer vision. Human action recognition is an essential task to many real-world applications, such as visual surveillance, video retrieval, and human-computer interaction. Most of the methods assume all the actions are captured for training and testing from the same camera view, which is may not be the case most of the time. In practice, the same

Table 4.10: Multiview action recognition on the IXMAS (%)

Methods	Accuracy	Methods	Accuracy
STDL [104]	91.9	Wang et al. [180]	87.8
LCKSVD [77]	87.5	Tran et al. [177]	80.2
JSRC	93.6	LDL [195]	88.4
MWDL	96.1	**HTLDL**	97.2
JTLDL$_{\ell_{12}}$	92.5	**JTLDL**$_{\ell_{11}\ell_{12}}$	95.6

action can be hard to recognize from a different angle, because the magnitude of variations of action characteristics, which discriminates one action from the others, may be even smaller than the variation originated by the change of viewpoints.

Dataset. We test our approach based on the IXMAS [183] multiple views action dataset. It includes 11 categories of daily actions: check watch, cross arms, scratch head, sit down, get up, turn around, walk, wave, punch, kick, and pick up. Ten actors performed each action three times. There are five cameras, four side views and one top view that are considered as modalities in this experiment. A multimodal sample of the IXMAS dataset is shown in Fig. (4.5). Following [183, 177, 180] leave-one-actor-out cross validation is performed and samples from all five views are used for training and testing. We extract dense trajectories from all samples using the code provided by [180]. Then, following [55] using k-means we made a code-book of 2000 words from a random subset of all the trajectories.

We consider 4 atoms per class, which leads to 44 atoms per view. However, the performance of JSRC in Table 4.10 is reported for a view-specific dictionary with all the training samples, *i.e.* $p = 297$. Also, the method in [180] besides dense trajectories, exploits three descriptors of motion boundary histograms (MBH), histogram of flow (HoF) and histogram of gradients (HoG), while we only use dense trajectories in HTLDL, JTLDL and MWDL.

The results show that MWDL outperforms competing methods more than 3% and enhances LDL with 7.0%. The performance of JTLDL is reported once without ℓ_{11} as JTLDL$_{\ell_{12}}$ and once as the joint $\ell_{11}\ell_{12}$. The results show that JTLDL that exploits feature

fusion and classifier fusion outperforms other methods. Specially, JTLDL enhances LCKSVD by more than 4.0%. Also, JTLDL$_{\ell_{11}\ell_{12}}$ gets superior results compared to the JTLDL$_{\ell_{12}}$.

Table 4.10 demonstrates that MWDL and JTLDL$_{\ell_{12}}$ outperforms JSRC and LDL. It shows that feature-fusion by applying structured norm in space of sparse codes generates more discriminative features. Plus, JTLDL$_{\ell_{12}}$, MWDL, HTLDL using only dense trajectories outperforms Wang et al. [180] with more than 6.7%. This demonstrates that the sparse codes generated in a multi-modal multi-task data-driven scheme are more distinctive than the hand-designed features of MBH, HoF and HoG. In addition, HTLDL with tree-structure achieves the best performance with 1.1% higher classification accuracy than the second best approach MWDL.

4.9 Conclusion

Multiview object and action recognition in the sparsity scheme was studied and a method proposed to learn a supervised dictionary and classifier while obtaining multimodal sparse representations of each sample using a joint sparsity model. The imposed joint sparsity enabled the algorithm to fuse information at the feature level by forcing each modality's sparse codes to have a similar structure within each class and at the decision level by augmenting the classifier decisions. We investigate the stability issue of fusion using ℓ_{12} regularization and provide an exact solution for robust feature fusion using $\ell_{11}\ell_{12}$ while simultaneously learning dictionary and classifier. JTLDL is able to learn reconstructive and discriminative dictionaries because it learns modality-based dictionaries such that the same subset of dictionary items from different modalities represent each certain class and also promotes learning dictionaries which are incoherent between classes in each modality. The experimental results demonstrate that the proposed method outperforms state-of-the-art dictionary learning methods in the challenging scenarios of multi-view object and multi-view action recognition.

In this Chapter, we presented a new method for learning multimodal dictionaries while multimodal sparse representations are forced to share the same sparsity patterns at the atom

level of modality-based dictionaries using $\ell_{1,2}$ regularization. The imposed joint sparsity model enabled algorithm to fuse information in feature-level and it can be easily extended to include augmenting the decision of modalities. To obtain discriminative dictionary suitable for classification task in each feature modality, relationship between dictionary atoms and class labels is defined as a weight matrix. The solution to the optimization problem is designed to jointly solve multimodal dictionaries, multimodal weights and multimodal sparse codes. The weights are updated adaptively in order to decrease the correlation between atoms of the modality-base dictionary, while sparse representation of different modalities of same class are obtained with high correlation. The solution provides a compact with small number of atoms dictionary in each modality that is suitable for discrimination task. The experimental results demonstrated that the proposed method outperforms state-of-the-art dictionary learning methods in most challenging scenarios.

Supervised dictionary learning can be divided into three categories based on the predefined relation between dictionary elements and class labels in the data:

- all-against-all or shared dictionary learning which each atom may represent all classes,
- one-against-all or class-specific dictionary learning that each dictionary element is assigned to only a single class,
- hybrid dictionary learning that is a combination of all-against-all and one-against-all.

In the first category, a dictionary is obtained in a data-driven scheme to be able to reconstruction input data independent of its label, while simultaneously, a discriminative regularization enforces the generated decomposition coefficients in the space of sparse codes to be discriminative between classes [104, 206, 77, 111, 71]. Often, these methods can be summarized as learning a dictionary shared among all classes and a classifier over sparse representation. On a bright side, these methods provide a compact dictionary with a small number of atoms and as a result, the estimation of the sparse representation would be faster in the testing phase. However, no class-specific representation residuals can be used, and there is no guarantee about the relation between atoms and classes.

In the one-against-all dictionary learning, for each class, a dictionary is learned which is optimal to reconstruct that specific class (it does not have any information about other classes). In this scenario, we know the label of each atom inside the dictionary. However, the obtained class-specific dictionaries are blind to information about other classes [107, 196, 197, 147]. Since each dictionary only captures information of one specific class, the classification can be done using the reconstruction error of each class-specific dictionary. However, the methods in this scenario do not consider the possible correlation between the classes and they do not bother themselves to learn dictionaries based on the information that discriminates classes from each other. Furthermore, to get a good classification accuracy, these methods need to have a large number of atoms for each class.

Recently, some studies are done on a combination of one-against-all and all-against-all to come up with a method that has advantages of the both schemes [158, 215, 89]. The dictionary learning method in this scheme is not trivial. The algorithm should be designed in a way that the shared and private parts of the dictionary can capture the underlying information in the data. In both of the two schemes, the relation between class labels and each atom is predefined. However, in our method, this connection is updated in each iteration, while the link is enforced to be supported from all modalities.

Chapter 5

Conclusions and Future Work

This Chapter summarizes the study observations and discusses possible improvements for future research. We briefly go over the key concepts.

- **Modality.** In various scenarios, information about the same phenomenon or event can be obtained from different types of detectors, at various conditions or in multiple experiments or subjects. Each such acquisition framework is a modality of the phenomena. Due to the complex and rich relation between the modalities of multimodal phenomena, a single modality cannot describe the event of interest. The fact that several modalities report on the same event introduces new challenges that are beyond degrees of freedom related to exploiting each modality separately.

- **Representation Learning.** Our intuition is to find a proper representation for the multimodal data. Such data is seen from multiple modalities (sensors). We intend to extract a discriminative representation of the multimodal data that leads to finding easier its essential characteristics in the subsequent analysis step, *e.g.*, regression and classification. In other words, using sensor fusion techniques, we obtain a discriminative representation for the multimodal data so that a better classification performance can be achieved compared to the case where individual modalities are utilized.

- **Key Factor in Multimodal Representation.** In the context of multimodal data analysis, the term "modality configuration" describes the correlation (relation) between different modalities that acquire information from the phenomenon. This information is vital for our goal, because it determines how strong is the coupling structure between modalities. That is why, we believe the key to make a representation for multimodal data is to figure out the fusion structure between modalities, a task which usually called *fusion architecture*. We use this modality configuration as a high-order prior information and formulate the a priori known coupling between modalities in the particular signal representation of sparse coding.

- **Why Regularization.** The method should be designed so that it can successfully generalize the unseen and new data; that is to make sure that the model does not suffer from the overfitting problem. It can occur due to a large number of basis or a small number of training samples. The prior information about the data or the form of the solution leads to the concept of regularization that show promising to deal with the overfitting problem.

- **Why Sparsity Regularization.** A simple a priori model is to assume the solution to be sparse. This bias towards sparsity can emerge in two scenarios: First, we know that the problem at hand has a sparse solution, or in the absence of sparsity prior information, our interest lies in seeking a simple reasoning for the task that is easy to interpret and has a low processing complexity. This is known as sparsity and can be assumed as selecting a small number of parameters to solve the problems.

- **Motivation of Dictionary Learning.** the sparse coding objective function only cares about reconstructing the input well, and does not attempt to make sparse codes useful as input for any particular task; hence, dictionary should be modified through online stochastic gradient descent to make sparse codes more useful for prediction.

- **Our Framework:** We designed a framework based on matrix factorization that connects the data modalities through a latent factors space. We formulate a unifying

framework that extracts the common structure of all the modalities, while at the same time can map one representation in an alternative space. In our latent factor model, the correlation between modalities directly depends on the modality configuration that we found before. Intuitively, we say that modalities have the same underlying semantics in the latent space. We target learning cross-modality correlations while at the same time try preventing unwanted co-adaptations between data modalities. This is critical to make the representation robust to missing signals and signal corruption.

Our future research lies in three main aspects.

- So far, we only consider supervised multimodal dictionary learning when grouping between modalities are either partitions of the set $\{1, \ldots, p\}$, and they do not overlap, or, they are hierarchically related in a tree-structure. We intend to extend the proposed methods by relaxing this constraint and allow the groups to be hierarchically correlated with a tree structure that is estimated from the data. In other words, instead of hand-coded tree-structure, the optimization solution provides the tree-structure that fits the data.

- We limit ourselves to the groups that are hierarchically related, *i.e.* the groups are intersection closed. In future work, we intend to relax this constraint to union closed grouping and evaluate its performance for various recognition tasks.

- We limit regularization over dictionary atoms to only consider unit-norm ball. In our future work, we extend the proposed optimization to apply norm-based regularization on rows and/or columns of dictionary so that a better more accurate structure can be obtained.

Bibliography

[1] Aharon, M. and Elad, M. (2008). Sparse and redundant modeling of image content using an image-signature-dictionary. *SIAM Journal on Imaging Sciences*, 1(3):228–247.

[2] Aharon, M., Elad, M., and Bruckstein, A. (2006). k -svd: An algorithm for designing overcomplete dictionaries for sparse representation. *Signal Processing, IEEE Transactions on*, 54(11):4311–4322.

[3] Ahmed, N., Natarajan, T., and Rao, K. (1974). Discrete cosine transfom. *IEEE Transactions on Computers*, 23(1):90–93.

[4] Alt, N., Hinterstoisser, S., and Navab, N. (2010). Rapid selection of reliable templates for visual tracking. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 1355–1362.

[5] Avidan, S. (2004). Support vector tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(8):1064–1072.

[6] Babagholami-Mohamadabadi, B., Jourabloo, A., Zarghami, A., and Baghshah, M. S. (2013a). Supervised dictionary learning using distance dependent indian buffet process. In *2013 IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6. IEEE.

[7] Babagholami-Mohamadabadi, B., Jourabloo, A., Zarghami, A., and Kasaei, S. (2014a). A bayesian framework for sparse representation-based 3-d human pose estimation. volume 21, pages 297–300. IEEE.

[8] Babagholami-Mohamadabadi, B., Jourabloo, A., Zolfaghari, M., and Shalmani, M. T. M. (2013b). Bayesian supervised dictionary learningâĂŐ. In *UAI Application Workshops*, pages 11–19. Citeseer.

[9] Babagholami-Mohamadabadi, B., Roostaiyan, S. M., Zarghami, A., and Baghshah, M. S. (2014b). Multi-modal distance metric learning: Abayesian non-parametric approach. In *European Conference on Computer Vision*, pages 63–77. Springer.

[10] Babagholami-Mohamadabadi, B., Zarghami, A., Zolfaghari, M., and Baghshah, M. S. (2013c). Pssdl: Probabilistic semi-supervised dictionary learning. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 192–207. Springer.

[11] Babenko, B., Yang, M. H., and Belongie, S. (2011). Robust object tracking with online multiple instance learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8):1619–1632.

[12] Bach, F., Jenatton, R., Mairal, J., and Obozinski, G. (2012). Optimization with sparsity-inducing penalties. *Found. Trends Mach. Learn.*, 4(1):1–106.

[13] Bahrampour, S., Nasrabadi, N. M., Ray, A., and Jenkins, W. K. (2016). Multimodal Task-Driven Dictionary Learning for Image Classification. *IEEE Transactions on Image Processing*, 25:24–38.

[14] Bahrampour, S., Ray, A., Nasrabadi, N. M., and Jenkins, K. W. (2014). Quality-based multimodal classification using tree-structured sparsity. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[15] Bao, C., Wu, Y., Ling, H., and Ji, H. (2012). Real time robust l1 tracker using accelerated proximal gradient approach. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1830–1837.

[16] Bishop, C. M. (2006). Pattern recognition. *Machine Learning*, 128:1–58.

[17] Blumensath, T. and Davies, M. E. (2009). Iterative hard thresholding for compressed sensing. *Applied and Computational Harmonic Analysis*, 27(3):265 – 274.

[18] Bottou, L. (1998). On-line learning in neural networks. chapter On-line Learning and Stochastic Approximations, pages 9–42. Cambridge University Press, New York, NY, USA.

[19] Bousquet, O. and Bottou, L. (2008). The tradeoffs of large scale learning. In Platt, J. C., Koller, D., Singer, Y., and Roweis, S. T., editors, *Advances in Neural Information Processing Systems 20*, pages 161–168. Curran Associates, Inc.

[20] Bradley , D. and Bagnell, J. A. D. (2008). Differentiable sparse coding. In *Proceedings of Neural Information Processing Systems 22*.

[21] Candès, E. J., Wakin, M. B., and Boyd, S. P. (2008). Enhancing sparsity by reweighted âĎŞ 1 minimization. *Journal of Fourier Analysis and Applications*, 14(5):877–905.

[22] CandÃĺs, E. J. and Donoho, D. L. (2004). New tight frames of curvelets and optimal representations of objects with piecewise c2 singularities. *Communications on Pure and Applied Mathematics*, 57(2):219–266.

[23] Cazzanti, L., Gupta, M. R., and Koppal, A. J. (2008). Generative models for similarity-based classification. *Pattern Recognition*, 41(7):2289 – 2297.

[24] Chandrasekhar, V., Takacs, G., Chen, D., Tsai, S., Grzeszczuk, R., and Girod, B. (2009). Chog: Compressed histogram of gradients a low bit-rate feature descriptor. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 2504–2511.

[25] Chen, P., Ahammad, P., Boyer, C., Huang, S.-I., Lin, L., Lobaton, E., Meingast, M., Oh, S., Wang, S., Yan, P., et al. (2008). Citric: A low-bandwidth wireless camera network platform. In *Distributed smart cameras, 2008. ICDSC 2008. Second ACM/IEEE international conference on*, pages 1–10. IEEE.

[26] Chen, S. S., Donoho, D. L., and Saunders, M. A. (2001). Atomic decomposition by basis pursuit. *SIAM Review*, 43(1):129–159.

[27] Cheng, Y., Zhao, X., Cai, R., Li, Z., Huang, K., and Rui, Y. (2016). Semi-supervised multimodal deep learning for rgb-d object recognition. IJCAI'16, pages 3345–3351. AAAI Press.

[28] Collins, R. T., Liu, Y., and Leordeanu, M. (2005). Online selection of discriminative tracking features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1631–1643.

[29] Comaniciu, D., Ramesh, V., and Meer, P. (2003). Kernel-based object tracking. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(5):564–577.

[30] Danelljan, M., Khan, F. S., Felsberg, M., and v. d. Weijer, J. (2014). Adaptive color attributes for real-time visual tracking. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 1090–1097.

[31] Davoudi, H., Taalimi, A., and Fatemizadeh, E. (2009). Extracting activated regions of fmri data using unsupervised learning. In *Neural Networks, 2009. IJCNN 2009. International Joint Conference on*, pages 641–645.

[32] Di Cataldo, S., Bottino, A., Islam, I. U., Vieira, T. F., and Ficarra, E. (2014). Subclass discriminant analysis of morphological and textural features for hep-2 staining pattern classification. *Pattern Recognition*, 47(7):2389–2399.

[33] Dollár, P., Tu, Z., Perona, P., and Belongie, S. (2009). Integral channel features. *BMVC*, 2(3):5.

[34] Donoho, D. (2006). Compressed sensing. *IEEE Transactions on Information Theory*, 52(4):1289 –1306.

[35] Efron, B., Hastie, T., Johnstone, I., and Tibshirani, R. (2004). Least angle regression. *Ann. Statist.*, 32(2):407–499.

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

[37] Ensafi, S., Lu, S., Kassim, A. A., and Tan, C. L. (2014a). Automatic cad system for hep-2 cell image classification. In *Pattern Recognition (ICPR), 2014 22nd International Conference on*, pages 3321–3326. IEEE.

[38] Ensafi, S., Lu, S., Kassim, A. A., and Tan, C. L. (2014b). A bag of words based approach for classification of hep-2 cell images. In *Pattern Recognition Techniques for Indirect Immunofluorescence Images (I3A), 2014 1st Workshop on*, pages 29–32. IEEE.

[39] Ensafi, S., Lu, S., Kassim, A. A., and Tan, C. L. (2015). Sparse non-parametric bayesian model for hep-2 cell image classification. In *Biomedical Imaging (ISBI), 2015 IEEE 12th International Symposium on*, pages 679–682.

[40] Faraki, M., Harandi, M. T., Wiliem, A., and Lovell, B. C. (2014). Fisher tensors for classifying human epithelial cells. *Pattern Recognition*, 47(7):2348–2359.

[41] Felzenszwalb, P., Girshick, R., McAllester, D., and Ramanan, D. (2010). Object detection with discriminatively trained part-based models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(9):1627–1645.

[42] Foggia, P., Percannella, G., Saggese, A., and Vento, M. (2014). Pattern recognition in stained hep-2 cells: Where are we now? *Pattern Recognition*, 47(7):2305–2314.

[43] Foggia, P., Percannella, G., Soda, P., and Vento, M. (2013). Benchmarking hep-2 cells classification methods. *Medical Imaging, IEEE Transactions on*, 32(10):1878–1889.

[44] Friedman, J. (1997). On bias, variance, 0/1–loss, and the curse-of-dimensionality. *Data Mining and Knowledge Discovery*, 1(1):55–77.

[45] Gholami, B. and Hajisami, A. (2016). Kernel auto-encoder for semi-supervised hashing. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–8. IEEE.

[46] Ghosh, S. (2015). Challenges in deep learning for multimodal applications. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, pages 611–615. ACM.

[47] Ghosh, S. and Chaudhary, V. (2012). Feature analysis for automatic classification of hep-2 florescence patterns: Computer-aided diagnosis of auto-immune diseases. In *Pattern Recognition (ICPR), 2012 21st International Conference on*, pages 174–177. IEEE.

[48] Gillis, N. (2012). Sparse and unique nonnegative matrix factorization through data preprocessing. *J. Mach. Learn. Res.*, 13(1):3349–3386.

[49] Golub, G. H., Hansen, P. C., and O'Leary, D. P. (1999). Tikhonov regularization and total least squares. *SIAM Journal on Matrix Analysis and Applications*, 21(1):185–194.

[50] González-Buitrago, J. M. and González, C. (2006). Present and future of the autoimmunity laboratory. *Clinica chimica acta*, 365(1):50–57.

[51] Grabner, H., Grabner, M., and Bischof, H. (2006). Real-time tracking via on-line boosting. In *Proc. BMVC*, pages 6.1–6.10. doi:10.5244/C.20.6.

[52] Gragnaniello, D., Sansone, C., and Verdoliva, L. (2014). Biologically-inspired dense local descriptor for indirect immunofluorescence image classification. In *Pattern Recognition Techniques for Indirect Immunofluorescence Images (I3A), 2014 1st Workshop on*, pages 1–5.

[53] Graham, D. B. and Allinson, N. M. (1997). Face recognition using virtual parametric eigenspace signatures. In *Image Processing and Its Applications, 1997., Sixth International Conference on*, volume 1, pages 106–110 vol.1.

[54] Guo, R., Liu, L., Wang, W., Taalimi, A., Zhang, C., and Qi, H. (2016). Deep tree-structured face: A unified representation for multi-task facial biometrics. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–8.

[55] Gupta, A., Shafaei, A., Little, J. J., and Woodham, R. J. (2014). Unlabelled 3d motion examples improve cross-view action recognition. In *BMVC*, volume 1, page 3.

[56] Guyon, I. and Elisseeff, A. (2003). An introduction to variable and feature selection. *J. Mach. Learn. Res.*, 3:1157–1182.

[57] Han, X.-H., Wang, J., Xu, G., and Chen, Y.-W. (2014). High-order statistics of microtexton for hep-2 staining pattern classification. *Biomedical Engineering, IEEE Transactions on*, 61(8):2223–2234.

[58] Hare, S., Saffari, A., and Torr, P. (2011). Struck: Structured output tracking with kernels. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 263–270.

[59] Henriques, J., Caseiro, R., Martins, P., and Batista, J. (2015). High-speed tracking with kernelized correlation filters. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 37(3):583–596.

[60] Henriques, J. F., Caseiro, R., Martins, P., and Batista, J. (2012). *Computer Vision – ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part IV*, chapter Exploiting the Circulant Structure of Tracking-by-Detection with Kernels, pages 702–715. Springer Berlin Heidelberg, Berlin, Heidelberg.

[61] Hong, Z., Chen, Z., Wang, C., Mei, X., Prokhorov, D., and Tao, D. (2015). Multi-store tracker (muster): A cognitive psychology inspired approach to object tracking. In *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, pages 749–758.

[62] Hong, Z., Mei, X., Prokhorov, D., and Tao, D. (2013). Tracking via robust multi-task multi-view joint sparse representation. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 649–656.

[63] Hosseini, M.-P. (2015). Proposing a new artificial intelligent system for automatic detection of epileptic seizures. *Journal of Neurological Disorders*, 3(4):DOI–10.

[64] Hosseini, M.-P., Hajisami, A., and Pompili, D. (2016a). Real-time epileptic seizure detection from eeg signals via random subspace ensemble learning. In *Autonomic Computing (ICAC), 2016 IEEE International Conference on*, pages 209–218. IEEE.

[65] Hosseini, M.-P., Nazem-Zadeh, M. R., Pompili, D., Jafari-Khouzani, K., Elisevich, K., and Soltanian-Zadeh, H. (2015). Automatic and manual segmentation of hippocampus in epileptic patients mri. In *6th annual New York Medical Imaging Informatics Symposium (NYMIIS)*. Staten Island University Hospital, NY, USA.

[66] Hosseini, M.-P., Nazem-Zadeh, M. R., Pompili, D., and Soltanian-Zadeh, H. (2014). Statistical validation of automatic methods for hippocampus segmentation in mr images of epileptic patients. In *Engineering in Medicine and Biology Society (EMBC), 2014 36th Annual International Conference of the IEEE*, pages 4707–4710. IEEE.

[67] Hosseini, M.-P., Soltanian-Zadeh, H., Elisevich, K., and Pompili, D. (2016b). Cloud-based deep learning of big eeg data for epileptic seizure prediction. In *IEEE Global Conference on Signal and Information Processing (GlobalSIP)*. IEEE.

[68] Hosseini, M.-P., Tran, T., Pompili, D., Elisevich, K., and Soltanian-Zadeh, H. (2017). Deep learning with edge computing for localization of epileptogenicity using multimodal rs-fmri and eeg big data. In *IEEE International Conference on Autonomic Computing*. IEEE.

[69] Hu, W., Li, W., Zhang, X., and Maybank, S. (2015). Single and multiple object tracking using a multi-feature joint sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(4):816–833.

[70] Huang, K. and Aviyente, S. (2006). Sparse representation for signal classification. In *In Adv. NIPS*.

[71] Jacob, L., Obozinski, G., and Vert, J.-P. (2009). Group lasso with overlap and graph lasso. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pages 433–440, New York, NY, USA. ACM.

[72] Jenatton, R., Mairal, J., Bach, F. R., and Obozinski, G. R. (2010). Proximal methods for sparse hierarchical dictionary learning. In *International Conference on Machine Learning*, pages 487–494.

[73] Jenatton, R., Mairal, J., Obozinski, G., and Bach, F. (2011). Proximal methods for hierarchical sparse coding. *J. Mach. Learn. Res.*, 12:2297–2334.

[74] Jenatton, R., Obozinski, G., and Bach, F. (2009). Structured Sparse Principal Component Analysis. *ArXiv e-prints*.

[75] Jepson, A. D., Fleet, D. J., and El-Maraghi, T. F. (2003). Robust online appearance models for visual tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(10):1296–1311.

[76] Jia, X., Lu, H., and Yang, M. H. (2012). Visual tracking via adaptive structural local sparse appearance model. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1822–1829.

[77] Jiang, Z., Lin, Z., and Davis, L. (2013). Label consistent k-svd: Learning a discriminative dictionary for recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(11):2651–2664.

[78] Jiang, Z., Zhang, G., and Davis, L. (2012). Submodular dictionary learning for sparse coding. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3418–3425.

[79] Kalal, Z., Mikolajczyk, K., and Matas, J. (2012). Tracking-learning-detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(7):1409–1422.

[80] Kastaniotis, D., Theodorakopoulos, I., Economou, G., and Fotopoulos, S. (2013). Hep-2 cells classification using locally aggregated features mapped in the dissimilarity space. In *Bioinformatics and Bioengineering (BIBE), 2013 IEEE 13th International Conference on*, pages 1–4.

[81] Khorsandi, R. and Abdel-Mottaleb, M. (2013a). Gender classification using facial images and basis pursuit. In *15th International Conference on Computer Analysis of Images and Patterns*.

[82] Khorsandi, R. and Abdel-Mottaleb, M. (2013b). Gender classification using facial images and basis pursuit. In *International Conference on Computer Analysis of Images and Patterns*, pages 294–301. Springer Berlin Heidelberg.

[83] Khorsandi, R. and Abdel-Mottaleb, M. (2015). Classification based on weighted sparse representation using smoothed l0 norm with non-negative coefficients. In *2015 IEEE International Conference on Image Processing (ICIP)*, pages 3131–3135.

[84] Khorsandi, R., Taalimi, A., and Abdel-Mottaleb, M. (2015a). Robust biometrics recognition using joint weighted dictionary learning and smoothed l0 norm. In *Biometrics Theory, Applications and Systems (BTAS), 2015 IEEE 7th International Conference on*, pages 1–6.

[85] Khorsandi, R., Taalimi, A., Abdel-Mottaleb, M., and Qi, H. (2015b). Joint weighted dictionary learning and classifier training for robust biometric recognition. In *2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pages 1307–1311.

[86] Kim, M. (2011). Discriminative semisupervised learning of dynamical systems for motion estimation. *Pattern Recognition*, 44(10-11):2325–2333. Semi-Supervised Learning for Visual Content Analysis and Understanding.

[87] Kim, S. and Xing, E. P. (2010). Tree-guided group lasso for multi-task regression with structured sparsity. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 543–550.

[88] Klausne, A., Tengg, A., and Rinner, B. (2007). Vehicle classification on multi-sensor smart cameras using feature- and decision-fusion. In *Distributed Smart Cameras, 2007. ICDSC '07. First ACM/IEEE International Conference on*, pages 67–74.

[89] Kong, S. and Wang, D. (2012). A dictionary learning approach for classification: separating the particularity and the commonality. In *Computer Vision–ECCV 2012*, pages 186–199. Springer.

[90] Kong, S. and Wang, D. (2013). Learning individual-specific dictionaries with fused multiple features for face recognition. In *Automatic Face and Gesture Recognition (FG), 2013 10th IEEE International Conference and Workshops on*, pages 1–8.

[91] Koren, Y., Bell, R., and Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37.

[92] Kristan, M., Pflugfelder, R., Leonardis, A., Matas, J., ÄŇehovin, L., Nebehay, G., VojÃŋÅŹ, T., and al, e. (2015). The visual object tracking vot2014 challenge results. In Agapito, L., Bronstein, M. M., and Rother, C., editors, *Computer Vision - ECCV 2014 Workshops*, volume 8926 of *Lecture Notes in Computer Science*, pages 191–217. Springer International Publishing.

[93] Kuncheva, L. (2002). A theoretical study on six classifier fusion strategies. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(2):281–286.

[94] Kwon, J. and Lee, K. M. (2010). Visual tracking decomposition. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 1269–1276.

[95] Lan, X., Ma, A. J., and Yuen, P. C. (2014). Multi-cue visual tracking using robust feature-level fusion based on joint sparse representation. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 1194–1201.

[96] Larsen, A., Vestergaard, J., and Larsen, R. (2014). Hep-2 cell classification using shape index histograms with donut-shaped spatial pooling. *Medical Imaging, IEEE Transactions on*, 33(7):1573–1580.

[97] Lazebnik, S., Schmid, C., and Ponce, J. (2006). Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Computer Vision and*

Pattern Recognition, 2006 IEEE Computer Society Conference on, volume 2, pages 2169–2178. IEEE.

[98] Li, J., Tseng, K.-K., Hsieh, Z. Y., Yang, C. W., and Huang, H.-N. (2014). Staining pattern classification of antinuclear autoantibodies based on block segmentation in indirect immunofluorescence images. *PloS one*, 9(12):e113132.

[99] Lian, W., Henao, R., Rao, V., Lucas, J., and Carin, L. (2015a). A multitask point process predictive model. In *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, pages 2030–2038.

[100] Lian, W., Rai, P., Salazar, E., and Carin, L. (2015b). Integrating features and similarities: Flexible models for heterogeneous multiview data. In *AAAI*, pages 2757–2763.

[101] Liu, L., Rahimpour, A., Taalimi, A., and Qi, H. (2017). End-to-end binary representation learning via direct binary embedding. In *2017 IEEE International Conference on Image Processing (ICIP)*.

[102] Lucas, B. D., Kanade, T., et al. (1981). An iterative image registration technique with an application to stereo vision. In *IJCAI*, volume 81, pages 674–679.

[103] Mairal, J. (2010). *Sparse coding for machine learning, image processing and computer vision*. PhD thesis. ThÃĺse de doctorat dirigÃľe par Bach, Francis et Ponce, Jean MathÃľmatiques appliquÃľes Cachan, Ecole normale supÃľrieure 2010.

[104] Mairal, J., Bach, F., and Ponce, J. (2012). Task-driven dictionary learning. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(4):791–804.

[105] Mairal, J., Bach, F., Ponce, J., and Sapiro, G. (2009a). Online dictionary learning for sparse coding. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pages 689–696, New York, NY, USA. ACM.

[106] Mairal, J., Bach, F., Ponce, J., and Sapiro, G. (2010). Online learning for matrix factorization and sparse coding. *J. Mach. Learn. Res.*, 11:19–60.

[107] Mairal, J., Bach, F., Ponce, J., Sapiro, G., and Zisserman, A. (2008a). Discriminative learned dictionaries for local image analysis. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8.

[108] Mairal, J., Bach, F., Ponce, J., Sapiro, G., and Zisserman, A. (2009b). Non-local sparse models for image restoration. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 2272–2279.

[109] Mairal, J., Elad, M., and Sapiro, G. (2008b). Sparse representation for color image restoration. *Image Processing, IEEE Transactions on*, 17(1):53–69.

[110] Mairal, J., Leordeanu, M., Bach, F., Hebert, M., and Ponce, J. (2008c). *Computer Vision – ECCV 2008: 10th European Conference on Computer Vision, Marseille, France, October 12-18, 2008, Proceedings, Part III*, chapter Discriminative Sparse Image Models for Class-Specific Edge Detection and Image Interpretation, pages 43–56. Springer Berlin Heidelberg, Berlin, Heidelberg.

[111] Mairal, J., Ponce, J., Sapiro, G., Zisserman, A., and Bach, F. R. (2009c). Supervised dictionary learning. In Koller, D., Schuurmans, D., Bengio, Y., and Bottou, L., editors, *Advances in Neural Information Processing Systems 21*, pages 1033–1040. Curran Associates, Inc.

[112] Mallat, S. (2008). *A Wavelet Tour of Signal Processing, Third Edition: The Sparse Way*. Academic Press, 3rd edition.

[113] Manandhar, A., Morton Jr, K. D., Collins, L. M., and Torrione, P. A. (2015). A nonparametric bayesian approach to multiple instance learning. *International Journal of Pattern Recognition and Artificial Intelligence*, 29(03):1551001.

[114] Manivannan, S., Li, W., Akbar, S., Wang, R., Zhang, J., and McKenna, S. J. (2014). Hep-2 cell classification using multi-resolution local patterns and ensemble svms. In *Pattern Recognition Techniques for Indirect Immunofluorescence Images (I3A), 2014 1st Workshop on*, pages 37–40. IEEE.

[115] Manivannan, S., Li, W., Akbar, S., Wang, R., Zhang, J., and McKenna, S. J. (2016). An automated pattern recognition system for classifying indirect immunofluorescence images of hep-2 cells and specimens. *Pattern Recognition*, 51:12 – 26.

[116] MARTINEZ, A. M. (1998). The ar face database. *CVC Technical Report*, 24.

[117] Mei, X. and Ling, H. (2009). Robust visual tracking using l1 minimization. In *2009 IEEE 12th International Conference on Computer Vision*, pages 1436–1443.

[118] Mei, X. and Ling, H. (2011). Robust visual tracking and vehicle classification via sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(11):2259–2272.

[119] Mei, X., Ling, H., Wu, Y., Blasch, E., and Bai, L. (2011). Minimum error bounded efficient l1 tracker with occlusion detection. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1257–1264.

[120] Mian, I., Bradwell, A. R., and Olson, A. J. (1991). Structure, function and properties of antibody binding sites. *Journal of Molecular Biology*, 217(1):133 – 151.

[121] Minaee, S. and Abdolrashidi, A. (2015). Highly accurate palmprint recognition using statistical and wavelet features. In *Signal Processing and Signal Processing Education Workshop (SP/SPE), 2015 IEEE*, pages 31–36.

[122] Minaee, S., Abdolrashidi, A., and Wang, Y. (2015a). Iris recognition using scattering transform and textural features. In *Signal Processing and Signal Processing Education Workshop (SP/SPE), 2015 IEEE*, pages 37–42.

[123] Minaee, S., Abdolrashidi, A., and Wang, Y. (2015b). Screen content image segmentation using sparse-smooth decomposition. In *2015 49th Asilomar Conference on Signals, Systems and Computers*, pages 1202–1206.

[124] Minaee, S., Fotouhi, M., and Khalaj, B. H. (2014). A geometric approach to fully automatic chromosome segmentation. In *Signal Processing in Medicine and Biology Symposium (SPMB), 2014 IEEE*, pages 1–6.

[125] Minaee, S. and Wang, Y. (2016a). Screen content image segmentation using robust regression and sparse decomposition. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, PP(99):1–12.

[126] Minaee, S. and Wang, Y. (2016b). Screen content image segmentation using sparse decomposition and total variation minimization. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 3882–3886.

[127] Moghaddam, B. and Yang, M.-H. (2000). Gender classification with support vector machines. In *Fourth IEEE International Conference on Automatic Face and Gesture Recognition*.

[128] Mousavi, H. S., Srinivas, U., Monga, V., Suo, Y., Dao, M., and Tran, T. D. (2014). Multi-task image classification via collaborative, hierarchical spike-and-slab priors. In *Image Processing (ICIP), 2014 IEEE International Conference on*, pages 4236–4240. IEEE.

[129] Naikal, N., Yang, A., and Sastry, S. (2010). Towards an efficient distributed object recognition system in wireless smart camera networks. In *Information Fusion (FUSION), 2010 13th Conference on*, pages 1–8.

[130] Naikal, N., Yang, A., and Sastry, S. (2011). Informative feature selection for object recognition via sparse pca. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 818–825.

[131] Natarajan, B. K. (1995). Sparse approximate solutions to linear systems. *SIAM Journal on Computing*, 24(2):227–234.

[132] Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., and Ng, A. Y. (2011). Multimodal deep learning. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 689–696.

[133] Nguyen, N., Nasrabadi, N., and Tran, T. (2011). Robust multi-sensor classification via joint sparse representation. In *Information Fusion (FUSION), 2011 Proceedings of the 14th International Conference on*, pages 1–8.

[134] Nosaka, R. and Fukui, K. (2014). Hep-2 cell classification using rotation invariant co-occurrence among local binary patterns. *Pattern Recognition*, 47(7):2428–2436.

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

[136] Parikh, N. and Boyd, S. (2014). Proximal algorithms. *Found. Trends Optim.*, 1(3):127–239.

[137] Pérez, P., Hue, C., Vermaak, J., and Gangnet, M. (2002). *Computer Vision — ECCV 2002: 7th European Conference on Computer Vision Copenhagen, Denmark, May 28–31, 2002 Proceedings, Part I*, chapter Color-Based Probabilistic Tracking, pages 661–675. Springer Berlin Heidelberg, Berlin, Heidelberg.

[138] Poling, B., Lerman, G., and Szlam, A. (2014). Better feature tracking through subspace constraints. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 3454–3461.

[139] Ponomarev, G. V., Arlazarov, V. L., Gelfand, M. S., and Kazanov, M. D. (2014). Ana hep-2 cells image classification using number, size, shape and localization of targeted cell regions. *Pattern Recognition*, 47(7):2360–2366.

[140] Rahimpour, A., Taalimi, A., Luo, J., and Qi, H. (2016). Distributed object recognition in smart camera networks. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 669–673.

[141] Rahimpour, A., Taalimi, A., and Qi, H. (2017). Feature encoding in band-limited distributed surveillance systems. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1752–1756.

[142] Rahmani, M. and Atia, G. (2015a). Innovation pursuit: A new approach to subspace clustering. *arXiv preprint arXiv:1512.00907*.

[143] Rahmani, M. and Atia, G. (2015b). Randomized robust subspace recovery for high dimensional data matrices. *arXiv preprint arXiv:1505.05901*.

[144] Rahmani, M. and Atia, G. (2016). A subspace learning approach for high dimensional matrix decomposition with efficient column/row sampling. In *Proceedings of The 33rd International Conference on Machine Learning*, pages 1206–1214.

[145] Rahmani, M. and Atia, G. (2017). High dimensional low rank plus sparse matrix decomposition. *IEEE Transactions on Signal Processing, 2017*.

[146] Rakotomamonjy, A., Bach, F. R., Canu, S., and Grandvalet, Y. (2008). Simplemkl. *Journal of Machine Learning Research*, 9(Nov):2491–2521.

[147] Ramirez, I., Sprechmann, P., and Sapiro, G. (2010). Classification and clustering via dictionary learning with structured incoherence and shared features. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 3501–3508.

[148] Rehn, M. and Sommer, F. T. (2007). A network that uses few active neurones to code visual input predicts the diverse shapes of cortical receptive fields. *Journal of computational neuroscience*, 22(2):135–146.

[149] Rosasco, L., Verri, A., Santoro, M., Mosci, S., and Villa, S. (2009). Iterative projection methods for structured sparsity regularization.

[150] Ross, A. and Jain, A. (2003). Information fusion in biometrics. *Pattern Recognition Letters*, 24(13):2115 – 2125. Audio- and Video-based Biometric Person Authentication (AVBPA 2001).

[151] Ross, A. A. and Govindarajan, R. (2005). Feature level fusion of hand and face biometrics. volume 5779, pages 196–204.

[152] Ross, D. A., Lim, J., Lin, R.-S., and Yang, M.-H. (2008). Incremental learning for robust visual tracking. *International Journal of Computer Vision*, 77(1-3):125–141.

[153] Ruta, D. and Gabrys, B. (2000). An overview of classifier fusion methods. *Computing and Information systems*, 7(1):1–10.

[154] SchÃlkopf, B., Platt, J., and Hofmann, T. (2007). *Efficient sparse coding algorithms*, pages 801–808. MIT Press.

[155] Shawe-Taylor, J. and Cristianini, N. (2004). *Kernel Methods for Pattern Analysis*. Cambridge University Press, New York, NY, USA.

[156] Shekhar, S., Patel, V., Nasrabadi, N., and Chellappa, R. (2014). Joint sparse representation for robust multimodal biometrics recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 36(1):113–126.

[157] Shen, L., Lin, J., Wu, S., and Yu, S. (2014). Hep-2 image classification using intensity order pooling based features and bag of words. *Pattern Recognition*, 47(7):2419–2427.

[158] Shen, L., Wang, S., Sun, G., Jiang, S., and Huang, Q. (2013). Multi-level discriminative dictionary learning towards hierarchical visual categorization. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 383–390.

[159] Shu, K. and Donghui, W. (2012). A Brief Summary of Dictionary Learning Based Approach for Classification. *ArXiv e-prints*.

[160] Srivastava, N. and Salakhutdinov, R. R. (2012). Multimodal learning with deep boltzmann machines. In *Advances in neural information processing systems*, pages 2222–2230.

[161] Stoklasa, R., Majtner, T., and Svoboda, D. (2013). Efficient knn based hep-2 cells classifier. *Pattern Recognition*.

[162] Storch, W. B. (2000). *Immunofluorescence in clinical immunology: a primer and atlas*. Springer.

[163] Suk, H.-I., Lee, S.-W., Shen, D., Initiative, A. D. N., et al. (2014). Hierarchical feature representation and multimodal fusion with deep learning for ad/mci diagnosis. *NeuroImage*, 101:569–582.

[164] Taalimi, A., Bayati, H., and Fatemizadeh, E. (2009). Clustering method for fmri activation detection using optimal number of clusters. In *Neural Engineering, 2009. NER '09. 4th International IEEE/EMBS Conference on*, pages 171–174.

[165] Taalimi, A., Ensafi, S., Qi, H., Lu, S., Kassim, A. A., and Tan, C. L. (2015a). *Multimodal Dictionary Learning and Joint Sparse Representation for HEp-2 Cell Classification*, pages 308–315. Springer International Publishing, Cham.

[166] Taalimi, A. and Fatemizadeh, E. (2009). fmri activation detection by obtaining bold response of extracted balloon parameters with particle swarm optimization. In *EUROCON 2009, EUROCON '09. IEEE*, pages 1437–1442.

[167] TAALIMI, A. and FATEMIZADEH, E. (2010). Activation detection in fmri images using nonlinear models: Wiener-hammerstein and narma. pages 231–248. IRANIAN JOURNAL OF BIOMEDICAL ENGINEERING.

[168] Taalimi, A. and Fatemizadeh, E. (2010). A new mathematical approach for detection of active area in human brain fmri using nonlinear model. volume 22, pages 409–418.

[169] Taalimi, A., Liu, L., and Qi, H. (2017a). Addressing ambiguity in multi-target tracking by hierarchical strategy. In *2017 IEEE International Conference on Image Processing (ICIP)*.

[170] Taalimi, A. and Qi, H. (2015). Robust multi-object tracking using confident detections and safe tracklets. In *2015 IEEE International Conference on Image Processing (ICIP)*, pages 1638–1642.

[171] Taalimi, A., Qi, H., and Khorsandi, R. (2015b). Online multi-modal task-driven dictionary learning and robust joint sparse representation for visual tracking. In *Advanced Video and Signal Based Surveillance (AVSS), 2015 12th IEEE International Conference on*, pages 1–6.

[172] Taalimi, A., Rahimpour, A., Capdevila, C., Zhang, Z., and Qi, H. (2016a). Robust coupling in space of sparse codes for multi-view recognition. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 3897–3901.

[173] Taalimi, A., Rahimpour, A., Liu, L., and Qi, H. (2017b). Multi-view task-driven recognition in visual sensor networks. In *2017 IEEE International Conference on Image Processing (ICIP)*.

[174] Taalimi, A., Shams, H., Rahimpour, A., Khorsandi, R., Wang, W., Guo, R., and Qi, H. (2016b). Multimodal weighted dictionary learning. In *2016 13th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 173–179.

[175] Theodorakopoulos, I., Kastaniotis, D., Economou, G., and Fotopoulos, S. (2014). Hep-2 cells classification via sparse representation of textural features fused into dissimilarity space. *Pattern Recognition*, 47(7):2367 – 2378.

[176] Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288.

[177] Tran, D. and Sorokin, A. (2008). Human activity recognition with metric learning. In *Computer Vision–ECCV 2008*, pages 548–561. Springer.

[178] Tropp, J. A. (2004). Greed is good: algorithmic results for sparse approximation. *IEEE Transactions on Information Theory*, 50(10):2231–2242.

[179] Varshney, P. K. (1997). Multisensor data fusion. *Electronics Communication Engineering Journal*, 9(6):245–253.

[180] Wang, H., Kläser, A., Schmid, C., and Liu, C.-L. (2013). Dense trajectories and motion boundary descriptors for action recognition. *International Journal of Computer Vision*, 103(1):60–79.

[181] Wang, Q., Chen, F., Xu, W., and Yang, M.-H. (2012). Online discriminative object tracking with local sparse representation. In *Applications of Computer Vision (WACV), 2012 IEEE Workshop on*, pages 425–432. IEEE.

[182] Wang, W., Taalimi, A., Duan, K., Guo, R., and Qi, H. (2016). Learning patch-dependent kernel forest for person re-identification. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–9.

[183] Weinland, D., Boyer, E., and Ronfard, R. (2007). Action recognition from arbitrary views using 3d exemplars. In *2007 IEEE 11th International Conference on Computer Vision*, pages 1–7.

[184] Wright, J., Ma, Y., Mairal, J., Sapiro, G., Huang, T. S., and Yan, S. (2010). Sparse representation for computer vision and pattern recognition. *Proceedings of the IEEE*, 98(6):1031–1044.

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

[186] Wu, B., Ai, H., and Huang, C. (2004). Facial image retrieval based on demographic classification. In *Proceedings of the 17th International Conference on Pattern Recognition, ICPR*, volume 3.

[187] Wu, Y., Blasch, E., Chen, G., Bai, L., and Ling, H. (2011). Multiple source data fusion via sparse representation for robust visual tracking. In *Information Fusion (FUSION), 2011 Proceedings of the 14th International Conference on*, pages 1–8.

[188] Wu, Y., Lim, J., and Yang, M.-H. (2013). Online object tracking: A benchmark. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 2411–2418.

[189] Wu, Y., Lim, J., and Yang, M. H. (2015). Object tracking benchmark. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(9):1834–1848.

[190] Yang, A., Iyengar, S., Sastry, S., Bajcsy, R., Kuryloski, P., and Jafari, R. (2008). Distributed segmentation and classification of human actions using a wearable motion sensor network. In *Computer Vision and Pattern Recognition Workshops, 2008. CVPRW '08. IEEE Computer Society Conference on*, pages 1–8.

[191] Yang, F., Lu, H., and Yang, M.-H. (2014a). Robust visual tracking via multiple kernel boosting with affinity constraints. *IEEE Transactions on Circuits and Systems for Video Technology*, 24(2):242–254.

[192] Yang, J., Wang, Z., Lin, Z., Shu, X., and Huang, T. (2012). Bilevel sparse coding for coupled feature spaces. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2360–2367.

[193] Yang, J., Yu, K., Gong, Y., and Huang, T. (2009). Linear spatial pyramid matching using sparse coding for image classification. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1794–1801.

[194] Yang, J., Yu, K., and Huang, T. (2010a). Supervised translation-invariant sparse coding. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 3517–3524.

[195] Yang, M., Dai, D., Shen, L., and Van Gool, L. (2014b). Latent dictionary learning for sparse representation based classification. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 4138–4145.

[196] Yang, M., Zhang, D., Feng, X., and Zhang, D. (2011). Fisher discrimination dictionary learning for sparse representation. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 543–550.

[197] Yang, M., Zhang, L., Feng, X., and Zhang, D. (2014c). Sparse representation based fisher discrimination dictionary learning for image classification. *International Journal of Computer Vision*, 109(3):209–232.

[198] Yang, M., Zhang, L., Yang, J., and Zhang, D. (2010b). Metaface learning for sparse representation based face recognition. In *17th IEEE International Conference on Image Processing (ICIP)*, pages 1601–1604.

[199] Yang, Y., Wiliem, A., Alavi, A., Lovell, B. C., and Hobson, P. (2014d). Visual learning and classification of human epithelial type 2 cell images through spontaneous activity patterns. *Pattern Recognition*, 47(7):2325 – 2337.

[200] Yuan, M. and Lin, Y. (2006). Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67.

[201] Yuan, X. T. and Yan, S. (2010). Visual classification with multi-task joint sparse representation. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 3493–3500.

[202] Zhang, H., Nasrabadi, N., Zhang, Y., and Huang, T. (2011). Multi-observation visual recognition via joint dynamic sparse representation. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 595–602.

[203] Zhang, H., Zhang, Y., Nasrabadi, N. M., and Huang, T. S. (2012a). Joint-structured-sparsity-based classification for multiple-measurement transient acoustic signals. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 42(6):1586–1598.

[204] Zhang, J., Ma, S., and Sclaroff, S. (2014). Meem: Robust tracking via multiple experts using entropy minimization. In Fleet, D., Pajdla, T., Schiele, B., and Tuytelaars, T., editors, *Computer Vision âĂŞ ECCV 2014*, volume 8694 of *Lecture Notes in Computer Science*, pages 188–203. Springer International Publishing.

[205] Zhang, K., Zhang, L., and Yang, M.-H. (2012b). Real-time compressive tracking. In Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., and Schmid, C., editors, *Computer Vision âĂŞ ECCV 2012*, volume 7574 of *Lecture Notes in Computer Science*, pages 864–877. Springer Berlin Heidelberg.

[206] Zhang, Q. and Li, B. (2010). Discriminative k-svd for dictionary learning in face recognition. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2691–2698.

[207] Zhang, T., Ghanem, B., Liu, S., and Ahuja, N. (2012c). *Computer Vision – ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part VI*, chapter Low-Rank Sparse Learning for Robust Visual Tracking, pages 470–484. Springer Berlin Heidelberg, Berlin, Heidelberg.

[208] Zhang, T., Ghanem, B., Liu, S., and Ahuja, N. (2012d). Robust visual tracking via multi-task sparse learning. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2042–2049.

[209] Zhang, T., Ghanem, B., Liu, S., and Ahuja, N. (2012e). Robust visual tracking via structured multi-task sparse learning. *International Journal of Computer Vision*, 101(2):367–383.

[210] Zhang, T., Liu, S., Xu, C., Yan, S., Ghanem, B., Ahuja, N., and Yang, M.-H. (2015). Structural sparse tracking. In *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, pages 150–158.

[211] Zheng, J. and Jiang, Z. (2013). Learning view-invariant sparse representations for cross-view action recognition. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 3176–3183.

[212] Zheng, Y. (2015). Methodologies for cross-domain data fusion: An overview. *IEEE Transactions on Big Data*, 1(1):16–34.

[213] Zhong, W., Lu, H., and Yang, M.-H. (2012). Robust object tracking via sparsity-based collaborative model. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1838–1845.

[214] Zhou, M., Chen, H., Paisley, J., Ren, L., Li, L., Xing, Z., Dunson, D., Sapiro, G., and Carin, L. (2012a). Nonparametric bayesian dictionary learning for analysis of noisy and incomplete images. *IEEE Transactions on Image Processing*, 21(1):130–144.

[215] Zhou, N., Shen, Y., Peng, J., and Fan, J. (2012b). Learning inter-related visual dictionary for object recognition. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3490–3497.

[216] Zhuang, Y., Wang, Y., Wu, F., Zhang, Y., and Lu, W. (2013). Supervised coupled dictionary learning with group structures for multi-modal retrieval. In *Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence*, AAAI'13, pages 1070–1076. AAAI Press.

[217] Zou, H. and Hastie, T. (2005a). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320.

[218] Zou, H. and Hastie, T. (2005b). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society. Series B: Statistical Methodology*, 67(2):301–320.

[219] Zuev, Y. and Ivanov, S. (1999). The voting as a way to increase the decision reliability. *Journal of the Franklin Institute*, 336(2):361 – 378.

Appendices

Publications

H. Davoudi, A. Taalimi, and E. Fatemizadeh. Extracting activated regions of fmri data using unsupervised learning. In *Neural Networks, 2009. IJCNN 2009. International Joint Conference on*, pages 641–645, June 2009.

R. Guo, L. Liu, W. Wang, A. Taalimi, C. Zhang, and H. Qi. Deep tree-structured face: A unified representation for multi-task facial biometrics. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–8, March 2016.

R. Khorsandi, A. Taalimi, and M. Abdel-Mottaleb. Robust biometrics recognition using joint weighted dictionary learning and smoothed l0 norm. In *Biometrics Theory, Applications and Systems (BTAS), 2015 IEEE 7th International Conference on*, pages 1–6, Sept 2015.

R. Khorsandi, A. Taalimi, M. Abdel-Mottaleb, and H. Qi. Joint weighted dictionary learning and classifier training for robust biometric recognition. In *2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pages 1307–1311, Dec 2015.

L. Liu, A. Rahimpour, A. Taalimi, and H. Qi. End-to-end binary representation learning via direct binary embedding. In *2017 IEEE International Conference on Image Processing (ICIP)*, Sept 2017.

A. Rahimpour, A. Taalimi, J. Luo, and H. Qi. Distributed object recognition in smart camera networks. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 669–673, Sept 2016.

A. Rahimpour, A. Taalimi, and H. Qi. Feature encoding in band-limited distributed surveillance systems. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1752–1756, March 2017.

M. E. Raoufat, A. Taalimi, K. Tomsovic, and R. Hay. Event analysis of pulse-reclosers in distribution systems through sparse representation. In *19th International Conference on Intelligent System Application to Power Systems (ISAP)*, 2017.

A. Taalimi, H. Bayati, and E. Fatemizadeh. Clustering method for fmri activation detection using optimal number of clusters. In *Neural Engineering, 2009. NER '09. 4th International IEEE/EMBS Conference on*, pages 171–174, April 2009.

A TAALIMI and E FATEMIZADEH. Activation detection in fmri images using nonlinear models: Wiener-hammerstein and narma. pages 231–248. IRANIAN JOURNAL OF BIOMEDICAL ENGINEERING, 2010.

A. Taalimi and H. Qi. Robust multi-object tracking using confident detections and safe tracklets. In *2015 IEEE International Conference on Image Processing (ICIP)*, pages 1638–1642, Sept 2015.

A. Taalimi, H. Qi, and R. Khorsandi. Online multi-modal task-driven dictionary learning and robust joint sparse representation for visual tracking. In *Advanced Video and Signal Based Surveillance (AVSS), 2015 12th IEEE International Conference on*, pages 1–6, Aug 2015.

A. Taalimi, A. Rahimpour, C. Capdevila, Z. Zhang, and H. Qi. Robust coupling in space of sparse codes for multi-view recognition. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 3897–3901, Sept 2016.

A. Taalimi, A. Rahimpour, L. Liu, and H. Qi. Multi-view task-driven recognition in visual sensor networks. In *2017 IEEE International Conference on Image Processing (ICIP)*, Sept 2017.

A. Taalimi, H. Shams, A. Rahimpour, R. Khorsandi, Wei Wang, Rui Guo, and H. Qi. Multimodal weighted dictionary learning. In *2016 13th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 173–179, Aug 2016.

Ali Taalimi, Shahab Ensafi, Hairong Qi, Shijian Lu, Ashraf A. Kassim, and Chew Lim Tan. *Multimodal Dictionary Learning and Joint Sparse Representation for HEp-2 Cell Classification*, pages 308–315. Springer International Publishing, 2015.

Ali Taalimi and Emad Fatemizadeh. fmri activation detection by obtaining bold response of extracted balloon parameters with particle swarm optimization. In *EUROCON 2009, EUROCON '09. IEEE*, pages 1437–1442, May 2009.

Ali Taalimi and Emad Fatemizadeh. A new mathematical approach for detection of active area in human brain fmri using nonlinear model. volume 22, pages 409–418, 2010.

Ali Taalimi, Liu Liu, and Hairong Qi. Addressing ambiguity in multi-target tracking by hierarchical strategy. In *2017 IEEE International Conference on Image Processing (ICIP)*, Sept 2017.

W. Wang, A. Taalimi, K. Duan, R. Guo, and H. Qi. Learning patch-dependent kernel forest for person re-identification. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–9, March 2016.

Vita

Ali Taalimi received the M.Sc. degree in Biomedical Engineering from the Sharif University of Technology in 2009. He then received second M.Sc. degree in Electrical and Computer Engineering from The University of Tennessee at Knoxville. He enrolled in the doctoral program at the University of Tennessee at Knoxville under the supervision of Professor Hairong Qi in the Department of Electrical Engineering and Computer Science. His research interests include machine learning, computer vision, signal and image processing.