Subspace Representations and Learning for Visual Recognition

Farzad Siyahjani

Follow this and additional works at: https://researchrepository.wvu.edu/etd

Recommended Citation
https://researchrepository.wvu.edu/etd/6652

This Dissertation is protected by copyright and/or related rights. It has been brought to you by the The Research Repository @ WVU with permission from the rights-holder(s). You are free to use this Dissertation in any way that is permitted by the copyright and related rights legislation that applies to your use. For other uses you must obtain permission from the rights-holder(s) directly, unless additional rights are indicated by a Creative Commons license in the record and/or on the work itself. This Dissertation has been accepted for inclusion in WVU Graduate Theses, Dissertations, and Problem Reports collection by an authorized administrator of The Research Repository @ WVU. For more information, please contact researchrepository@mail.wvu.edu.
Subspace Representations and Learning for Visual Recognition

Farzad Siyahjani

Dissertation submitted to the college of engineering at West Virginia University in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Electrical Engineering

Gianfranco Doretto, PhD., Chair
Bojan Cukic, PhD.
Mark Culp, PhD.
Arun Ross, PhD.
Natalia Schmid, PhD.
Department of Computer Science and Electrical Engineering

Morgantown, West Virginia
2017

Keywords: [subspace learning, visual recognition, low rank decomposition, joint learning, face recognition, interaction recognition, object recognition.]

Copyright © 2017 Farzad Siyahjani
Abstract

Subspace Representations and Learning for Visual Recognition

Farzad Siyahjani

Pervasive and affordable sensor and storage technology enables the acquisition of an ever-rising amount of visual data. The ability to extract semantic information by interpreting, indexing and searching visual data is impacting domains such as surveillance, robotics, intelligence, human-computer interaction, navigation, healthcare, and several others. This further stimulates the investigation of automated extraction techniques that are more efficient, and robust against the many sources of noise affecting the already complex visual data, which is carrying the semantic information of interest. We address the problem by designing novel visual data representations, based on learning data subspace decompositions that are invariant against noise, while being informative for the task at hand. We use this guiding principle to tackle several visual recognition problems, including detection and recognition of human interactions from surveillance video, face recognition in unconstrained environments, and domain generalization for object recognition.

By interpreting visual data with a simple additive noise model, we consider the subspaces spanned by the model portion (model subspace) and the noise portion (variation subspace). We observe that decomposing the variation subspace against the model subspace gives rise to the so-called parity subspace. Decomposing the model subspace against the variation subspace instead gives rise to what we name invariant subspace.

We extend the use of kernel techniques for the parity subspace. This enables modeling the highly non-linear temporal trajectories describing human behavior, and performing detection and recognition of human interactions. In addition, we introduce supervised low-rank matrix decomposition techniques for learning the invariant subspace for two other tasks. We learn invariant representations for face recognition from grossly corrupted images, and we learn object recognition classifiers that are invariant to the so-called domain bias.

Extensive experiments using the benchmark datasets publicly available for each of the three tasks, show that learning representations based on subspace decompositions invariant to the sources of noise lead to results comparable or better than the state-of-the-art.
Contents

1 Introduction 1
  1.1 Visual Recognition and Applications 2
  1.2 Contributions and Applications 5
  1.3 Background 13
    1.3.1 Gaussian Noise Model: 14
    1.3.2 Low Rank Decomposition 15
    1.3.3 Sparse Representation 17

2 Parity Subspace for Human Interaction Recognition 21
  2.1 Introduction 21
  2.2 Literature Review 23
  2.3 Kernel Regression Models 24
  2.4 Kernel State Space Models 26
  2.5 Online Temporal Segmentation 28
    2.5.1 Human Interaction Segmentation 29
  2.6 Recognition 30
  2.7 Experiments 31

3 Uniform Invariant Subspace 33
  3.1 Introduction 33
  3.2 Related Work 35
  3.3 Low Rank Decomposition of Nuisance Factors 37
  3.4 Recognition via the Invariant Subspace 38
  3.5 Invariant Subspace Learning 40
    3.5.1 Low Rank Matrix Recovery 41
    3.5.2 Supervised Learning 42
    3.5.3 Optimization 44
3.6 Classification ................................................. 47
  3.6.1 Local Metric Learning .................................. 47
  3.6.2 Class Separation ........................................ 48
3.7 Experiments .................................................. 49
  3.7.1 Synthetic Data ........................................... 49
  3.7.2 AR Dataset ................................................ 49
  3.7.3 AT&T Dataset ............................................. 52
  3.7.4 Extended-Yale Dataset .................................. 53
  3.7.5 I-LIDS Dataset ........................................... 53
  3.7.6 CAVIAR4REID ............................................. 56
  3.7.7 Labeled Face in Wild .................................... 58
3.8 Model Variations: ........................................... 58
  3.8.1 Uncorrupted Data, $\alpha = \infty$ ....................... 59
  3.8.2 Decomposition without Geometric Constraint, $\gamma = 0$ 59
  3.8.3 Decomposition with Exact Orthogonality Constraint, $\gamma = \infty$ 61
  3.8.4 Unsupervised Decomposition, $\beta = 0$ .................. 64
3.9 Derivative of B ............................................... 71

4 Maximal Margin Invariant Subspace ......................... 73
  4.1 Introduction ................................................ 73
  4.2 Related Works .............................................. 76
  4.3 Unsupervised Low Rank Domain Generalization .......... 78
    4.3.1 Problem Setup .......................................... 78
    4.3.2 Robust SVM ............................................ 79
    4.3.3 Multi-class Learning and Testing ..................... 83
  4.4 Experiments: ................................................ 84
    4.4.1 Office-Caltech10 ....................................... 85
    4.4.2 Action Recognition .................................... 87
    4.4.3 Face Recognition: ..................................... 88

5 Sparse Coding and Dictionary Learning for Recognition ... 89
  5.1 Introduction ................................................ 89
  5.2 Sparse Coding and Dictionary Learning .................. 91
    5.2.1 Adding Discriminative Power .......................... 92
  5.3 Context Model ............................................... 93
    5.3.1 Adding Context Layout information ................... 93
5.3.2 Learning with K-SVD .................................................. 95
5.3.3 Simultaneous Object Localization and Recognition ............... 97
5.3.4 Balancing the Context ............................................... 99
5.4 Experiments ............................................................. 101
   5.4.1 Performance on VOC2007 ....................................... 104
   5.4.2 Performance on SUN09 ........................................... 105

6 Conclusions and Future Work ........................................ 109

Bibliography ................................................................. 113
List of Figures

1.1 Object recognition in the brain, abstraction layers, and their potential links . . . . . . 4
1.2 Additive Linear Model . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6
1.3 Decomposition model: Parity and Invariant subspace . . . . . . . . . . . . . . . . . 7

2.1 Phases of a hand-shaking interaction . . . . . . . . . . . . . . . . . . . . . . . . . 23
2.2 HAUS-PI samples from the human activity . . . . . . . . . . . . . . . . . . . . . . 28
2.3 Temporal segmentation comparison . . . . . . . . . . . . . . . . . . . . . . . . . 29
2.4 F-1 Score on KAUS-PI . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 30
2.5 AMOC curves . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 32

3.1 Additive invariant subspace modeling . . . . . . . . . . . . . . . . . . . . . . . . 39
3.2 Common nuisance (low rank) data across subjects . . . . . . . . . . . . . . . . . . 42
3.3 Identity subspace learning for different subjects . . . . . . . . . . . . . . . . . . . 44
3.4 Decomposition results on synthetic data . . . . . . . . . . . . . . . . . . . . . . . 46
3.5 AR face dataset sample images . . . . . . . . . . . . . . . . . . . . . . . . . . . 49
3.6 Decomposition on AR face attributes . . . . . . . . . . . . . . . . . . . . . . . . 51
3.7 Recognition rates on AR dataset . . . . . . . . . . . . . . . . . . . . . . . . . . . 52
3.8 Extended Yale samples . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 53
3.9 Decomposition on Extended-Yale Face dataset . . . . . . . . . . . . . . . . . . . . 54
3.10 Extended Yale B dataset recognition scores . . . . . . . . . . . . . . . . . . . . . 55
3.11 i-LIDS dataset samples . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 55
3.12 Decomposition on i-LIDS person re-identification . . . . . . . . . . . . . . . . . . 56
3.13 i-LIDS MCTS dataset CMC curves . . . . . . . . . . . . . . . . . . . . . . . . . 56
3.14 CMC Curves comparison on CAVAIR person re-identification . . . . . . . . . . . 57
3.15 LFW dataset original samples . . . . . . . . . . . . . . . . . . . . . . . . . . . . 58
3.16 LFW dataset cropped and aligned samples . . . . . . . . . . . . . . . . . . . . . . 59
3.17 Decomposition on Extended-Yale Face dataset . . . . . . . . . . . . . . . . . . . . 60
3.18 Uncorrupted data decomposition on Extended-Yale face dataset . . . . . . . . . . . 61
3.19 Decomposition without orthogonality constraint on Extended-Yale face dataset . . 62
3.20 Decomposition with exact orthogonality constraint on Extended-Yale face dataset . 63
3.21 Unsupervised decomposition on Extended-Yale face dataset $\beta = 0$ . . . . . . . 65
3.22 Visual results of parameter selection, $\alpha = .5, \gamma = 0, rank = 7$ . . . . . . . 66
3.23 Visual results of parameter selection, $\alpha = .8, \gamma = 0, rank = 26$ . . . . . . 67
3.24 Visual results of parameter selection, $\alpha = .5, \gamma = 0, rank = 2$ . . . . . . . 67
3.25 Visual results of parameter selection, $\alpha = .5, \gamma = .01, rank = 7$ . . . . . . 68
3.26 Visual results of parameter selection, $\alpha = .8, \gamma = .01, rank = 27$ . . . . . 68
3.27 Visual results of parameter selection, $\alpha = .8, \gamma = .01, rank = 2$ . . . . . . 69

4.1 Regular SVM vs. proposed method . . . . . . . . . . . . . . . . . . . . . . . . . . 78
4.2 Office-Caltech10 dataset samples . . . . . . . . . . . . . . . . . . . . . . . . . . . . 84
4.3 Recognition accuracy on AR dataset . . . . . . . . . . . . . . . . . . . . . . . . . . 88

5.1 Context aware sparse code . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 90
5.2 Layout distributions . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 96
5.3 Spatial context subdivision . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 97
5.4 Co-occurrence vs. Map layout . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 102
5.5 Precision-Recall curves . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 104
5.6 Mean AP across all classes vs. Maximum number of initial detections . . . . . . 106
5.7 Detection rescoring instances . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 107
5.8 Object localization and recognition . . . . . . . . . . . . . . . . . . . . . . . . . . 108
List of Tables

2.1 Rand index ................................................. 31

3.1 Recognition accuracy comparison on AR face dataset ...................... 52
3.2 Recognition Rate on AT&T face dataset ................................ 53

4.1 Domain adaptation on office-caltech using DeCAF features experiment 1 ... 84
4.2 Domain adaptation on office-caltech using DeCAF features experiment 2 ... 85
4.3 Domain adaptation on office-caltech using landmark features ............. 86
4.4 Recognition accuracy for domain adaptation on IXMAS ................... 88

5.1 Context vs. non-context per-class AP scores on the PASCAL VOC 2007 dataset ... 103
5.2 Per-class AP scores on the PASCAL VOC 2007 ............................ 105
5.3 Mean AP (Averaged across all object categories) on the SUN09 ............ 107
Chapter 1

Introduction

During last two decades, emerging technology in the digital world has led to many improvements in modeling of data in image processing and computer vision. Pervasive digital camera technology has created a great opportunity for vision related programming and development. Affordable and ubiquitous high-resolution camera technology from one side, portable and capable processors from the other hand have sparked a momentum for development of smart algorithms for automatic interpretation of the surrounding world in sight of the camera, acquired in the form of images or frames. On the other hand, this prevailing technology has created a massive pool of data that needs to be processed automatically. Otherwise, it would be tedious and somehow impossible for human operators to process all these data manually.

Computer vision is the field of acquiring, processing, analyzing and in general understanding of the images or sequential frames. In general computer vision techniques are utilized to produce numerical or symbolic information such as decision. Visual recognition, either for the purpose of surveillance applications or detection and recognition of a particular class/identity is one of the most important and exciting applications engaged with the vast pool of visual data produced on a daily basis. Detection, recognition and identification of different modalities are some of the most challenging topics in computer vision, and despite the enormous amount of research devoted to filling the gap between digital values acquired by cameras and semantic modalities in the real world, still there is a wide gap open to fill for new researchers. Many of this semantics such as class/identity appear in a collection of high-dimensional discrete data. However, these high dimensional data lie close to low dimensional structures corresponding to few classes in the dataset, therefore finding generalizable techniques to model and compare these data is of crucial importance in the field of computer vision.

The process of extracting meaningful information from visual data is called visual recognition. In general visual recognition consist of recognition tasks on static images and videos, such
as object and identity recognition, scene understanding, activity recognition and object tracking. In most of these tasks, the primary goal is to extract low-dimensional representations based on the high dimensional features of appearance, geometry or dynamics of a scene. In this work, we will introduce new techniques for subspace modeling and learning methods along with their applications in the field of computer vision such as face and object recognition, the activity detection and domain adaptation.

1.1 Visual Recognition and Applications

Given huge amount of visual data acquired on a daily basis, visual recognition has become an in-evitable part of many devices surrounding us. Without automated visual recognition, this huge amount of data would be useless if we want to spend human force to manually go through them. To mention couple of applications of visual detection and recognition in computerized systems, we can allude to Autonomous Vehicles, Face Recognition, Content Based Image Retrieval, Optical Character Recognition, Remote Sensing, Robotics, Video Surveillance, Security Systems, Object Modeling, Entertainment and etc... As one can expect, there is an increasing and broad spectrum of applications for the recognition algorithms, therefore increasing the recognition accuracy and speed in current algorithms is of extreme importance.

Visual recognition extensively has been the topic of many kinds of research since the advent of camera technology, but due to the complexity of the problem, it hasn’t been addressed thoroughly yet. During recent years improvements in technology such as the advent of ultra-fast processors from one hand and the introduction of robust methods from the other hand, have made an impressive impact in increasing accuracy of the recognition algorithms and modeling of the data. Here in this dissertation, we will use recent advances in linear and non-linear modeling of data and build new approaches and techniques to improve the state of art results for automated detection and recognition in various computer vision applications.

Although the recognition accuracy of visual recognition systems in controlled environments is close to satisfactory, but the performance in real world applications such as surveillance camera is still an open problem and it is partially due to gross noise factors such as misalignment, noise affection, lack of powerful features, dimensional mismatch or domain shift[1]. With increasing utilization of surveillance cameras in various places, there are increasing demands for face recognition, person re-identification and activity recognition of humans present in data gathered from surveillance cameras, ranging from small-scale stand-alone cameras in banks and supermarkets to large-scale multiple networked closed-circuit televisions on public streets. In such cases, subjects are far from cameras, and face and body regions tend to be small, occluded, etc.. In such settings
that Images are taken without subject’s cooperation, images of the face along with the body image may be contaminated severely.

In surveillance systems, the temporal segmentation and recognition of continuous activities from the video is a core problem to address for enabling intelligent systems that can extract and manage content fully automatically. Temporal segmentation can support core tasks such as video understanding and action or activity analysis [2], as well as motion capture data analysis and animation [3]. Despite a significant amount of research focusing on recognizing actions and activities, the problem of their time localization has received considerably less attention [4]. Even more so, if the localization has to be performed online, and enable the processing of video in real-time.

In order to gain some insights to visual recognition requirements of our modern era, maybe it is a good idea to briefly look into the human visual system before starting to explain the contributions and models and methods that we have used in this work. Some lessons can be learned from the human visual system and reviewing current knowledge about the structure of the human brain and the way it handles visual recognition problem from a higher view, will provice a better insight into visual recognition problem and it will help us with coming up solutions for automated visual recognition problem. Although, we do not yet fully know how the brain solves object recognition [5], but there have been some improvements in understanding the mechanism behind the visual recognition of the brain. It is known that primates can perform visual recognition of a given visual scene rapidly (< 200 ms viewing duration) without any object-specific or location-specific pre-cueing [6]. It is noteworthy that this task is performed regardless of the changes in object position, size, viewpoint, and visual context, which all are identity-preserving transformations, therefore there are a vast array of the images from a unique scene that should be labeled as same. The response of a population of neurons to a peculiar view of one object is a response vector in a space in which dimensionality is defined as the number of neurons in the population [5]. Object recognition is described elegantly with a large number of feedforward nonlinear filtering and is expressed as a population rate code (similar to sparse coding) in less than 50ms. Fig. 1.1 provides a brief overview on how human vision handles the object recognition, in four different areas of the brain along with the dimension of data that each section processes.

In Fig. 1.1 it is shown that in early sensory neural population level, population activity patterns corresponding to different objects are tangled together and as the information proceeds in the ventral system and IT, they are represented in which, this representation untangles the data corresponding to different objects. In order to fully understand the remarkable architectural homogeneity of the mammalian brain, high-throughput computer simulations are required to explore the vast space of possible subnetwork algorithms systematically, implementing each possibility as a cascaded, full-scale algorithm, and measuring performance in carefully considered benchmark
object recognition tasks. For further details on the architecture and object recognition in the human brain, please refer to [5]. Despite its intricate and gigantic network of possibilities, recently there have been some improvements in the implementation of such neural networks for limited (but relatively high) number of objects such as Convolutional Neural Networks (CNNs) [7], which yields a noticeable boost to the accuracy of existing visual recognition systems. However, these implementations need millions of training samples to learn from and yet their training will take in some cases weeks to complete and learn the deep structure of neural networks. CNNs exploit spatially-local correlation by enforcing a local connectivity pattern between neurons of adjacent layers. In other words, the inputs of hidden units in layer $m$ are from a subset of units in layer $m - 1$ that have spatially adjacent receptive fields. As we study the evolution of high-dimensional visual inputs throughout these layers, we find out that as we move forward with layers the complexity of these high dimensional data wanes and their rank degenerate, and eventually it diverges to an invariant subspace. Inspired with the knowledge gained from the subspace untanglers and invariant subspace representations in brain, we devote this thesis to investigating unexplored subspaces in the literature of subspace modeling for visual recognition.

Data representation plays a key role in the success of the machine learning algorithms, as it is
believed that the explanatory and representative data are hidden in data. Generic priori along with domain specific knowledge can be used to design representation models, moreover, with the drive for integration of Artificial Intelligence in various applications, the design of such powerful data representation models with such priori has become more important. In this thesis, we will introduce new data representation models based on linear decomposition and its utilization on non-linear feature spaces. This thesis is about representation learning, where these representations make it easier to extract useful information, and as a result, improve the performance of the classifiers and the predictors defined on these representations. In order to exploit the full power of such representations, we combine learning such representations with learning classifiers to boost the power of such preprocessing in final models. We will introduce two new subspaces as *Parity Subspace* and *Invariant Subspace* along with their applications in visual recognition on simple linear additive noise decomposition of input data. In the following, we will show how these subspaces are defined and how effective they are in various visual recognition applications.

### 1.2 Contributions and Applications

Linear subspaces and other manifolds are widely used to represent data. Classification, clustering and dimensionality reduction are often driving motivations. We can categorize subspace learning and representation methods from machine learning perspective into two main categories of *Supervised* and *Unsupervised* subspace learning methods. For example in the case of supervised methods as Principle Component Analysis (PCA) [8], Metric Multidimensional Scaling (MDS) [9], Locally Linear Embedding (LLE) [10], Locality Preserving Projections (LPPs) [11], Sparse Representation Classification (SRC) [12] or many of other methods such as AE, RP, DCT, Wavelet..., while most of these methods are designed for data representation and preserving the main components of the data for some specific purposes such as compression or reconstruction, the supervised methods are purely focusing on data representation methods, where the main goal is classification or data clustering, given the labels for training samples. Some of the main contributions in this category are: Linear/Fisher Discriminant Analysis (LDA/FDA) [13], Canonical Correspondence Analysis(CCA) [14], Independent Component Analysis (ICA) [15], SDA, Sparse Discriminant Analysis [16], PLS [17] etc...

Also from computer vision aspect, we can divide subspace modeling methods into three main categories of subspace modeling methods using *appearance, shape* and *motion*. In order to mention some of the works that have used appearance in subspace modeling we can mention Tensor Subspace Analysis (TSA) [18, 19, 20], Vector-based Linear Subspace Learning for Visual Tracking (VLSL) [21, 22, 23], Tensor-based Linear Subspace Learning RSSL [24], Robust Struc-
tured Subspace Learning [25], Active Appearance Models (AAM) [26] and EigenFaces, FisherFaces [27], LaplacianFaces [28], Shape subspace modeling such as Dynamic Shape Manifolds (DSM) [29], Robust Subspace Learning (RSL) [30] and Motion subspace models such as Low Rank Tracker(LR) [31], Spatiotemporal Features With 3D Convolutional Networks [32], Vector-based Linear Subspace Learning for Visual Tracking (VLSL) [21].

Almost all of subspace modeling methods introduced for visual recognition application are investing on learning the model subspace or variation subspace, while there are other subspaces in the simple additive noise model that is kept untouched. As it can be seen in Fig. 1.2, an observation data point $X$, can be modeled with simple additive decomposition model $X = S + V$, where $X$ is the input data point and it can be refactored into two components. $S$ is the portion of the data that pertain to model subspace and $V$ is the variation or noise component. We will introduce two new subspaces as parity subspace and invariant subspace, where parity subspace is the orthogonal complement of the model subspace and invariant subspace is the orthogonal complement of the variation subspace. We define the parity subspace as a space that can be used for measuring the integrity of the observations with respect to model subspace. Therefore the projection of the data point on this subspace (given that such a projection exists) will be used to measure the conformity of the data point with respect to the model that we are interested in for recognition application. On the other hand, the invariant subspace is the subspace that spans the sufficient component of the model where no variation element exists. Fig. 1.3 illustrates respective geometry of the introduced subspaces. In this thesis mainly we will focus on how these subspaces can be learned and how they are useful for the visual recognition applications.

We will introduce new learning problems to model input data on parity and invariant subspaces to infer different semantics in the applications as diverse as human activity recognition, identifica-
tion, object recognition and domain adaptation (please see chapter 4 for the definition of domain adaptation). To formulate the idea of exploiting subspace evolution to identify an action, we will introduce a theoretically grounded approach that combines the geometry of Reproducible Kernel Hilbert Space (RKHS) with linear models for the purpose of performing temporal segmentation and recognition of human interactions. We will propose an online approach to coping with the high dimensions of the data, as well as the complexity of their variability by combining notions from two well-understood theories and formalisms. The first one is the theory on reproducing kernel Hilbert spaces [33], and the second is the theory of state space models [34]. Exploiting the power of kernels allows a flexible and efficient blending of heterogeneous high-dimensional features which can be mapped into a suitable Hilbert space where they can easily be modeled, even with linear models. Exploiting the theory on state space models allows borrowing some well-understood results about their estimation, and their power for doing analysis, recognition, and detection based on multidimensional temporal sequences. So in pursuit of our subspace modeling methods, we will specifically use the concept of parity subspace for the human interaction segmentation and recognition. For this purpose, we will use a Linear Dynamic System to model the dynamics of the human activity, and we will introduce the idea of subspace evolution during an event in video sequences in conjunction with a projection on parity subspace to localize the activity.

In order to capture the variation space \( (V) \) and its orthogonal complement the invariant subspace \( (V_\perp) \), we will use low-rank modeling framework, where we use low-rank constraint to model nuisance factors. We exploit the concept of invariant subspace in two different applications. First
we show how this subspace can be learned and used in applications like face recognition and more general framework, like person re-identification, and second, we show how invariant subspace can be used for domain generalization. In both of these approaches we use a low-rank minimization problem to capture the variation subspace; however, for the invariant constraint, we use different constraints which will make each one of these approaches suitable for various applications. We introduce these two invariant subspace modeling approaches in chapters (3) and (4) respectively.

Last but not the least, we will study a method to add contextual information in subspace learning to increase the recognition accuracy, we extend our work to model contextual information of objects in a scene and embed this contextual information in high dimensional data before attempting to untangle them. We introduce a novel method to encode contextual information inside high-dimensional appearance features, then use K-SVD method to learn a dictionary that is efficient in a sparse representation framework. In other words, we are jointly learning a dictionary by finding an optimum subspace that best models visual discriminative features and contextual information jointly in a sparse reconstruction framework. Here, in the following, we will elaborate more on each proposed model with motivations for our approach.

**Online subspace tracking for activity recognition:** handling the complexity of the variability of data that represent activities, which is inherently multidimensional in applications like temporal segmentation and recognition of continuous activities from the video, is a core problem to address for enabling intelligent systems that can extract and manage content fully automatically. In chapter (2), we propose an online approach to cope with the high dimensions of the data, as well as the complexity of their variability by combining notions from two well-understood theories and formalisms. The first one is the theory on reproducing kernel Hilbert spaces [33], and the second is the theory of state space models [34]. Exploiting the power of kernels allows a flexible and efficient blending of heterogeneous high-dimensional features which can be mapped into a suitable Hilbert space where they can easily be modeled, even with linear models. Exploiting the theory on state space models allows borrowing a number of well-understood results about their estimation, and their power for doing analysis, recognition, and detection based on multidimensional temporal sequences. The resulting approach allows to extend the notion of parity space, developed within the context of detection based on linear models [35], for its use together with kernel regression, and kernel state space models, which are the Hilbert space counterparts of the linear versions. Rather than using Euclidean geometry to project data onto the parity space and reveal a detection, we exploit the geometry of linear operators in Hilbert space, and derive closed-form solutions for the computation of normalized test statistics, based solely on kernel evaluations. The framework
Supervised low rank decomposition: we introduce a novel approach for primarily face recognition and person re-identification based on learning invariant subspaces, which along with the experimental results proves to be effective in any classification problem. During recent years low rank methods have proven to be effective in many applications in computer vision. It has proven to produce promising results on denoising [12], transformation learning[38] and 3D reconstruction [39], data alignment [40], subspace clustering [41], dimension reduction [42], face recognition [43], classification [44, 45] tracking [46].

Despite the fact that the concept of matrix rank minimization have been used since long time ago, but introducing the concept of sparsity on one hand and recent improvements in techniques introduced for optimization of nuclear norm and $l_1$-norm as convex approximation of rank and $l_0$-norm, respectively, has evolved the concept of noise modeling. Consequently, this has led to many key developments in the area of matrix subspace analysis. It has been proven that if matrix $X$ is composed of the columns which contain samples of appearance data points from instances of a training set, typically these data points lie in a low dimensional structure which represents certain categories. So far all research work developed around the low-rank methods have the assumption that images from an identity lay in the space spanned by columns of a low-rank matrix plus a gross noise which is modeled with a sparse matrix. However, in chapter (3) we will show that there is a problem with this assumption, where although these high dimensional data lay in a span of a low-rank matrix, but in the presence of multiple identities, subjects with different identities share a noticeable span of subspace among each other, in which we call it Variation subspace (V). Fig. 3.2 illustrate an extreme case where nuisance factors almost dominate in appearance features and carry a great span of common subspace between two different identities, therefore in such cases low-rank matrix will not lay in the span of subspace for a unique identity, though it will lay in dominant subspace which represents nuisance factors. To address this problem in this chapter rather than learning an identity subspace using rank minimization, we propose to learn the common nuisance factor span among identities using low-rank minimization. In this way, gallery and test images could be projected onto the orthogonal complement of the nuisance subspace, in order to be compared without the effect of the corruption. In particular, given the gallery of images, we introduce a model where:

(a) the columns of a low-rank matrix are used to span the space of nuisance factors that are common across the gallery;
(b) the sparse noise models the nuisance factors that are peculiar of a given image in the gallery;
the remaining image information should be representative of a given identity and should lie in a space that is orthogonal to the space spanned by the nuisance factors.

In chapter (3) we will propose a new optimization problem which models all highlighted goals and will provide the solution for the proposed problem. We will investigate the effect of any of proposed elements in the model and analyze the possible modifications and variations of the model, eventually in the experimental section we will test proposed model on three face recognition benchmarks and to show the generalization property of the approach, we will test it on person re-identification benchmarks as well. As it is illustrated in experimental section our approach outperforms existing state of the art identification algorithms using sparse and low-rank matrix decomposition.

**Robust max margin learning:** we introduce a novel domain generalization method without any prior constraint on the training data. We propose a new method for learning unbiased max margin classifiers in which, given samples from multiple source domains, is robust against nuisance factors such as domain shifts. The proposed method is an entirely unsupervised adaptation method, despite most of the works done in past that need the target domain to find the intermediary subspace.

It has been shown that existence of nuisance features in machine learning during training does not even increase complexity and convergence time of the learning algorithm, it also deteriorates the learning accuracy and eventually degenerates the classification performance. Even more with the explosion of the online image and video dataset the importance of robust learning methods to overcome the challenge of nuisance factors in the object manifestation (i.e. overcoming features extracted from background across different categories) in this pool of images and videos is felt more than ever. These common nuisance factors naturally can be divided into low-level factors such as occlusion, clutter, viewpoint changes, image saturation, background nuisance features common across different categories and geometric or photometric variations, on the other hand, high-level factors are such as dataset bias or label bias in scalable applications. Numerous publications discuss intuitively and illustrate empirically that almost all attempts to come up with an unbiased dataset have failed[47, 48], and best approach to handle dataset bias is to model it during learning and equip the learning machine to model the dataset bias and eliminate it from the equation. Naturally, some of these variations are inherently linear and of additive nature. In this work we propose a new method in which we explore and subtract the nuisance subspace or in a more conventional paradigm the existing bias in training data and learn large margin hyperplane on the remanent of the data. Usually, both of high-level and low-level nuisance factors appear in structured factors that can be modeled as linear subspaces.

It is well known that robust PCA can capture dominant global data distribution information and
PCA is used widely for dimension reduction in classification applications, there are numerous applications of the robust principle component analysis in video or image processing that successfully disintegrate the background and foreground or the bias from the real object of interest[12]. In this work, we use matrix factorization to obtain the bias over data and since shared subspace (bias) is predominantly orthogonal to separating hyperplane we add the orthogonality constraint to better distinguish the bias; therefore the learnt hyperplane would be a robust decision boundary that represents the actual classification hyperplane. One might raise the question that nonlinear SVM is introduced to address these variations and using a custom kernel can resolve the problem mentioned above, but still, these methods fall short in situations that there is a structured drift in data as well as there is not a deterministic solution for kernel selection or custom kernel design. Our method is able to automatically analyze the input data, detect the bias and nuisance factors and eliminate them in learning procedure. Therefore the learned pair of bias and decision boundary can be used in testing time to map the data into the expected margin. In this method using an unsupervised max margin learning we are learning a decision boundary, however since this decision boundary is prone to learn the database bias we use an unsupervised robust principle component analysis to subtract the bias factor from efficient data. Eventually each learned support vector will have a corresponding bias factor which will be used in testing time to eliminate the data bias.

Context-aware dictionary learning: we introduce a novel dictionary learning approach for sparse coding where the context of the objects along with their appearance features are learned in dictionary simultaneously. Sparse coding has been used successfully in a variety of image processing, and computer vision applications, like denoising [49], restoration [50], and classification [12, 51]. In sparse representation the main idea is to approximate a signal with an over-complete dictionary. The way in which such dictionary is learned from data greatly affects the task at hand. In object recognition, one of the central computer vision tasks, the dictionary is learned under stringent conditions dictated by the nature of the problem. In particular, the number of training data samples is very large, and high classification performance requires a dictionary with discriminative power. Sparse coding learns a dictionary while maximizing the reconstructive power given the sparsity constraint, with large training datasets, this leads to large dictionaries, which is a problem when the input data is given to the sparse solver in charge of computing the corresponding sparse code. The successful SRC algorithm [12] deals with large datasets by manually selecting the training samples used to construct the dictionary. This is suboptimal in many respects [52], and it does not scale well with the number of object classes to recognize, which requires a proportional increase in the dataset size. To address this problem, small-size dictionary learning has been approached by [52, 53, 54, 55, 56, 57]. These works deploy or extend either the method of optimal directions (MOD) [55] or the K-SVD algorithm [56]. Supporting recognition tasks requires a discriminative
dictionary. On the other hand, the K-SVD algorithm, which has established itself for its simplicity, numerical efficiency, and fast convergence, focuses on learning an over-complete dictionary that maximizes the reconstructive power. Rather than decoupling reconstructive and discriminative properties by first learning a dictionary for representation and then training a classifier, like it is done in [53, 58, 59], recent works [52, 57] have achieved state-of-the-art performance by elegantly extending the K-SVD algorithm to simultaneously learn a dictionary that is compact, reconstructive, and discriminative. On the other hand, many works have proven that contextual information of great importance for holistic understanding and improvement of detection. This intact application intrigued us to extended sparse representation classification to consider the whole image in a holistic manner and exploit contextual information in sparse reconstruction and boost recognition accuracy. Later in the chapter (5) we will elaborate on the formulation and as we will see in experimental section practical results are best approval of using contextual information in sparse representation.

To demonstrate the effectiveness of the proposed method, we design a framework of structured prediction that addresses the problem of correctly detecting and localizing in an image, multiple objects from multiple classes. As input, such framework takes an image with multiple bounding box hypotheses Bounding box hypotheses can be obtained with a sliding window detector and assigns an object class label to all of them simultaneously. Even with a simplified model we show that it is possible to estimate the object layout at a very high speed, and obtain performance in line with the state-of-the-art on the PASCAL VOC 2007 dataset [60].

This thesis is organized into five chapters that each of them will introduce an application of the parity space and invariant subspace for various computer vision applications, such as identification, domain adaptation, activity detection and object recognition, please note that applications are not limited to reported experiments and they can be used in any application that requires mining low dimensional subspaces from high dimensional data. Chapter one provides some introduction to the content of this thesis, and it will elaborate motivations, applications and some prerequisites for the remaining chapters of the thesis. In this chapter we provide some general background on fundamentals of low-rank decomposition and sparse representation.

In chapter 2, by exploiting lessons learned from modeling orthogonal subspaces for learning invariant subspace, we investigate the effectiveness of mapping data to non-linear Reproducible Kernel Hilbert Space (RKHS) and introduce new statistics based on linear methods in Hilbert space that we developed to improve the detection and recognition accuracy on benchmark challenges. We will present a new method for online temporal segmentation and recognition of human interactions in video sequences. We handle the complexity of the high-dimensional data variability over time by combining the representation in kernel Hilbert space with the use of linear models. This allows
the use of the *parity space* in Hilbert space, and deriving closed form statistics based on kernel evaluations for online segmentation.

In chapter 3, we will introduce a model to learn the invariant components, a representation invariant to nuisance factors. This enables a very efficient and robust classification based on a simple nearest neighbor. We will develop a framework based on the geometry that ensures a uniform inter-class separation. Besides, we will show how the approach is equivalent to a local metric learning, where we will introduce a novel matrix decomposition framework, which using enough labelled training data we learn a supervised subspace data using low-rank methods, this work will benefit from multiple instances of a given subject/class to learn a robust and unique identity subspace for each subject/class where can be used in testing phase for distance measure of the test sample from each unique subject subspaces.

In chapter 4, in a closely related architecture to the model presented in second chapter, we will introduce a model again to learn the invariant components, however contrary to the previous approach instead of simple geometry we will learn a max marginal classifier which is robust to a different type of nuisance factors such as domain bias. Chapter 3 and 4 are closely related and they are proposing similar formulation, in both of proposed methods we are learning orthogonal subspaces which are learned to model sufficient and nuisance subspaces.

And eventually, in chapter 5, we will continue to investigate the applications of subspace learning and sparsity in the accuracy of recognition individually. We will study effect of object layout in a given image and subsequently in detected objects, then develop a novel framework to learn a dictionary for sparse coding scheme, where object layout is also involved, in other words, detected objects are represented based on learned context-aware dictionary, and this dictionary is learned using iterative K-SVD which learns a subspace constructed by both appearance features of detected objects and their layout.

### 1.3 Background

In the following, we will provide an introductory knowledge over rank minimization and sparse representation and will try to cover existing preliminary works on it. In order to provide an overview about the reason behind appearance of sparse representation, and how it is used to model gross noises present in many instances of real-world visual data, we start off with the case where white Gaussian noise is added to a given signal and we are interested in recovering original data from its contaminated version by gaussian noise, the assumption of white gaussian noise although beneficial in many applications but unfortunately is not an enough assumption in most of the applications in image processing, where semantic reasoning is based on appearance features of the
2-D signal. The main reason for the shortcoming of gaussian noise assumption is that in vision applications the biggest obstacles in automated recognition are of nonlinear nature and originate from facts like occlusion, pose, illumination change, the intra-class variability of objects, misalignment etc. On the other hand, it is shown that in many vision applications due to plenty of training samples in varying visual conditions most of the data lay in a low-rank data matrix. Therefore in order to model all these variations, an establishment of more expert and advanced modeling tools is inevitable. In the following we will provide a background on solution for gaussian noise modeling and later we will show how this modeling is generalized to gross sparse noise modeling and rest of this section is devoted to sparse representation for sparse noise modeling (i.e. occlusion) and low-rank methods for subspace and component analysis in presence of sparse noise.

### 1.3.1 Gaussian Noise Model:

Given \(N\) noiseless low rank data points in a matrix \(A = [a_1, a_2, ..., a_N]\) corrupted by gaussian noise \(E\), can be written as \(X = A + E\), the problem of finding low-rank matrix of \(A\) can be formulated as

\[
\min_{A} \|X - A\|_F^2 \quad \text{s.t.} \quad \text{rank}(A) \leq r
\]  

(1.1)

The optimal solution to this problem which is in fact Principal Component Analysis (PCA), is given by \(\hat{A} = U\Theta_\epsilon(\Sigma)V^\top\), where \(X = U(\Sigma)V^\top\) is the singular value decomposition of \(X\), \(\Theta(\cdot)\) is the hard thresholding operator

\[
\Theta_\epsilon(x) = \begin{cases} 
  x & |x| \geq \epsilon \\
  0 & \text{else}
\end{cases}
\]  

(1.2)

In (1.1), \(r\) is unknown and varying \(r\) triggers a trade off between rank of \(A\) and noise power \(\|X - A\|_F^2\). In order to make \(r\) fluctuate, the problem (1.1) can be formulated as

\[
\min_{A} \quad \text{rank}(A) + \frac{\alpha}{2} \|X - A\|_F^2,
\]  

(1.3)

despite (1.1) where \(r\) sets the trade off between noise cancelation and rank of \(A\), in (1.3), \(\alpha\) is the parameter that sets the trade off between noise and rank. The optimal solution for a fixed rank \(r\) in (1.3) is \(\hat{A} = U\Theta_{\sigma_{r+1}}(\Sigma)V^\top\), we can reformulate (1.3) and look for optimum \(r\), with a fixed proportion of noise cancelation and rank, therefore (1.3) can be written as:

\[
\min_r \quad r + \frac{\alpha}{2} \sum_{k>r} \sigma_k^2.
\]  

(1.4)
In this problem optimal $r$ is the smallest $r$ such that $\sigma_{r+1} \leq \sqrt{2/\alpha}$, therefore it can be inferred that the optimal solution is $\hat{A} = U\Theta \sqrt{2/\alpha}(\Sigma)V^\top$.

Since rank minimization is in general NP-hard, it is common practice to use nuclear norm $\|A\|_* = \sum_k \sigma_k$ where $\sigma_k$ is the singular value of $A$, as a convex hull for the rank operator. Then replacing rank operator with nuclear norm in (1.3), convex minimization can be written as

$$\min_A \|A\|_* + \alpha/2 \| X - A \|_F^2,$$  \hspace{1cm} (1.5)

where $\alpha > 0$ is a parameter defined by user to set noise level, It is shown [61] that the optimal solution to (1.5) is given by $\hat{A} = U\xi_{\alpha-1}(\Sigma)V^\top$, where $\xi$ is the shrinkage-thresholding operator defined as following:

$$\xi_{\alpha}(x) = \begin{cases} 
  x - \epsilon & |x| > \epsilon \\
  x + \epsilon & |x| < -\epsilon \\
  0 & \text{else}
\end{cases}$$  \hspace{1cm} (1.6)

Notice that the later solution does not coincide with the one given by PCA, which performs hard-thresholding of the singular values of $X$ without shrinking them by $\epsilon$.

1.3.2 Low Rank Decomposition

In the previous section, we saw Euclidean distance of given data points can best model gaussian white noise. However, it fails for the most ubiquitous noise instances in vision applications such as occlusion, pose, illumination change, inter-class variability misalignment etc. It is shown [62] that this kind of noises can be best modeled as sparse nuisance components in a low-rank space, [62] proposed an unconstrained decomposition of matrix $X$ to low rank components $A$ and sparse $E$ nuisance factors.

$$\min_{A,E} \text{rank}(A) + \alpha\|E\|_0 \quad \text{s.t.} \quad X = A + E,$$  \hspace{1cm} (1.7)

where $\alpha$ is a user defined parameter, which sets the trade off between sparsity of $E$ and rank of $A$, $\|E\|_0$ is the number of non zero entries of matrix $E$. Similar to (1.4), nuclear norm is used as convex hull for the rank operator, on the other hand $l_0 - \text{norm}$ is non-convex operator too and solving it is NP hard, therefore in order to get convex hull of the optimization problem in (1.7), the $l_0 - \text{norm}$ is replaced with its convex hull $l_1 - \text{norm}$ [62], therefore 1.7 can be written as

$$\min_{A,E} \|A\|_* + \alpha\|E\|_1 \quad \text{s.t.} \quad X = A + E,$$  \hspace{1cm} (1.8)
Since this problem doesn’t have closed form solution or at least is not known yet, many works has attempt to minimize the objective function (1.8) using convex optimization techniques. A chronological evolution of methods proposed to solve this problem can be found in section (?), and here we will elaborate on Augmented Lagrangian Multiplier (ALM) as the most recent and effective method available. In which augmenting a lagrangian multiplier \( \lambda \) enables us to add constrain \( X = A + E \) as an additional term to the cost function and in an iterative framework, minimize the derived objective function

\[
\min_{A,E} \|A\|_* + \alpha \|E\|_1 + \langle \lambda, X - A - E \rangle + \frac{\mu}{2} \|X - A - E\|_F^2 \quad \text{s.t.} \quad X = A + E, \quad (1.9)
\]

In (1.9) assuming \( \lambda \) is fixed , the optimization problem can be solved using convex numerical methods with respect to \( A \) and \( E \), then Eq. (1.9) can be written as following:

\[
\min_{A,E} \|A\|_* + \alpha \|E\|_1 + \frac{\mu}{2} \|X - A - E\|_F^2 + \mu \|\lambda\|_F^2 - \|\lambda\|_F^2, \quad (1.10)
\]

This equation is similar to the one in (1.5) and given \( \lambda \) and \( E \), its optimal solution is given by \( A = U \xi_{\mu^{-1}}(\Sigma)V^\top \) where \( U \Sigma V^\top \) is the singular value decomposition of \( (X - E + \mu^{-1}\lambda) \), and it is shown that given \( Y \) and \( A \), the optimal \( E \) satisfies the following equation:

\[
-\mu(X - A - E + \mu^{-1}\lambda) + \alpha \text{sign}(E) = 0, \quad (1.11)
\]

It is shown in [62] that this equation can be solved in closed form using shrinkage-thresholding operator as \( E = \xi_{\alpha,\mu^{-1}}(X - A + \mu^{-1}\lambda) \). Therefore all the elements necessary for solving the optimization problem (1.8) are in place and can cast as following

\[
(U, \Sigma, V^\top) = \text{svd}(X - E + \mu^{-1}\lambda) \\
A_{k+1} = U \xi_{(\eta \mu_k)^{-1}}(\Sigma) V^\top \\
E_{k+1} = \xi_{(\alpha \mu_k)^{-1}}[X - A_k + \mu^{-1}\lambda_k] \\
\lambda_{k+1} = \lambda_k + \mu_k[X - A_k - E_{k+1}] \\
\mu_{k+1} = \rho \mu_k \quad (1.12)
\]

this framework is referred as inexact ALM solution for RPCA, the reason to refer to this algorithm as inexact solution is the convergence condition of this algorithm, where if \( \sum_{k=1}^{+\infty} \mu_k^{-1} = \infty \) is not satisfied then there is no guarantee that this algorithm converges to the optimal solution of RPCA (in Eq. (1.12) we might have \( \sum_{k=1}^{+\infty} \mu_k^{-1} < \infty \)), however, there is an exact algorithm that ensures \( \mu_k \) does not grow too fast or in other words \( \sum_{k=1}^{+\infty} \mu_k^{-1} = \infty \), for more details and discussion on exact
and inexact solution reader is referred to [62]. Here in this dissertation, we will build on 1.12 and in chapter 3 will show how this low-rank decomposition can be extended for learning an identity subspace which can be used in recognition of test sample.

### 1.3.3 Sparse Representation

As we saw in previous section low-rank methods and sparse noise decomposition was introduced as one of the tools used for modeling noisy data in vision applications where nuisance factors such as occlusion, illumination change, etc... disguise the data. However It is not the only model used for handling noisy data in vision applications. There is another powerful tool to handle this kind of nuisance data which is sparse representation [12]. Given N data points in a matrix

\[ X = [x_1, x_2, ..., x_N], \]

sparse representation is introduced to reconstruct matrix \( X \) using linear combination of sparse elements from a given dictionary \( D \):

\[ < D, A > = \arg \min_{D,A} \| X - DA \|_2^2 \quad \text{s.t.} \quad \| A \|_0 < T, \tag{1.13} \]

This section briefly reviews concepts on sparse coding and how to construct a dictionary for sparse coding, later in chapter (5) we will explain how to learn dictionaries for this representation and how discriminative power and contextual information can be added to it. In practice, signals tend to be compressible, rather than sparse. Mathematically, a compressible signal has a representation whose entries decay rapidly when sorted in order of decreasing magnitude. Compressible signals are well approximated by sparse signals, so the sparse approximation framework applies to this class. In practice, it is usually more challenging to identify approximate representations of compressible signals than of sparse signals.

The counting function \( \| . \|_0 : \mathbb{R}^N \rightarrow \mathbb{R} \) returns the number of nonzero components in its argument. We say that a vector \( a \) is s-sparse when \( \| a \|_0 \leq s \). When \( x = Da \), we refer to \( a \) as a representation of the signal \( x \) with respect to the dictionary \( D \). Where \( D = [d_1, \cdots, d_N] \in \mathbb{R}^{n \times k} \) with \( k \gg n \) atom elements. The most basic problem we consider is to produce a maximally sparse representation of an observed signal \( x \) as in following:

\[ \min_a \| a \|_0 \quad \text{s.t.} \quad x = Da. \tag{1.14} \]

One natural variation is to relax the equality constraint to allow some error tolerance \( \varepsilon \geq 0 \), in case the observed signal is contaminated with noise

\[ \min_a \| a \|_0 \quad \text{s.t.} \quad \| x - Da \|_2 \leq \varepsilon. \tag{1.15} \]
It is most common to measure the prediction observation discrepancy with the Euclidean norm (see 1.3.1), but other loss functions may also be appropriate. The elements of Equation (1.15) can be combined in several ways to obtain related problems. For example, we can seek the minimal error possible at a given level of sparsity $s \geq 1$

$$\min_a \|x - Da\|_2 \quad \text{s.t.} \quad \|a\|_0 \leq S.$$  

(1.16)

We can also use a parameter $\lambda > 0$ to balance the twin objectives of minimizing both error and sparsity

$$\min_a \frac{1}{2}\|x - Da\|^2_2 + \lambda\|a\|_0.$$  

(1.17)

We can simply generalize the problem in (1.16) from a vector to a full matrix reconstruction form and also consider dictionary learning step in optimization problem then for a Given set $X \in \mathbb{R}^{n \times N}$ of $N$ $n$-dimensional signals, a reconstructive over-complete dictionary $D \in \mathbb{R}^{n \times k}$ with $k \gg n$ atom elements for the sparse representation of $X$ is learned by solving the problem where $A \in \mathbb{R}^{k \times N}$ is the sparse representation of the signals in $X$. The cost function that is being minimized is the reconstruction error, the sparse representation of $X$ is learned by solving the problem

$$\langle D, A \rangle = \arg \min_{D, A} \|X - DA\|^2_2 \quad \text{s.t.} \quad \|A\|_0 < S,$$  

(1.18)

while the sparsity constraint imposes that the $\ell^0$-norm of $A$ should be less than $S$. This means that every sparse code $a_i$, representing $x_i$, should have no more that $S$ elements different than zero. Also in order to refrain entries of $D$ growing unbounded, we confine norm of each column of $D$ to be unit, which we don’t include in the formulation for simplicity. If there are no restrictions on the dictionary and the signal $x$, then the sparse approximation is at least as hard as a general constraint satisfaction problem. Indeed, for fixed constants $C, K \geq 1$, it is NP-hard to produce an $(S)$-sparse approximation whose error lies within a factor $K$ of the minimal $s$-term approximation error[63]. Nevertheless, over the past decade, researchers have identified many interesting classes of sparse approximation problems that submit to computationally tractable algorithms. These striking results help to explain why sparse approximation has been such an important and popular topic of research in recent years. In practice, sparse approximation algorithms tend to be slow unless the dictionary admits a fast matrix vector multiply. Let us mention two classes of sparse approximation problems where this property holds. First, many naturally occurring signals are compressible with respect to dictionaries constructed using principles of harmonic analysis [64] (e.g., wavelet coefficients of natural images).
This type of structured dictionary often comes with a fast transformation algorithm. Second, in compressive sampling, we typically view as the product of a random observation matrix and a fixed orthogonal matrix that determines a basis in which the signal is sparse. Again, fast multiplication is possible when both the observation matrix and sparsity basis are structured.

Recently, there have been substantial efforts to incorporate more sophisticated signal constraints into sparsity models. In particular, Baraniuk et al. have studied model-based compressive sampling algorithms, which use additional information such as the tree structure of wavelet coefficients to guide reconstruction of signals [65]. Successful exploitation of sparse representation in various vision application has lead to more sophisticated research over sparse reconstruction where it can be shown that sparse representation is able to recover the outlying subspace among a subset of data points[41]. Here in the following we will briefly introduce the existing methods to solve sparse representation

**Major Algorithmic Approaches:**

1. **Greedy pursuit.** Iteratively refine a sparse solution by successively identifying one or more components that yield the greatest improvement in quality [7].

2. **Convex relaxation.** Replace the combinatorial problem with a convex optimization problem. Solve the convex program with algorithms that exploit the problem structure [1].

3. **Bayesian framework.** Assume a prior distribution for the unknown coefficients that favors sparsity. Develop a maximum a posteriori estimator that incorporates the observation. Identify a region of significant posterior mass [8] or average over most-probable models [9].

4. **Nonconvex optimization.** Relax the $l^0$ problem to a related nonconvex problem and attempt to identify a stationary point [10].

5. **Brute force.** Search through all possible support sets, possibly using cutting-plane methods to reduce the number of possibilities.

It is shown that selection of proper $D$ leads to variety of application on sparse representation. Later in chapter 5 we will show how it can be learned for regression and classification considering contextual information in the image.
Chapter 2

Parity Subspace for Human Interaction Recognition

2.1 Introduction

Handling the complexity of the variability of high dimensional data that represent activities, which is inherently multidimensional in applications like temporal segmentation and recognition of continuous activities from a video, is a core problem to address for future intelligent systems to extract and manage content fully automatically. Unfortunately, despite a significant amount of research focusing on recognizing actions and activities on high dimensional video data, the problem of their time localization has received considerably less attention [4]. Even more so, if the localization has to be performed online, and enable the processing of video in real-time.

In this chapter, we propose an online approach to coping with the high dimensions of the data, as well as the complexity of their variability by combining notions from two well-understood theories and formalisms. The first one is the theory on reproducing kernel Hilbert spaces [33], and the second is the theory of state space models [34]. Exploiting the power of kernels allows a flexible and effective blending of heterogeneous high-dimensional features which can be mapped into a suitable Hilbert space where they can easily be modeled, even with linear models. Exploiting the theory on state space models allows borrowing a number of well-understood results about their estimation, and their power for doing analysis, recognition, and detection based on multidimensional temporal sequences. The resulting approach allows to extend the notion of parity space, developed within the context of detection based on linear models [35], for its use together with kernel regression, and kernel state space models, which are the Hilbert space counterparts of the linear versions. Rather than using Euclidean geometry to project data onto the parity space and reveal a detection, we exploit the geometry of linear operators in Hilbert space and derive closed-form solutions for
the computation of normalized test statistics, based solely on kernel evaluations. The framework is suitable to work online through the use of an incremental temporal window, and online parameter estimation techniques, such as online kernel PCA [36, 37], and recursive least squares [34]. The resulting approach allows to extend the notion of parity space, developed within the context of detection based on linear models [35], for its use together with kernel regression, and kernel state space models, which are the Hilbert space counterparts of the linear versions. Rather than using Euclidean geometry to project data onto the parity space and reveal a detection, we exploit the geometry of linear operators in Hilbert space, and derive closed form solutions for the computation of normalized test statistics, based solely on kernel evaluations. The framework is suitable to work online through the use of a temporal incremental window, and online parameter estimation techniques, through which real-time performance can be achieved.

The framework based on kernel state space models allows to account for the temporal correlation of activities, and can easily be extended to do recognition [66, 67]. In particular, we are interested in recognizing binary human interactions. Those can be represented by temporal sequences, and require the use of pairwise kernels to model the symmetry of their space [68]. Here we augment the recognition approach introduced in [67, 68], and combine it with the temporal segmentation to obtain an online segmentation and recognition framework. Modeling and recognizing interactions is a relatively unexplored area, given that datasets have started to become available only recently [69, 70]. Therefore, to test our combined segmentation and recognition framework, we collected a new, large, and challenging dataset of binary human interactions, over which we extensively test the proposed approach.

The resulting approach allows to extend the notion of parity space, developed within the context of detection based on linear models [35], for its use together with kernel regression, and kernel state space models, which are the Hilbert space counterparts of the linear versions. Rather than using Euclidean geometry to project data onto the parity space and reveal a detection, we exploit the geometry of linear operators in Hilbert space, and derive closed form solutions for the computation of normalized test statistics, based solely on kernel evaluations. The framework is suitable to work online through the use of a temporal incremental window, and online parameter estimation techniques.

In the following of this chapter sections 2.3 and 2.4 introduce the kernel regression, and kernel state space models, and generalize the use of the parity space with kernels. Finally, Section 5.4 validates the proposed approach by achieving very promising results.
2.2 Literature Review

Kernel state space models have been first introduced in [66] for dynamic texture recognition and subsequently have been used in [67, 68] for action and interaction recognition. Compared to those works, we introduce a theoretically grounded framework that extends those models for temporal segmentation. There is a body of work that exploits kernel-based methods to solve the two-sample test problem, and that applies this framework to the change-point detection problem (see [71] and references therein). Those approaches have been shown to work with either univariate temporal sequences or with small-dimensional sequences. However, in [72] they indicate that the maximum mean discrepancy (MMD) distance [73] can be used for the online temporal segmentation of actions. Mainly, our framework differs from theirs because we can also account for the temporal correlation of sequences, and Section 5.4 also shows that even the simpler kernel regression model outperforms the MMD. Finally, this work also relates to the elegant framework introduced in [74]. Compared to them we do not focus on the combined segmentation and early recognition. Instead, we offer a fast, unified model to perform segmentation and recognition right at the conclusion of an interaction, given that for real-time analysis the idea of early detection often has less importance outside of applications such as affective computing.
2.3 Kernel Regression Models

Given a temporal sequence \( \{ y_t \} \), assuming values in a space \( S \), which in general may not be Euclidean, let us consider the Mercer kernel \( \kappa(y_t, y'_t) = \langle \phi(y_t), \phi(y'_t) \rangle \), where \( \phi(\cdot) \) is mapping \( S \) to \( \mathcal{H} \), a Reproducing Kernel Hilbert Space (RKHS) [33]. We assume that \( \{ y_t \} \) is mapped to a sequence \( \{ \phi(y_t) \} \), which can be expressed with the following kernel regression (KR) model

\[
\phi(y_t) = C x_t + w_t . \tag{2.1}
\]

The quantity \( C \) may not be a matrix but a linear operator \( C : \mathbb{R}^n \rightarrow \mathcal{H} \), acting on the regressor \( x_t \in \mathbb{R}^n \) at time \( t \), to account that \( \mathcal{H} \) could be an infinite dimensional space. Indeed, \( C \) can be represented as

\[
C = [c_1, \ldots, c_n], \quad C x = \sum_{i=1}^{n} c_i x_i, \quad \text{where} \quad x = [x_1, \ldots, x_n]^\top .
\]

The observation noise \( w_t \) is modeled as a zero-mean Gaussian process.

Inspired by the successful concept of parity space, developed within the context of fault detection applications based on linear models [35], we are interested in extending that approach to be used with kernels. We do so by introducing the concept of kernel parity Hilbert space (KPHS), which is the subspace of \( \mathcal{H} \) defined as

\[
P = \{ v \in \mathcal{H} | \langle c_i, v \rangle = 0, i = 1, \ldots, n \} .
\]

We also indicate with \( P \) the operator that projects a vector \( v \in \mathcal{H} \) onto \( P \), given by \( P v \), whereas \( \xi_t = P \phi(y_t) \) is called kernel parity vector.

If a temporal sequence \( \{ y_t \} \) is made of i.i.d. samples and is modeled by (2.1), then for an input sample \( y_t \), the kernel parity vector \( \xi_t \) indicates in what direction and by how much the sample does not belong to the span of the \( \{ c_i \} \). In particular, since it is also true that \( \xi_t = P w_t \), \( \xi_t \) tells, in feature space \( \mathcal{H} \), by how much the measurement noise has spilled into \( P \) in order for model (2.1) to hold. This fact is suggesting that if we knew the noise model, monitoring \( \xi_t \) would reveal whether the current sample \( y_t \) implies a noise model different than the one given, which means that model (2.1) should no longer hold.

Let’s now consider the residual error \( e_t = \phi(y_t) - C \hat{x}_t \), where \( \hat{x}_t \) is the maximum likelihood estimation of the regressor, given the observation \( y_t \), and the model given by \( \kappa \) and \( C \). Under the hypothesis of the noise \( w_t \) being i.i.d. realizations from an uncorrelated stationary Gaussian process, which means that its autocorrelation function is given by \( \sigma^2 \delta \), where \( \delta \) is a Dirac distribution defined over a suitable domain, the maximum likelihood estimation \( \hat{x}_t \) coincides with the least squares estimation

\[
\hat{x}_t = \arg \min_x \| \phi(y_t) - C x \|^2 . \tag{2.2}
\]

Under the hypothesis expressed above we now show how starting from the samples \( y_1, \ldots, y_T \), and from the kernel \( \kappa \), it is possible to estimate model (2.1), connect the residual error to the kernel parity vector, and define a rule to establish whether the sample \( y_t \) is in accordance with model (2.1).
The obvious choice to model the variability of \( \{y_t\} \) in feature space is to apply Kernel PCA (KPCA) [33]. To this end it is convenient to introduce the notation \( \Phi \equiv [\phi(y_1), \cdots, \phi(y_T)] \), and the kernel matrix \( K \equiv \Phi^T \Phi \), where \( [K]_{st} = \kappa(y_s, y_t) \). KPCA evaluates \( T \) kernel principal components out of a linear combination of the elements of \( \Phi J \), where \( J \equiv (I - \frac{1}{T}ee^T) \) is the so-called centering projection matrix, and \( e = [1, \cdots, 1]^T \in \mathbb{R}^T \). The linear combination coefficients are computed from the eigen-decomposition of \( K \), after assuring that data in feature space has zero-mean. This is done by computing \( JKJ \equiv \alpha\Lambda \alpha^T \), where \( \Lambda \equiv \text{diag}(\lambda_1, \cdots, \lambda_T) \) and \( \alpha \) are the eigenvalue and eigenvector matrices. The set of orthonormal kernel principal components can be expressed as \( \Phi J \alpha \Lambda^{-\frac{1}{2}} \). Assuming that \( \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_T \), in order to model the highest amount of data variability in feature space with only \( n < T \) components, it is well known that the first \( n \) have to be picked. If \( \beta \equiv \alpha \Lambda^{-\frac{1}{2}} \) indicates the first \( n \) kernel principal components coefficients obtained by removing the columns of \( \Lambda \) after the first \( n \), we set the observation operator of model (2.1) to 

\[
\hat{C} \equiv \Phi J \beta . \tag{2.3}
\]

For the noise model we notice that \( \sigma^2 \equiv E[\langle w_t, w_t \rangle] \), and its sample estimation is available from the KPCA, which is given by \( \hat{\sigma}^2 = \frac{1}{T} \sum_{t=1}^{T} \|w_t\|^2 = \frac{1}{T} \sum_{i=n+1}^{T} \lambda_i \).

It is now possible to relate the residual error to the kernel party vector. We start by plugging (2.3) into (2.2), we remove the mean of the model \( \frac{1}{T} \Phi e \), and after expanding the square, taking the derivative with respect to \( x \), and setting it to zero one can obtain 

\[
\hat{x}_t = \beta^T J \tilde{\kappa}(y_t) , \tag{2.4}
\]

where \( \tilde{\kappa}(y_t) \equiv (\tilde{\kappa}(y_t) - \frac{1}{T} K e) \), and \( \tilde{\kappa}(\cdot) \equiv [\kappa(y_1, \cdot), \cdots, \kappa(y_T, \cdot)]^T \). Moreover, by combining (2.4) and (2.2) we can see that \( \min_x \|\phi(y_t) - \frac{1}{T} \Phi e - C x\|^2 = \|P_{C^\perp} (\phi(y_t) - \frac{1}{T} \Phi e)\|^2 \), where \( P_{C^\perp} \) is the projection operator defined by 

\[
P_{C^\perp} = I - \Phi J \beta \beta^T J \Phi^T , \tag{2.5}
\]

where \( I \) here indicates the identity operator. Thus, we can say that \( e_t = P_{C^\perp} (\phi(y_t) - \frac{1}{T} \Phi e) \), and by construction \( P_{C^\perp} \) represent an orthonormal projection onto the orthogonal complement of the span of the \( \{c_i\} \), and therefore it is equivalent to \( P_{\mathcal{F}} \). In particular, we have \( \|e_t\|^2 = \|\xi_t\|^2 \). Note that everything so far has been derived under the hypothesis of \( w_t \) being an uncorrelated stationary Gaussian process, which is an idealized scenario. If, for instance, the noise is correlated, the autocorrelation function is not a Dirac delta, and the residual error should be estimated with generalized least squares. This means that this derivation and implementation become more complex because the autocorrelation function needs to be estimated.
Finally, the fact that $\|e_t\|^2 = \|\xi_t\|^2$ suggests that the criterion for establishing whether or not the new sample $y_t$ is in accordance with model (2.1) is to simply check if the normalized residual error $\|e_t\|^2 / \sigma^2$ is lower or greater than a threshold $\nu$, appropriately chosen. Also, note that through (2.5) the analytical expression of $\|e_t\|^2$, function only of the kernel $\kappa$ is readily available, which is given by

$$\|e_t\|^2 = \kappa(y_t, y_t) - \frac{1}{T} \bar{e}^\top (\bar{\kappa}(y_t) + \bar{\kappa}(y_t)) - \bar{\kappa}(y_t)^\top J \beta \beta^\top J \bar{\kappa}(y_t).$$  

(2.6)

### 2.4 Kernel State Space Models

The kernel regression model assumes that the samples of the temporal sequence $\{y_t\}$ are i.i.d. When those are correlated, the regressor temporal sequence $\{x_t\}$, also referred to as the *state* sequence, can be modeled by an autoregression, thus obtaining a *kernel state-space (KSS)* model, given by

$$\begin{align*}
x_{t+1} &= A x_t + v_t, \\
\phi(y_t) &= C x_t + w_t.
\end{align*}$$

(2.7)

Here the new elements of the model are $A \in \mathbb{R}^{n \times n}$, which describes the dynamics of the state evolution, and the system noise $v_t$, which is zero-mean i.i.d. Gaussian distributed with covariance $Q$, and independent from $w_t$.

Along the lines of the previous section, we introduce an extension of the concept known as observability matrix in the theory of linear dynamical systems (LDS). Specifically, we consider the linear operator $O_t : \mathbb{R}^n \rightarrow \mathcal{H}^\tau$, mapping $x$ to $O_t x$, where $O_t = [C^\top, A^\top C^\top, \cdots, A^\top^{T-1} C^\top]^\top$. Based on this we extend the definition of KPHS into *kernel parity Hilbert space of order $\tau$ (KPHS-$\tau$)*, which is the subspace of $\mathcal{H}^\tau$ defined as $P_\tau = \{v \in \mathcal{H}^\tau | v^\top O_t = 0\}$.

Following the standard representation used in system identification [34], we define the vectors $\Phi_{t-\tau+1}^t = [\phi(y_{t-\tau+1})^\top, \cdots, \phi(y_t)^\top]^\top, W_{t-\tau+1}^t = [w_{t-\tau+1}^\top, \cdots, w_t^\top]^\top, V_{t-\tau+1}^t = [v_{t-\tau+1}^\top, \cdots, v_t^\top]^\top$, and the following matrix

$$\tilde{O}_\tau = \begin{bmatrix} 0 & \cdots & \cdots & 0 \\ \vdots & \cdots & \vdots & \vdots \\ 0 & \vdots & \vdots & O_{\tau-1} \end{bmatrix},$$

(2.8)

with which model (2.7) can be rewritten as

$$\Phi_{t-\tau+1}^t = O_t x_{t-\tau+1} + \tilde{W}_{t-\tau+1}^t,$$

(2.9)

where $\tilde{W}_{t-\tau+1}^t = \tilde{O}_\tau V_{t-\tau+1}^t + W_{t-\tau+1}^t$ is a zero-mean Gaussian process noise with autocorrelation
matrix function \( \tilde{O}_\tau I_\tau \otimes Q \tilde{O}_\tau^T + I_\tau \otimes \sigma^2 \delta \), and \( \otimes \) indicates the Kronecker product.

As in Section 2.3, \( P_\tau \) is the operator that projects a vector \( v \in \mathcal{H}^\tau \) onto \( P_\tau \), given by \( P_\tau v \), whereas \( \Xi_{t-\tau+1} = P_\tau \Phi_{t-\tau+1} \) is the kernel parity vector. According to the definition of KPHS-\( \tau \), \( \Xi_{t-\tau+1} = P_\tau \tilde{W}_{t-\tau+1}^t \) which shows that it is independent from the state \( x_{t-\tau+1} \), and it can be interpreted with respect to \( y_{t-\tau+1}, \cdots, y_t \) exactly in the same way as \( \xi_t \) is interpreted with respect to \( y_t \).

We now shift the attention to the reconstruction error \( E_{t-\tau+1}^t = \Phi_{t-\tau+1} - \hat{O}_\tau \hat{x}_{t-\tau+1} \), where \( \hat{x}_{t-\tau+1} \) is the maximum likelihood estimation of \( x_{t-\tau+1} \), and we make the further simplifying assumption that the autocorrelation matrix function of \( \tilde{W}_{t-\tau+1}^t \) is given by \( I_\tau \otimes \sigma^2 \delta \). This allows to compute \( \hat{x}_{t-\tau+1} \) with a simple least squares estimation. We also assume that model (2.7) is learned from data as it is done in Section 2.3, and the reader is referred to [66] for the procedure to estimate the parameters \( A \) and \( Q \). Under these conditions it is easy to show that

\[
\hat{x}_{t-\tau+1} = \left( \sum_{i=0}^{\tau-1} A_i^T A_i \right)^{-1} \sum_{i=0}^{\tau-1} A_i^{T-1-i} \beta^T J \tilde{k}(y_{t-i}) . \tag{2.10}
\]

From Equation (2.10) one can compute the projection operator \( P_{O^\perp} \) such that \( E_{t-\tau+1}^t = P_{O^\perp} (\Phi_{t-\tau+1} - e_\tau \otimes \frac{1}{T} \Phi e) \), where \( e_\tau \) is a column vector with \( \tau \) ones, which is given by

\[
P_{O^\perp} = I - O_\tau \left( \sum_{i=0}^{\tau-1} A_i^T A_i \right)^{-1} O_\tau^T . \tag{2.11}
\]

By construction \( P_{O^\perp} \) represents an orthonormal projection onto the orthogonal complement of the span of the columns of \( O_\tau \), and therefore it is equivalent to \( P_{\tilde{P}_\tau} \), and in particular \( \| E_{t-\tau+1}^t \|^2 = \| \Xi_{t-\tau+1}^t \|^2 \). As in Section 2.3 this is true under the hypothesis of \( \tilde{W}_{t-\tau+1}^t \) being an uncorrelated stationary Gaussian process, which is an idealized scenario.

Similarly to the regression model, the criterion for establishing whether or not the trajectory \( y_{t-\tau+1}, \cdots, y_t \) is in accordance with model (2.7) is to simply check if the normalized residual error \( \| E_{t-\tau+1}^t \|^2 / \tau \sigma^2 \) is lower or greater than a threshold \( \nu \), appropriately chosen. By using (2.11) the analytical expression of \( \| E_{t-\tau+1}^t \|^2 \), function only of the kernel \( \kappa \) can be computed in closed
form and is given by

\[ \|E_{t-\tau+1}^t\|^2 = \sum_{i=0}^{\tau-1} \kappa(y_{t-i}, y_{t-\tau-i}) - \frac{1}{T} e^\top (\tilde{\kappa}(y_{t-i}) + \kappa(y_{t-\tau-i})) \]
\[ - \sum_{i=0}^{\tau-1} \tilde{\kappa}(y_{t-i}) J \beta A^{(t-\tau-1)-i} \left( \sum_{i=0}^{\tau-1} A^i A^i \right)^{-1} \]
\[ \cdot \sum_{i=0}^{\tau-1} A^{(t-\tau-1)-i} \beta^\top \tilde{\kappa}(y_{t-\tau-i}) , \]  

As expected, we notice that when \( \tau = 1 \), Equations (2.10), (2.11), and (2.12), collapse to (2.4), (2.5), and (2.6), respectively.

## 2.5 Online Temporal Segmentation

In this section we apply the framework developed in Sections 2.3 and 2.4 to the problem of segmenting online a time series. We then apply the approach to the segmentation of human interactions. We deploy a temporal incremental window \([3, 72]\), and sequentially detect segmentation cuts.

Let’s assume that in monitoring a temporal sequence \( \{y_t\} \), the last segmentation cut was observed at time \( s < t \), where \( t \) is the current time. We want to test whether at time \( t - \tau \) a new cut should be detected. To this end either a kernel regression model (2.1), or a kernel state space model (2.7) is estimated from the data in the training time window \([s+1, \cdots, t-\tau]\), of length \( T_t = t - \tau - s \), i.e. \( y_{s+1}, \cdots, y_{t-\tau} \). See Figure 2.1. A cut should be detected if the data observed in the subsequent test time window \([t-\tau+1, \cdots, t]\), i.e. \( y_{t-\tau+1}, \cdots, y_t \), and the model previously estimated do not fit “well enough”.

The geometric framework introduced in Sections 2.3 and 2.4 tells us to project the test data onto the kernel parity Hilbert space (KPHS for KR models, or KPHS-\( \tau \) for KSS models), and compare
this projection with the noise model to decide whether data and model can fit. More formally, for
the KR model one should compute the following statistic
\[ ε_{t-τ}^{KR} = \frac{1}{τσ^2} \sum_{i=0}^{τ} \|e_{t-i}\|^2 , \] (2.13)
whereas for the KSS model one should compute
\[ ε_{t-τ}^{KSS} = \frac{1}{τσ^2} \| E_{t-τ+1} \|^2 . \] (2.14)
Finally, \( ε_{t-τ}^{KR} \) and \( ε_{t-τ}^{KSS} \) can be used to test the hypotheses “yes cut”, i.e. \( H_1 \), versus “no cut”, i.e. \( H_0 \). In particular,
\[ ε_{t-τ} \leq ν \Rightarrow H_0 \text{ is true}, \quad ε_{t-τ} > ν \Rightarrow H_1 \text{ is true} . \] (2.15)
If \( H_0 \) is true, test (2.15) is repeated at time \( t + Δt \). If \( H_1 \) is true, the next test is performed at time \( t + τ \), with a training time window that restarts with length \( T_{t+τ} = τ \).

### 2.5.1 Human Interaction Segmentation

The main advantage of exploiting the kernel parity Hilbert space is that data belonging to a multi-
dimensional non Euclidean space \( S \) can be modeled effectively in the RKHS \( H \) through the KR, or
the KSS model. This is the typical scenario encountered when dealing with sequences of features
describing activities.

We are particularly concerned with segmenting and recognizing human interactions happening
between two individuals, \( a \) and \( b \), observed in video. For this we borrow the representation de-
scribed in [68], where an interaction is described by a segment of a temporal sequence \( \{y_t\} \). The features \( y_t \) are obtained by tracking person \( a \) and \( b \), and by aggregating their distance \( d_t \), together with histograms \( a_t \) and \( b_t \), describing the body motion of person \( a \) and person \( b \), respectively, so that \( y_t = [a_t, b_t, d_t]^\top \). Given that the interaction between persons \((a, b)\) should be the same of the interaction between persons \((b, a)\), in [68] they design, so-called pairwise kernels [75] that account for this special symmetry, as well as for the geometric structure of the input space \( S \), which indeed is non-Euclidean. In particular, we use the kernel

\[
\kappa((a, b, d), (a', b', d')) = \kappa^{TL}((a, b), (a', b'))e^{-\gamma(d-d')^2}
\]

where \( \kappa^{TL} \) is the tensor learning pairwise kernel [75], which is reported to be the best performer. The constant \( \gamma \) is estimated with cross-validation. For further details about how to compute the features and the precise expression of \( \kappa^{TL} \), given space constraints we refer the reader to [68].

### 2.6 Recognition

Given a temporal sequence segment \( Y_{s:t} = [y_s, \cdots, y_t] \), obtained with the online segmentation of Section 2.5, since we are interested in recognizing human interactions, characterized by a temporally correlated sequence, we assume it can be modeled by a KSS model. Therefore, recognizing a segment entails comparing KSS models. For the linear case, where the KSS degenerates to a linear dynamical system (LDS), and \( S \) is Euclidean, methods for comparing LDSs include geometric distances, algebraic kernels, and information theoretic metrics [67]. When \( S \) is a non-Euclidean space it is possible to compare KSS models through the use of Binet-Cauchy kernels [76]. In particular, [67] describes their use for action recognition when the input features are a temporal sequence of histograms, and [77] uses them for modeling and recognizing binary temporal sequences. Since
Table 2.1: Rand index. This is a measure of the similarity between two data clustering. We computed the (RI) of the interaction segmentations against the ground-truth labels. A higher RI means better interaction localization.

<table>
<thead>
<tr>
<th></th>
<th>RI</th>
<th>KSS</th>
<th>KR</th>
<th>MMD</th>
</tr>
</thead>
<tbody>
<tr>
<td>HAUS</td>
<td>0.72</td>
<td>0.71</td>
<td>0.70</td>
<td></td>
</tr>
<tr>
<td>UT</td>
<td>0.72</td>
<td>0.69</td>
<td>0.68</td>
<td></td>
</tr>
</tbody>
</table>

Our implementation framework is based on the features of [68], we apply the Binet-Cauchy kernel that they refer to as $\kappa_{NLDS}$ and which embeds kernel (2.16). Differently from [68], from $\kappa_{NLDS}$ we form a Gaussian kernel based on the derived kernel distance, which we found to be more effective, and that is given by

$$\kappa^{KSS}(Y, Y') = e^{-\eta(\kappa_{NLDS}(Y,Y') + \kappa_{NLDS}(Y',Y') - 2\kappa_{NLDS}(Y,Y'))}$$

With the above kernel we use the libSVM [78] to train a multiclass SVM classifier. For further details about the computation of $\kappa_{NLDS}$, the reader is referred to [67, 68].

2.7 Experiments

We tested our approach on the UT-Interactions dataset [69] and on the HAUS-PI dataset. A new dataset that we collected and that we plan to release to the public.

**UT-Interaction Dataset:** It contains five human-human interaction classes: handshake, hug, kick, punch and push. We developed an annotation tool that allows to draw boxes, track individuals, and assign labels indicating when an interaction starts and ends. We indicate those time windows as active parts, as opposed to the inactive when an interaction is not happening.

**HAUS-PI:** We collected the Human Activities Under Surveillance – Person Interaction (HAUS-PI) dataset. In our experiments we considered 12 person interaction classes: handshake, hugging, high-five, kicking, punching, pushing, slapping, bowing, waving, shooting, stabbing, patting. Since the persons were allowed to enter the scene from any direction, the interactions are recorder with a very high viewpoint change variation. Also, the number of classes collected and the number of samples per class (roughly 50), makes this a very challenging dataset. The length of the sequences is enough to be able to learn an detect the beginning and the ending of the interaction. The original frame resolution is 1280x720. The camera calibration is available, and with our annotation tool we annotated the data and obtained labeled tracks calibrated with respect to the ground plane.

We mostly follow the evaluation protocol introduced in [74]. We compare three methods: the approach described in [72], indicated as MMD, the KR and the KSS model. As expected the KSS is clearly the best performer on bot UT-Interaction as well as the HAUS-PI datasets. Also, even though the KR model works under the same assumptions as the MMD, we do observe better
performance. Another measure for the localization accuracy is given by the Rand index, which is included in Table 2.1. Even according to this measure we obtain a similar comparison between the models.

Figure 2.5 plots the normalized time to detection (see [74] for definition), which indicates the timeliness with which the beginning of an interaction is identified. In particular, the left and the center plot clearly show that the KSS approach is much less sensitive to the length $\tau$ of the test time window.

The experimental results illustrates that the proposed model is not even more accurate than the state of the art methods, it has less latency compared to those methods in online detection applications. All the results indicate that the proposed framework is very promising, and can become an important part of a system for the analysis in realtime of human behavior from video.
Chapter 3

Uniform Invariant Subspace

3.1 Introduction

Recent approaches based on sparse representation [12] and low-rank matrix decomposition [62] have demonstrated significant potential for addressing the problem of human identification, based on matching face images. So much so that sparse coding has led to impressive performance even for image classification [53, 79], and also low-rank methods, after being applied to domains such as segmentation and grouping [80], tracking [46], and 3D visual recovery [81], now are being used also for classification [45]. For face recognition the sparse representation based classification (SRC) method [12] has shown robustness for a high degree of noise and occlusions in the test images. At the same time, sparse coding dictionary learning was shown to be sensitive to training samples corrupted by structural nuisance factors, such as occlusions, disguise, pose, lighting variations, and so on. This has motivated the development of low-rank matrix decomposition approaches [43, 62, 82], which have the ability to learn a representational dictionary even in the presence of corrupted data. Those methods build a generative representation of the data that focuses on capturing all the information descriptive of an entity. This leads to complex training and testing for building robustness against, and filter out unwanted data variations due to nuisance factors.

In this chapter, we introduce a low-rank modeling framework that gives up capturing all the descriptive information of an entity (referred to as the sufficient component) and focuses on learning a representation that is invariant to nuisance factors (referred to as the invariant component). The main advantage of this approach is a fast procedure for computing and comparing invariant components for recognition. Indeed, we will see that a simple matrix multiplication can achieve this. On the other hand, the main challenge of this approach is that different entities may originate the same invariant component, thus preventing their discrimination. We will show that the pro-
posed framework not only learns different invariant representations for various entities, but such representations promote a uniform inter-class separation.

Human identification in unconstrained scenarios, typical of the video surveillance settings, implies that images characterizing the identity of an individual are being captured “on the field,” without subject cooperation. Those may include face images, but also images of the entire body may be acquired for person re-identification across camera field of views [83]. The scenario is such that images forming the identities in the gallery, as well as the probe images may be severely contaminated by structural nuisance factors. While this has been the natural testbed for low-rank methods applied to the particular scenario of human identification based on face [43], their use based on the whole body appearance for person re-identification has not yet been attempted.

So far, every low-rank matrix decomposition approach developed for human identification has made the assumption that images representing an identity lay in the space spanned by the columns of a low-rank matrix, and that a sparse matrix models nuisance factors and remaining noise. In this chapter, we propose to explore the use of law-rank matrix decompositions in quite a different framework. Rather than attempting to reconstruct the subspace span by each identity from corrupted data, we propose to learn the common subspace of the nuisance factors that have caused the corruption.

The approach couples simple geometry tools with recent advances in low-rank matrix recovery theory [62], and develops a supervised model for learning the proposed invariant representation, which spans an invariant subspace. Such subspace has to be orthogonal to the variation subspace, generated by data variation induced by nuisance factors on all the entities. We make the assumption that the variation subspace is low-rank. Although this is an approximation, we empirically verify that it leads to very promising results for face recognition when training and testing data are highly corrupted, which is typical in video surveillance applications. Besides, we also challenge the approach by using it for person re-identification, where whole body images are matched for identifying people across camera field of views [83].

While the framework is grounded on geometry, we will show how it relates to metric learning [84, 85, 86, 87], typically used for improving nearest neighbor (NN) classification based on the Euclidean distance. We will show that learning the invariant components is equivalent to learning the representatives of a set of entities (or classes); thus classification is based on identifying the nearest invariant component. Less intuitively, the same invariant components define a global metric, and more importantly, they also define a local metric. This is important because local metric learning approaches [88, 89, 90, 91], improve upon global ones by taking into account the variability of the discriminative power of features across different neighborhoods. In particular, most of the approaches learn local metrics for different neighborhoods independently and use reg-
ularization to avoid overfitting. Our framework learns the invariant components, and therefore the local metrics, jointly, and in a way that promotes uniform inter-class separation. To the best of our knowledge, this is the first approach for metric learning based on a low-rank framework.

To achieve this we propose a novel low-rank matrix decomposition model, which is augmented with adequate regularizing constraints for the simultaneous learning of the nuisance subspace, and a gallery representation lying in its orthogonal complement, and that takes into account that images of the same identity should generate equal representations. Learning is performed by leveraging the Augmented Lagrange Multipliers framework [92]. We have applied this idea to face recognition with corrupted training images, and have found our method to yield very promising results. We have also attempted to challenge the approach by applying it to a couple of person reidentification datasets that were providing enough gallery images. Although the approach is not designed to work in those conditions, where there are limited training samples per person and vast amounts of misalignments, we still found our approach not to deteriorate the performance very significantly.

The rest of the chapter will build more connections and differentiations with the related literature by taking advantage of the introduced notation. In addition, Section 4.3.2 introduces the idea of invariant subspace. Section 3.4 highlights its advantages and challenges for classification. Section 3.5 describes a supervised model for training. Section 3.6 shows how classification is done, its properties, and defines the global and local metrics being learned. Finally, Section 5.4 validates the proposed approach.

3.2 Related Work

During recent years sparse representation and low-rank methods have proven to be useful in many application in computer vision. Low-rank matrix recovery from noisy data has been proposed by Wright in a convex optimization framework based on iterative thresholding of non-convex cost functions [12], and subsequently, the Accelerated Proximal Gradient Approach was introduced [93, 94]. The Augmented Lagrangian Multiplier was introduced to accelerate the convergence further up to 5 times [92], compared to APG. Since then it has been applied and extended in several applications in image processing and computer vision with promising results [43, 45, 95, 96, 97, 98].

Several works have shown how the combination of low-rank and sparse modeling of the pixel values of an image with parametric transformations of the image domain can be used for holistic symmetry detection and rectification. [38] has used low-rank matrix decomposition to learn intrinsic invariant low-rank texture of objects and extract linear transformation of the 3-D scene over associated planar region. Using this new proposed method they were able to overcome the lack of
invariant image features under projective transformation, where it enable to recover the exact low-rank structure by simultaneously discarding sparse noise and transforming the feature space. The resulting tool, called Transform Invariant Low-rank Texture (TILT), has been profitably applied to practical problems such as urban 3D reconstruction [39], calibration [98], and optical character recognition [99]. Unlike [38] which was seeking for a transfer on a single image [40] attempt to find the robust alignment for closely distributed data points.

[41] introduced the low-rank subspace clustering (LRSC) in a convex formulation using the idea of adding the self-expressiveness constraint into low-rank decomposition. Vidal built the (LRSC) method on the idea that clean data points lying on the same subspace can be expressed as linear combination of themselves, therefore by directly imposing the identity of the dictionary to its multiplication with a symmetric matrix, they were able to decompose subspace clusters in an unsupervised manner from noisy data. [100] added a fixed rank constraint to the problem of low-rank representation, and proved that it can be used for feature extraction.

[101] has described the theory and applications for sparse subspace clustering in contrast to [41] which is devoted to low-rank subspace clustering. Vidal in [42] introduced a dimension reduction method where it learns a projection matrix and a sparse coefficients matrix to reduce the dimension of the data points jointly and does the unsupervised clustering in a single objective function. [44] used an structured matrix to do supervised clustering of data points, where with a new goal function they decompose data points to a low rank and sparse noise where the low-rank matrix is the linear combination of the basis in a dictionary and using a structured matrix they force subjects from the same category lay in same subspace.

[102] used low-rank representation in a transformed source domain to explore the linear correspondences between target samples and source samples. They used a basis transformation matrix to transfer samples from the source domain to target domain, then using low-rank decomposition they reconstructed transferred samples using target samples. They never used source sample labels; therefore this process is unsupervised and was used to expand limited training samples with those from the internet.

[82] proposed a decomposition method for a given face image to model the nuisance factors (i.e. illumination change, large occlusions, etc) by a low-rank matrix and each image specific changes by a sparse noise matrix. Finally, they learned a classifier for each class to classify core component of each class. Unlike our method that we reshape each image in form of a vector and ensemble all data point in a matrix and proceed by decomposition of obtained matrix, [82] propose their decomposition on a matrix that is the same size of the image; therefore the low-rank matrix only captures the gist of each image that might coincide across all images. [43] performed face recognition by adding to the basic low-rank model an incoherence regularizer to enforce low-
rank matrixes from different subjects to be independent. [103] used ALM to learn a low rank and
discriminative dictionary. [45] similar to [44] used a structured multiplier to boost the discrimina-
tive power of a low-rank matrix, only to subsequently fitting another regressor for classification. 
Unlike [45] our method does not need to learn a dictionary, which requires a nested loop for dic-
tionary optimization, and during testing, the data does not need to be represented according to the
dictionary, but it is quickly projected onto the subspace of gallery identities, embedded into the
orthogonal complement of the structural nuisance subspace. x As in this section, we will conduct
experiments on person reidentification datasets, we briefly review the literature on person reiden-
tification [83]. The common setting is based on a gallery of images acquired in unconstrained
scenarios, with images of people appearing from different viewpoints, and under different pose
and illumination conditions. The degree of alignment is typically much less compared to the case
of face recognition, so attempting to apply our approach to such a scenario, and directly on the
raw pixels, really challenges the approach. A fair amount of state-of-the-art approaches relies on
a feature descriptor to perform the identity matching, like [104, 105, 106, 107]. Machine learning
approaches to reidentification instead have been proposed for learning salient features [108], attri-
butes [109], and ranking functions [110]. Metric learning methods perform very well, like Large
Margin Nearest Neighbors (LMNN) [111], Metric Learning by Collapsing Classes (MCC) [112],
Probabilistic Relative Distance Comparison (PRDC) [113], and Pairwise Constrained Component
Analysis (PCCA) [114]. In our experiments, we challenge our approach by comparing it against a
metric learning method, despite the fact that we have not exploited the possibility to use a dedicated
descriptor or to do an extensive learning with a dedicated dataset, and the fact that the proposed
approach was forced to handle massive amounts of misalignments.

3.3 Low Rank Decomposition of Nuisance Factors

We assume that a data point $x \in \mathbb{R}^m$, representing an entity (e.g., the vectorized version of the
image pixels of a face), can be modeled by two additive components. The first one, $s \in \mathbb{R}^m$, 
represents all the information necessary to recognize the entity (e.g., everything that describes the
specific identity of the individual depicted by the face image). From a statistical point of view, we
can imagine $s$ to be the equivalent of a sufficient statistic for recognition, and we refer to it as the
sufficient component. The second component, $v \in \mathbb{R}^m$, is meant to represent how the data of a
generic entity might change by the effect of nuisance factors, which are not descriptive of any par-
ticular entity. For instance, the image of a face might be modified by different lighting conditions,
facial expressions, occlusions, etc. It is assumed that all the changes inducible by nuisance factors
form a variation subspace $\mathcal{V}$, where the variation component $v$ is defined. Therefore, a data point
is modeled as
\[ x = s + v . \] (3.1)

If \( P_\mathcal{V} : \mathbb{R}^m \to \mathcal{V} \) is the projection operator mapping an \( m \)-dimensional vector onto \( \mathcal{V} \), \( x \) can be further decomposed as \( x = (P_\mathcal{V}s + v) + (s - P_\mathcal{V}s) \). In particular, the first component \( a = P_\mathcal{V}s + v \), is defined in \( \mathcal{V} \), whereas the second component \( b = s - P_\mathcal{V}s \), is defined in the orthogonal complement of the variation space, \( \mathcal{V}^\perp \).

The decomposition \( x = a + b \) has the following property. Let us assume that \( x_1 \) and \( x_2 \) are two different points representing the same entity. According to (3.1), it must be that \( x_1 = s + v_1 \) and \( x_2 = s + v_2 \), because they have been affected by different nuisance factors. This means that \( a_1 = P_\mathcal{V}s + v_1 \), and \( a_2 = P_\mathcal{V}s + v_2 \); however, \( b_1 = s - P_\mathcal{V}s = b_2 \), which highlights that the component \( b \) is invariant to the changes induced by the nuisance factors. We refer to the subspace where \( b \) is defined as the invariant subspace \( \mathcal{B} \), which will be a subspace of \( \mathcal{V}^\perp \). Fig 3.1 illustrates the details of our additive model to extract invariant subspace in a block diagram.

### 3.4 Recognition via the Invariant Subspace

We assume that a set of \( n \) training data samples from \( N \) different entities, or object classes (e.g. images of people faces, or people whole body appearances), are given, where each class \( i \) has \( n_i \) samples. Every sample \( x_j \) is modeled according to (3.1), and we concatenate the data into a matrix \( X = [X_1, X_2, \cdots, X_N] \in \mathbb{R}^{m \times n} \), where \( X_i \in \mathbb{R}^{m \times n_i} \) is the training data matrix obtained by lining up the samples for class \( i \).

Model (3.1) has been implicitly adopted by the most successful recent approaches to the face recognition problem. In particular, the SRC method [12] aims at “carefully” composing each of the \( X_i \)’s in such a way that the selected samples can represent the salient components \( s_i \)’s in the best possible way. The matching between a test point \( x = s + v \), and a salient component \( s_i \) (i.e. the classification) is based on sparse coding and residual computation and has demonstrated a remarkable robustness against the variation component \( v \), leading to high recognition rates. The SRC approach has been further improved against potential corruptions of the test data point. For instance, [115] improves upon occlusions and computational cost, [116] robustifies the sparse coding problem by computing a sparsity-constrained maximum likelihood solution, [117] simultaneously handles the misalignment, pose and illumination invariance, and [118] addresses the issue of reducing the large amount of training data needed by SRC to be effective.

To address the more general case where also the training data is highly affected by nuisance factors, and a “careful” composition of \( X \) is not possible, the SRC approach has been augmented in different ways. In [43] a low-rank matrix recovery [62] approach is designed for pre-processing
the corrupted training data. After this step, the SRC method can be applied more effectively. Another approach, [119], proposes to apply sparse coding for modeling the sufficient component by learning a dictionary of prototypes, each of which, given by the average of the data in $X_i$, is meant to approximate $s_i$. In addition, sparse coding is also used for modeling the variation subspace. The concatenation of the prototype and the variation dictionaries form a new dictionary with which the SRC method can be applied more effectively.

In this work we propose to address the recognition problem with highly corrupted training and testing data by exploiting model (3.1) in a very different way than previous work. The idea is based on a simple observation. Suppose that the projection operator $P_V$ was available. Then, a test sample $x$ could be processed by computing $x - P_V x = b$. Similarly, for the training dataset, following the property of the invariant subspace, computing $X - P_V X$ produces $[b_11_{n_1}^\top, b_21_{n_2}^\top, \ldots, b_N1_{n_N}^\top]$, where $b_i$ is the invariant of class $i$, and $1_{n_i}$ is a column vector of ones with length $n_i$. Therefore, recognition could be done by a simple matching between $b$ and the set of $b_i$'s. This means that corruption (or intra-class variability) in training and testing data, as well as recognition, could be handled in a very easy, and efficient way with simple geometry tools.

One major challenge of the proposed approach is posed by the case when two different sufficient components $s_1 \neq s_2$, are such that $s_1 - P_V s_1 = s_2 - P_V s_2$. This means it would be impossible to discriminate between the corresponding classes. The supervised learning approach introduced in the following sections will: (1) allow to learn the invariant subspace, and (2) inherently address the challenge just outlined by promoting a uniform inter-class separability.

### 3.5 Invariant Subspace Learning

We begin by observing that since every data point is modeled as $x_j = a_j + b_j$, the training data set $X$, can be decomposed by $X = A + B$, where $A \in \mathbb{R}^{m \times n}$ collects all the $a_j$'s, and $B \in \mathbb{R}^{m \times n}$ collects all the invariant components, $b_j$'s. We assume that the variation subspace $V$ has a finite dimension, which is lower than $\min\{m, n\}$. This is reasonable because it states that there are enough data for learning the variation subspace of interest, it allows avoiding overfitting, and it makes the problem tractable. Therefore, attempting to recover $A$, which in turn allows recovering $B$, entails solving a low-rank matrix recovery problem.

In practice, the training data will also be affected by noise. We admit that a small percentage of the entries of $X$ are corrupted by values not modeled by the variation and invariant components, which means that such noise should be sparse. This will account for data deviations unlikely to be captured by a finite dimensional linear subspace, such as those induced by image saturations, like image glare, or the presence of strong edges. Therefore, if $E \in \mathbb{R}^{m \times n}$ is the matrix of sparse noise,
the model for the training dataset is given by

\[ X = A + B + E . \] (3.2)

Before posing the optimization problem for the estimation of \( A \), and \( B \), we review the standard low-rank matrix recovery problem with sparse noise.

### 3.5.1 Low Rank Matrix Recovery

Low Rank (LR) matrix recovery seeks to decompose a data matrix \( X \) into \( A + E \), where \( A \) is a low-rank matrix and \( E \) is the associated sparse error. More precisely, given the input data matrix \( X \), LR minimizes the rank of the matrix \( A \) while reducing \( \| E \|_0 \) to derive the low-rank approximation of \( X \). Since the optimization mentioned above problem is NP-hard, [62] proposed to relax the original problem into the following tractable formulation

\[ \min_{A,E} \| A \|_* + \alpha \| E \|_1 \quad \text{s.t.} \quad X = A + E . \] (3.3)

In (3.3), the nuclear norm \( \| A \|_* \) (i.e. the sum of the singular values) approximates the rank of \( A \), and the \( \ell_0 \)-norm \( \| E \|_0 \) is replaced by the \( \ell_1 \)-norm \( \| E \|_1 \), which sums up the absolute values of the entries of \( E \). It is shown in [62] that solving the relaxed version of the problem (3.3) is equivalent to solving the original low-rank matrix approximation problem, as long as the rank of \( A \) to be recovered is not too large, and the number of errors in \( E \) is small (sparse). To solve the optimization problem (3.3) it is possible to apply the efficient method of augmented Lagrangian multipliers (ALM) [92].

In face recognition, \( X \) represents the gallery of images of \( N \) subjects. By performing the low-rank matrix recovery (3.3), \( X \) gets decomposed into \( A = [A_1, \cdots, A_N] \), and \( E = [E_1, \cdots, E_N] \). The desired effect is for a subject \( i \) to produce a low-rank matrix \( A_i \) with columns that look very much alike and span a very narrow space around the sufficient component \( s_i \) [43]. The corresponding sparse matrix \( E_i \) is expected to pick up the variation components, caused by nuisance factors (e.g., occlusions, disguise, lighting variations, pose, etc.). In [43] the low-rank matrices \( A_i \)'s are iteratively optimized with robust PCA [62]. Also, for an increased class separation, a structural incoherence prior is included in the optimization. Other approaches instead, increase discriminability by learning a dictionary, in combination with sparse coding and low-rank modeling. In particular, [103] learns a low-rank discriminative dictionary for every class to operate the sparse representation of data samples. [45] instead learns a discriminative dictionary for a sparse and low-rank representation. In [103] testing is similar to the SRC, in [45] the learning of an additional
linear multi-class classifier is required. Fig 3.2 illustrates how images from different subjects can be affected by nuisance factors, which lay in a low-rank space, in Fig 3.2 subjects with glass and scarf produce a low-rank nuisance subspace that increases the similarity of subjects with different identities.

Unlike previous work, we do not learn a dictionary, and the columns of the low-rank matrix $A$ are meant to span the variation space $\mathcal{V}$, not the space of the sufficient components. Discriminability comes from learning the invariant components $B$, which leads to a very simple and efficient rule for classification, and can promote class separation with a supervised learning approach described in the following section.

### 3.5.2 Supervised Learning

To learn model (3.2), standard LR (3.3) is insufficient because we also need to learn the invariant components $B$. To do so, we need to take into account the geometric, and invariance constraints of (3.2).

**Geometric constraint.** In particular, the invariant subspace should be included in the orthogonal complement of the variation subspace $\mathcal{V}^\perp$. Therefore, $A$ and $B$ should satisfy the relationship

$$B^T A = 0.$$  \hfill (3.4)

**Invariance constraint.** In addition, given two data points $x_1 = a_1 + b_1 + e_1$ and $x_2 = a_2 + b_2 + e_2$, if they are representative of the same class $i$, the invariant components should be the same, i.e. $b_1 = b_2$. To express this in an algebraic form, $b_1$ and $b_2$ should be the solution to the linear system given by the equations $b_1 = 1/2(b_1 + b_2)$, and $b_2 = 1/2(b_1 + b_2)$. For $n$ data points, where $B = [B_1, B_2, \cdots, B_N]$, the constraint on the invariant components would be $b_1 = b_2 = \cdots = b_n$, for $B_1$, $\cdots$, and $b_{n-n_N+1} = b_{n-n_N+2} = \cdots = b_n$, for $B_N$. This can still be expressed in an algebraic
form, by generalizing the system of two linear equations to the following expression

\[ B(I - Q) = 0, \]  

(3.5)

where \( I \) is the identity matrix, and \( Q \) is a block-diagonal matrix, given by following structure:

\[
Q = \begin{pmatrix}
  r_1 & \frac{1}{N_1} & \frac{1}{N_1} & 0 & \cdots & 0 \\
  r_2 & \frac{1}{N_1} & \frac{1}{N_1} & 0 & \cdots & 0 \\
  r_3 & 0 & 0 & \frac{1}{N_2} & \cdots & 0 \\
  \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\
  r_{N-1} & 0 & 0 & \cdots & \frac{1}{N_{n-1}} & \frac{1}{N_{n-1}} \\
  r_N & 0 & 0 & \cdots & \frac{1}{N_n} & \frac{1}{N_n}
\end{pmatrix}
\]

In order to learn \( A \) and \( B \), we propose to augment problem (3.3) with model (3.2), the geometric constraint (3.4), and the invariance constraint (3.5). In particular, to make the problem more tractable, the geometric and invariance constraints are relaxed to the penalty terms \( \| B^T A \|_F^2 \), and \( \| B(I - Q) \|_F^2 \) in the following optimization problem

\[
\min_{A,B,E} \| A \|_* + \alpha \| E \|_1 + \beta \| B(I - Q) \|_F^2 + \gamma \| B^T A \|_F^2 \\
\text{s.t. } X = A + B + E, 
\]

(3.6)

where \( \| \cdot \|_F \) indicates the Frobenius norm, and \( \alpha, \beta, \) and \( \gamma \) are penalty weights. Note that the addition of the invariance constraint (3.5) as a penalty, through \( Q \) injects the training dataset labeling information inside the learning problem, turning it into a supervised approach.
3.5.3 Optimization

In order to solve problem (3.6), we use the exact ALM method [92], and start by computing the augmented Lagrangian function $L(A, B, E, \lambda)$, given by

$$L = \|A\|_* + \alpha \|E\|_1 + \beta \|B(I - Q)\|_F^2 + \gamma \|B^\top A\|_F^2$$

$$+ \langle \lambda, X - A - B - E \rangle + \frac{\mu}{2} \|X - A - B - E\|_F^2$$

$$= \|A\|_* + \alpha \|E\|_1 + \beta \|B(I - Q)\|_F^2 + \gamma \|B^\top A\|_F^2$$

$$+ \frac{\mu}{2} \|X - A - B - E\|_F^2 - \frac{1}{2\mu} \|\lambda\|_F^2$$

$$= \|A\|_* + \alpha \|E\|_1 + \beta \|B(I - Q)\|_F^2 + h(A, B, E, \lambda, \mu)$$

$$- \frac{1}{2\mu} \|\lambda\|_F^2,$$

where $\langle X, Y \rangle \equiv \text{trace}(X^\top Y)$, $\mu$ is a positive scalar, $\lambda$ is a Lagrange multiplier matrix, and $h(A, B, E, \lambda, \mu) = \frac{\mu}{2} \|X - A - B - E\|_F^2 + \frac{\lambda}{\mu} \|2\|_F^2 + \gamma \|B^\top A\|_F^2$ is a quadratic convenience function. We optimize (3.7) with an alternating direction strategy, and at every outer iteration of Algorithm 2, $A$, $B$, and $E$ are first iteratively updated until convergence; subsequently, $\lambda$ and $\mu$ are updated. The inner iteration updates of Algorithm 2 are given below.

Updating $A_{k+1}$: From the reduced augmented Lagrangian it is convenient to use the linearization technique of the LADMAP method [120], very effectively used also by other approaches [45, 103,

![Identity Subspace Learning](image)

Figure 3.3: Identity Subspace Learning. This Figure illustrates how images corresponding to a subject is mapped to identity subspace.
Algorithm 1 Invariant Components Learning via the Exact ALM Method

Require: Observation matrix $X$, labels $Q$, and penalty weights $\alpha, \beta, \gamma$

1: $k = 0$; $\rho > 1$; $\mu_0 > 0$; $\eta = \|X\|_F^2$; $\lambda_0 = \max(\|\text{sgn}(X)\|_F, \alpha^{-1} \|\text{sgn}(X)\|_\infty)$; $A_0 = 0$; $B_0 = XQ$; $E_0 = 0$
2: while not converged do
3:     $j = 0$; $A_k^0 = A_k$; $B_k^0 = B_k$; $E_k^0 = E_k$
4:     while not converged do
5:         $(U, \Sigma, V) = \text{svd}(X - B_k^j - E_k^j + \mu - 1 \lambda_k - \gamma B_k^j B_k^j\top A_k^j)$; $A_{j+1}^k = U \Sigma_s(\eta \mu_k)^{-1}(\Sigma) V^\top$
6:         $E_{j+1}^k = S_{\alpha \mu_k}^{-1}(X - A_{j+1}^k - B_k^j + \mu^{-1} \lambda_k)$
7:         Update $B_{j+1}^k$ by solving (3.11) with $A_{j+1}^k$ and $E_{j+1}^k$
8:         $j \leftarrow j + 1$
9:     end while
10:    $A_{k+1} = A_{j+1}^k$; $B_{k+1} = B_{j+1}^k$; $E_{k+1} = E_{j+1}^k$
11:   $\mu_{k+1} = \rho \mu_k$; $\lambda_{k+1} = \lambda_k + \mu_k (X - A_{k+1} - B_{k+1} + E_{k+1})$
12:  $k \leftarrow k + 1$
13: end while

Ensure: $A_k, B_k, E_k$

121], and replace the quadratic term $h$ with its first order approximation, computed at iteration $k$, and add a proximal term, giving the following update

$$A_{k+1} = \arg \min_A \|A\|_s + \eta \nabla_A h(A_k, B_k, E_k, \lambda_k, \mu_k),$$

$$= \arg \min_A \|A\|_s + \frac{\eta \mu_k}{2} \|A - A_k\|_F^2$$

$$- \gamma B_k B_k\top A_k\|_F^2,$$  \hspace{1cm} (3.8)

where $\eta$ must be greater than $\|A\|_F^2$ [120]. The solution to (3.8) is reported in Algorithm 2, and is obtained by applying the singular value thresholding algorithm [122], with the soft-thresholding shrinkage operator $S_\epsilon(x)$, which is equal to: $x - \epsilon$ if $x > \epsilon$, $x + \epsilon$ if $x < -\epsilon$, and 0 elsewhere.

Updating $E_{k+1}$: From (3.7), the augmented Lagrangian reduces to

$$E_{k+1} = \arg \min_E \alpha \|E\|_1 + \frac{\mu_k}{2} \|E - (X - A_{k+1} - B_k + \lambda_k)\|_F^2$$  \hspace{1cm} (3.9)

and the solution, reported in Algorithm 2, is still obtained with an instance of the singular value thresholding algorithm [122].

Updating $B_{k+1}$: This update is computed as
Figure 3.4: Synthetic data. (a) Decomposition of 12 synthetic data points. (b) Decomposition of the same 12 points with Algorithm 2. Top row: input points $X$. Second row: $A$ components. Third row: Sparse errors $E$. Bottom row: Invariant components $B$. Columns with invariant components depicting the same digit belong to the same class. The digits appear "hazy" as a result of being orthogonal to the $A$ components by construction.

\[
B_{k+1} = \arg\min_B \frac{\mu_k}{2} \|X - A_{k+1} - E_{k+1} - B + \frac{\lambda_k}{\mu_k} \|^2_F + \\
\beta \|B(I - Q)\|^2_F + \gamma \|B^T A_{k+1}\|^2_F.
\]  

(3.10)

Note that the cost function in (3.10) is quadratic in $B$. Therefore, the update can be obtained by computing the partial derivative with respect to $B$ of the cost function, and then set it to zero. This leads to a Sylvester equation in $B$, given by

\[
\gamma A_{k+1} A_{k+1}^T + B \left( (\beta + \frac{\mu_k}{2}) I - 2\beta Q + \beta QQ^T \right) = \\
\frac{\mu_k}{2} \left( X - A_{k+1} - E_{k+1} + \frac{\lambda_k}{\mu_k} \right).
\]  

(3.11)

Therefore, the update (3.10) can be computed with a standard Sylvester equation solver. The full optimization procedure is summarized in Algorithm 2.
3.6 Classification

Given a test data point $x$, even if, strictly speaking, we are not in an instance-based learning setting, the obvious approach to perform classification is to compute a label $y$ with a nearest-neighbor (NN) approach, where $y = \arg \min_i d(x, B_i)$, and $d(\cdot, \cdot)$ is a suitable distance between $x$ and the invariant matrix $B_i$, representing class $i$.

Following the strategy outlined in Section 3.4, from the invariant components $B_i$ one can estimate $P_{B_i} : \mathbb{R}^m \to B_i$, the operator that projects data points directly onto $B_i \subset B$, the invariant subspace for class $i$. Doing so has the advantage that the projection of $x$ onto $V^\perp$ gives $b + P_{V^\perp}e$, whereas the projection of $x$ onto $B_i$ gives $b + P_{B_i}e$, and since $B_i \subset V^\perp$, it follows that $\|P_{B_i}e\|_F \leq \|P_{V^\perp}e\|_F$, which means a lower noise corruption. Therefore, we propose to use the following Frobenius norm $d_F(x, B_i) = \|b_i - P_{B_i}x\|_F$. Note that if $B_i$ can be approximated with $b_i1_n^\top$, as it normally should, then the distance computation is even faster, because given by

$$d_F(x, B_i) = \|b_i - P_{B_i}x\|_F. \quad (3.12)$$

3.6.1 Local Metric Learning

The approach outlined above, which has been derived using geometry, is amenable to an interpretation from a metric learning perspective. Let us recall the definition of Mahalanobis distance between two points $x_i$ and $x_j$, given by $d_M(x_i, x_j) = \sqrt{(x_i - x_j)^\top M (x_i - x_j)}$, where $M$ is a symmetric positive semi-definite matrix. A global linear metric learning method learns a matrix $M$ according to a specific criterion. Since the decomposition $M = L^\top L$ is always possible, the Mahalanobis distance can be expressed also as $d_M(x_i, x_j) = \|L(x_i - x_j)\|_F$.

Metric learning improves the performance of the NN classifier if used instead of the Euclidean metric. It has been applied effectively for classification [123], retrieval [124], person reidentification [125], and widely for face verification [86, 126, 127, 128, 129]. Different aspects of metric learning have been investigated, like distance parameters selection, scalability, whether training data should be used in pairs [86], triplets [84] or quadruplets [130], or whether data undergoes a linear [85, 131], or nonlinear [87, 129, 132, 133] transformation. Global metric learning methods learn the importance and correlation of different input features, and take them into account for NN classification, regardless of the specific feature neighborhood where they are applied. Since the discriminative power of input features might vary between different neighbors, learning a global metric may be suboptimal. This has motivated the development of local metric learning approaches [88, 89, 90, 91, 134], which increase the discriminative power of global Mahalanobis metric learning by learning a number of local metrics.
The proposed approach can be seen as a local metric learning approach, where for the neighborhood of each of the invariant components we learn a Mahalanobis metric. In particular, if $B_i = U_{B_i}S_{B_i}V_{B_i}^T$ is the singular value decomposition (SVD) of $B_i$, then the distance (3.12) can be rewritten as $d_F(x, B_i) = \|U_{B_i}U_{B_i}^T(x - b_i)\|_F$. This means that $d_F(x, B_i) = d_M(x, b_i)$, i.e., the Mahalanobis distance between $x$ and $b_i$, with respect to $M_i = U_{B_i}U_{B_i}^T$. Therefore, learning a representation based on the invariant components $B$, is equivalent to learning a set of cluster centers $\{b_i\}$, and a set of Mahalanobis matrices $\{M_i\}$ that act on the neighborhood of each center, and with which labels are assigned based on the NN rule $y = \arg \min_i d_{M_i}(x, b_i)$.

### 3.6.2 Class Separation

Most of the local approaches learn the metrics for each neighborhood independently [91], and require the addition of a form of regularization to avoid overfitting. In contrast, related to [135], our approach learns the metrics jointly, according to the constraints (3.4) and (3.5). While the first eliminates the effects of nuisance factors, the second ensures not only invariance but also class separation. More specifically, since the invariance constraint (3.5) can be re-written as $Q = B^T(BB^T)^+B$, it is easy to realize that the Mahalanobis distance $d_M(b_i, b_j)$, with $M = (BB^T)^+/n$, between the invariant components $b_i$ and $b_j$, for classes $i$ and $j$, is such that

$$d_M(b_i, b_j) = \begin{cases} 0 & \text{if } i = j, \\ \sqrt{2N} & \text{otherwise}, \end{cases}$$

(3.13)

where for simplicity it is assumed $n_i = n_j$. Without loss of generality, if we assume that the columns of $B$ are zero mean, $M$ is the inverse of the covariance of $B$ (for a short discussion there is no need to address the rank deficiency of $B$, and the use of the pseudoinverse $(BB^T)^+$). Therefore, (3.13) means that the invariant subspace $B$ is such that two different sufficient components $s_i$ and $s_j$ originate two invariant components $b_i$ and $b_j$ that are different (i.e., $b_i = s_i - P_B s_i \neq s_j - P_B s_j = b_j$), and equidistant (i.e., $d_M(b_i, b_j) = \sqrt{2N} \forall i \neq j$), thus promoting a uniform class separation.

The observation above suggests also the use of a global Mahalanobis metric for NN classification, e.g., in the form of $d_M^2(x, B_i) = n_i^{-1} \sum_{b \in B_i} d_M^2(x, b)$. However, it is more efficient to use the corresponding similarity measure $\kappa_M(b_i, b_j) = b_i^T(BB^T)^+b_j$, which gives 0 if $i \neq j$, and $1/n_i$ if $i = j$. Therefore, we propose the global Mahalanobis similarity measure defined as $\kappa_M(x, B_i) = 1_{n_i}B_i^T(BB^T)^+x$, and the label assignment is done according to $y = \arg \max_i \kappa(x, B_i)$. If $B_i = b_i1_{n_i}^T$, the similarity reduces to

$$\kappa_M(x, B_i) = n_i b_i^T(BB^T)^+x.$$  

(3.14)
3.7 Experiments

To examine the proposed method in a practical setting, we have done meticulous experiments using synthetic data and in two challenging real-world application of face recognition and person re-identification. For the face recognition performance, we have tested the algorithm on three publicly available datasets of AR, AT&T and Extended Yale face datasets, also in order to illustrate the generalization property of the proposed method. We have provided re-identification results on I-LIDS and CAVIAR4REID datasets which again are publicly available for person re-identification, while trying the proposed method on different data we use grid searching to find best parameters for different data, but it is noteworthy that values were toned mainly around $\beta = 100, \gamma = .5$ for most of the cases.

3.7.1 Synthetic Data

To empirically verify the convergence of Algorithm 2, we have created a synthetic dataset made of $n = 120$ images of $32 \times 28$ pixels, with $N = 10$ invariant components depicting digits, and with image patterns representing $A$. The synthetic $A$ and $B$ satisfy the constraints (3.4), and (3.5), and we have added sparse noise $E$, corrupting 20% of randomly selected pixels, with values drawn from a uniform distribution between 0 and the largest possible pixel value in the image. Figure 3.4(a) shows the decomposition in $A$, $E$, and $B$ of 12 synthetic data points, $X$ (top row), and Figure 3.4(b) shows the estimated decomposition of the same points. Visually, the recovered decomposition closely resembles the originals, and the coefficients of variation (i.e., $\|\hat{z} - z\|_F/\|z\|_F$ where $\hat{z}$ is the estimated quantity), are 9.71%, 8.67%, 37.2%, for $A$, $B$, and $E$, respectively.

3.7.2 AR Dataset

![AR dataset](image1)

a) AR dataset

![AT&T dataset](image2)

b) AT&T dataset

Figure 3.5: a) AR face dataset. 26 cropped and aligned images taken under varying controlled lighting conditions in two different sessions b) AT&T face dataset. 10 face images captured under various poses taken from 10 subjects
In this section we present results obtained from the proposed method on AR face dataset, AR dataset contains over 4,000 frontal images of 126 people’s faces (70 men and 56 women). Images are taken in two sessions and under different facial expressions, illumination conditions, and occlusions (such as sunglass and scarf). Fig. 3.5(a) illustrates 13 images taken from one subject in one session, in second session images are taken in the exactly similar way as they are taken in the first session. Three of images are occluded by sunglass and another three are occluded by the scarf and taken under different laboratory-controlled lightings. The cropped and normalized face images are of size $165 \times 120 = 19,800$ pixels and converted into grayscale, we downsample images in order of $4 \times 4$, Also in this experiment we select a subset of 50 men and 50 women (as [12, 43]).

Fig. 3.6 illustrates the unconstrained decomposition of face images from AR dataset, four rows of these figures represent $D = X, A, E, Y$ correspondingly from top to bottom, as it is evident the proposed algorithm can extract similar bases from same subjects, and sparse noise can capture the irregularities peculiar to each subject, and matrix $A$ captures the low-rank information across all subjects. In face recognition literature most of the works do the training using uncluttered images and they measure the robustness of the algorithm when a cluttered image is introduced. Here we will show that the proposed algorithm is robust to clutter even during training and it rules out the nuisance information (i.e. clutter) correctly, where it accumulates them in the low-rank matrix $A$.

Therefore, we are intrigued to test the algorithm using cluttered images in training; consequently, we adopt scenarios similar to [43, 45] with three following scenarios.

**Sunglass:** In this experiment, we used occluded images by sunglass, in training process. We randomly, select one image with sunglass and seven neutral images from session 1 for training and the remaining neutral images and images with sunglass for testing (seven neutral and three sunglass images from session 2 and two sunglass images from session 1). Therefore we will have a total of 8 training images and 12 testing images per person, and it is noteworthy that images with sunglass are occluded about 20%.

**Scarf:** Second scenario is pretty much similar to the first one with a slight difference that occluded images are people with a scarf. Images taken from people with scarf on, are more dire cases where about 40% of face images are occluded, here. Again we pick an image with scarf randomly from the first session along with remaining neutral images from session one, overall constructing our training set and remaining neutral images and images with scarf from both sessions are used as testing set.

**Sunglass+Scarf:** In this last scenario we will include occluded images both with sunglass and scarf in training, we randomly pick one image from sunglass and one image from scarf and seven neutral images from session 1, and we use seventeen remaining images from session 1 and 2 for
Figure 3.6: Decomposition on AR face attributes In here, we illustrate decomposition results using our objective function on three different identities (which mostly have attribute or lighting variation), first row is the original downsampled image $X$, second row is the low rank matrix $A$, third row is the sparse noise $E$, and last row is the identity subspace $Y$. It is noteworthy that, all images are rescaled to $[0, 255]$ for better visualization of decomposition.
Figure 3.7: AR dataset. Recognition rates versus different numbers $p$, of corrupted training images per class for the three scenarios: sunglasses (left), scarf (center), sunglasses and scarf (right).

<table>
<thead>
<tr>
<th>dimension</th>
<th>$p=1$</th>
<th>$p=2$</th>
<th>$p=3$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>sunglass</td>
<td>scarf</td>
<td>mixed</td>
</tr>
<tr>
<td>Our Method</td>
<td>87.3±0.30</td>
<td>83.9±0.52</td>
<td>84.6±0.37</td>
</tr>
<tr>
<td>SLR</td>
<td>87.1±0.64</td>
<td>83.0±0.57</td>
<td>81.8±0.70</td>
</tr>
<tr>
<td>LR w. Incoh.</td>
<td>86.8±0.40</td>
<td>82.9±0.34</td>
<td>79.1±0.56</td>
</tr>
<tr>
<td>SRC</td>
<td>84.9±0.23</td>
<td>76.2±0.42</td>
<td>79.2±0.42</td>
</tr>
</tbody>
</table>

Table 3.1: Recognition accuracy comparison on AR dataset. Comparison between our method and SLR [45], low-rank with incoherence [43], and SRC [12] on the AR face dataset using the same protocol described in Section 6 of this chapter.

To illustrate the robustness of the proposed method in excluding the nuisance data like sunglasses and scarves, we set up above experiments with more occluded images in training. For instance, for the case of sunglasses, instead of including only one image with sunglasses in training, we will include $p = 2, 3$ images with sunglasses in training and use the remaining $3 - p$ images along with images from session 2 for testing. We repeat the same experiment for the cases of scarf and scarf+sunglass where basically for the case of scarf+sunglass and $P = 3$ we will have 6 cluttered images in training set.

3.7.3 AT&T Dataset

AT&T dataset is maintained at AT&T laboratories in Cambridge University. The dataset includes face images of 40 subjects taken in 10 controlled variations, which involves facial gestures (i.e. smiling and non-smiling) neutral or with glasses, also face rotational in four directions with no
<table>
<thead>
<tr>
<th>approach</th>
<th>Recognition Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Our Method</td>
<td>95%</td>
</tr>
<tr>
<td>ICA</td>
<td>85%</td>
</tr>
<tr>
<td>2DPCA</td>
<td>96%</td>
</tr>
<tr>
<td>SRC</td>
<td>93%</td>
</tr>
<tr>
<td>SLR</td>
<td>93.5%</td>
</tr>
</tbody>
</table>

Table 3.2: Recognition Rate on AT&T face dataset

Figure 3.8: a) Extended Yale Dataset. Up to 64 images in face space, cropped and aligned in different controlled lighting condition, this dataset involves 38 subjects.

more than 20 degrees. Similar to protocol in [136] we pick the first five images for training and last five images of each individual for testing. Tab. 3.7.3 illustrate the results for AT&T dataset. As it is illustrated in table its performance is on par with well-known methods.

### 3.7.4 Extended-Yale Dataset

The Extended Yale B dataset [137] contains tightly cropped face images of 38 subjects each subject has around 59 to 64 images taken under varying lighting conditions (see Fig. 3.8), which in total adds up to 2,414 images. The cropped images are of the size $192 \times 168 = 32,256$ pixels.

Fig. 3.17 visualize the outcome of the proposed method on the Yale dataset. As it is evident from experiments outcome, the proposed algorithm is able to capture the common identity features across various images of the same identity. One should also note that the images depicted in last row of the Fig. 3.17 are in fact reshaped vectors which lay in orthogonal space of the low rank subspace of the reshaped vector in second row.

Fig. 3.10 illustrate the numerical precision comparison on the recognition rate achieved from the proposed method along with the recognition rate obtained from SLR [45] and LR with incoherence [43] and SRC [12].

### 3.7.5 I-LIDS Dataset

i-LIDS MCTS dataset was captured indoor in a busy airport arrival hall. This dataset contains 476 person images of 119 people captured by multiple non-overlapping cameras. There are four images on average per person. We excluded the person who had only 1 or 2 images for our experiments.
All the images are normalized to $128 \times 64$ pixels, Fig 3.11 illustrates sample images from I-LID dataset. This dataset is challenging as it is a real world scenario and it contains a lot of illumination changes, occlusions and view angle change. In order to show generalization property of the proposed method, we try the proposed method on this dataset. Fig. 3.12 illustrate decomposition results obtained from the proposed method on raw pixels values in 3 channel.

Usually in person re-identification literature, people divide the dataset in 3 sets of training, gallery and probe set, then a model is learned on training set and is used to match images form probe set to test set. Here, simply we use only gallery and probe set for matching and instead of the sophisticated person re-identification features, we use only RGB raw values. Fig. 3.13 illustrates the comparison of CMC curves for our method and two mostly used algorithms in re-identification applications. As we see even though, this comparison is not done in fair conditions and all odds are against our method, still it produce comparable results to well known methods. For doing comparison with other methods we have modified their experiments for fair comparison. To compare

Figure 3.9: Decomposition on yale Illustrates decomposition results using our objective function on five different subjects, first row is the original downsampled image(D), second row is the low rank matrix(A), third row is the sparse noise(E), and last row is the identity subspace(B). It is noteworthy that, all images are rescaled to $[0 255]$ in all channel colors for better visualization of decomposition.
Figure 3.10: **Extended Yale B dataset.** Top from left to right: Recognition scores at different image downsampling rates for 8 and 32 training samples per subject; recognition rates obtained with the global metric (3.14) (Scheme 1) and the local metric (3.12) (Scheme 2) at various image resolutions and 32 training samples; Bottom running time in seconds of our Matlab implementations for training and testing.

![Extended Yale B dataset](image)

our results with RDC we have used 3 images in the gallery set and 1 image in the probe set. We run the experiment four times to make sure every image is part of the probe set and get our final results. For SDALF we learned the signature from 3 images in the gallery set for fair comparison.

For evaluation, we use the average cumulative match characteristic (CMC) curves to show the ranked matching rates. A rank r matching rate indicates the percentage of the probe images with correct matches found in the top r ranks against the p gallery images. Rank 1 matching rate denotes the correct matching/recognition rate. Although a high rank 1 matching rate is important, the top r ranked matching rate with a small r value is no less important as the top matched images will normally be verified by a human operator in most scenarios.
Figure 3.12: **Decomposition on i-LIDS person re-identification** Illustrates decomposition results using our objective function on five different subjects (mostly distinguished by clothing type and color), first row is the original downsampled image(X), second row is the low rank matrix(A), third row is the sparse noise(E), and last row is the identity subspace(Y), It is noteworthy that, all images are rescaled to [0 255] in all channel colors for better visualization of decomposition.

Figure 3.13: **i-LIDS MCTS dataset.** Cumulative matching curves obtained with 30 and 50 and 80 individuals. The plots refer to our method, the SDALF [105], the RDC [138], and to the matching done, as in [138], with a color histogram and the \( \ell_1 \)-norm (L1), and the Bhattacharyya distance (Bhatt).

### 3.7.6 CAVIAR4REID

CAVIAR4REID is a new dataset for evaluating person re-identification algorithms. As the name suggests, the dataset has been extracted from the CAVIAR dataset mostly famous for person tracking and detection evaluations. It is a challenging dataset because it has broad changes in resolution and it is extracted from a real scenario where re-identification is necessary due to the presence of multiple cameras and the pose variations between the images are severe. For this dataset we carried our experiments in a similar fashion like i-LIDS dataset.

About the computational burden of the proposed method, since, low-rank decomposition steps are very well discussed in [12], we won’t elaborate on it. However, the only added stall is the computation of Sylvester equation. Then time spent on training is akin to the low-rank method. However since our learned bases are in an orthogonal subspace of other nuisance elements, our
testing is extremely fast and in a Matlab implementation on a 2.40GHZ (2 processors) machine can be done less the half a second depending on the dimension of the feature vector.

Similarly to Figure 3.6, Figure 3.10 shows the decomposition into model (3.2) for two people and 8 training data points each, with the invariant components at the bottom.

Figure 3.10, right, also shows a comparison between the local metric approach (Scheme 2), based on (3.12), and the global metric approach (Scheme 1), based on (3.14), on a subset of the dataset, with 32 training data points per person, against different image resolutions. As expected, the local metric learning approach, because it adapts to the invariant component where it operates on, is able to provide better performance. From a geometric perspective, as highlighted in Section 3.6, the performance drop is justified by the fact that the global approach is not able to filter out as much noise as the local approach is capable of.

Figure 3.10, far right, shows a running time comparison between the Matlab implementations of ours, the SLR, and the LRwIn methods, running on a high-end PC. Our training procedure appears slightly more costly than the others, but, as anticipated, testing appears faster than SLR by a factor of 10, and faster than LRwIn by a factor of 25.
3.7.7 Labeled Face in Wild

We have tested the large margin nearest neighbor (LMNN) metric learning approach [140], and ours on the Labeled Faces in the Wild (LFW) dataset [139]. Fig. 3.15 illustrates the original dataset sample of the (LFW). Out of the 13,233 face images of 5749 unique individuals, we selected those with at least 10 images, for a total of 143 people and 4174 face images, which were aligned using deep funneling [141], tightly cropped to include only face information, and resized to $106 \times 96$ pixels (see Fig. 3.16). For each subject, we randomly selected seven images for training, and the rest were used for testing. The penalty parameters were $\alpha = 0.5$, $\beta = 1000$, $\gamma = 0.2$. The actual processing for both algorithms was repeated ten times and was done with the cropped images down-sampled by a factor of 4. In such a scenario with a highly non-linear variation space, we obtained comparable recognition accuracies, given by $46.9\% \pm 0.1$ for LMNN, and $47.1\% \pm 0.15$ for our method, whereas the baseline Euclidean distance provided an accuracy of $15.4\% \pm 0.05$. We also run LMNN and our method on the AR dataset on the highly corrupted scenario given by SUNGLASSES+SCARF with $p = 2$. The accuracy rate for LMNN was $62.8 \pm 0.1$, for our method was $71.2\% \pm 0.52$, and for the Euclidean distance was $29.3\% \pm 0.05$, which shows that our method performs better especially when robustness against corrupted samples in the gallery is needed.

3.8 Model Variations:

In this section we will study variations of the proposed model in (3.6); As It is mentioned in previous section primary objective of the optimization problem (3.6) was to learn a unique subspace for each one of subjects in training set, and to accomplish this, four identity equations were introduced. In the following, we will scrutinize through all these constraints that are enforced while learning to get these unique factor spaces. Basically, in this section, we will introduce more general formulation of the problem in hand, and then investigate the behavior of the optimization problem under different assumptions and parameters. The first constraint ($X = A + B + E$) is the most important and crucial equation that we enforce.
on data decomposition, it guarantees convergence of accumulation of decomposed elements to be same as the original data $X$. Next $B^\top A = 0$ enforces the orthogonality of learned unique subspaces ($Y$) to *nuisance conjoint space* ($A$). It should be noticed that this is also another important constraint which ensures $A$ lie in an orthogonal space of allocated subspaces to each unique subject. $B = BQ$ enforces closer Euclidean distance for bases from each particular subject, this term can be interpreted as *intra-class gaussian noise* model.

Combination of all these identities in our objective function can be formulated as:

Here we will study variations of the proposed model and will consider a broad spectrum of assumptions. In the end, we will show more results of the main model on some other datasets. Fig. 3.17 illustrates more results obtained through implementation of the main model on Extended Yale dataset.

### 3.8.1 Uncorrupted Data, $\alpha = \infty$

Setting $\alpha = \infty$ means ruling out $E$ from the optimization problem. This corresponds to a decomposition of $X$ with clear data, without any noise, giving the problem:

\[
< A, B >= \arg \min_{A,B} \|A\|_* + \beta \|B(I - Q)\|_F^2 + \gamma \|B^\top A\|_F^2 \quad \text{s.t.} \quad X = A + B
\]

(3.15)

This problem can be solved exactly as we solved (3.6) and the only difference is that we set $E = 0$. Our update scheme reduces down to updating $A$ and $B$. Fig. 3.18 illustrates the decomposition results obtained from solving problem (3.15).

### 3.8.2 Decomposition without Geometric Constraint, $\gamma = 0$

Consider $\gamma = 0$ in the main model equation [(6) in the main paper], which means removing the orthogonality constraint on $B$. This boils down to extracting the average value of each class, along
with other sparse and low-rank components:

\[
<A, B, E> = \arg \min_{A,B,E} \|A\|_* + \alpha\|E\|_0 \\
+ \beta\|B(I - Q)\|_F^2 \quad \text{s.t.} \quad X = A + B + E
\] (3.16)

Fig. 3.19 illustrates decomposition results obtained by solving problem (3.16). As we can see, due to the elimination of the orthogonality constraint, \(B\) looks like the average image of each subject, where shadows and saturation effects have been removed.

![Figure 3.17: Decomposition on yale](image)

Illustrates decomposition results using our objective function on five different subjects, first row is the original downsampled image(D), second row is the low rank matrix(A), third row is the sparse noise(E), and last row is the identity subspace(B). It is noteworthy that, all images are rescaled to [0 255] in all channel colors for better visualization of decomposition.
3.8.3 Decomposition with Exact Orthogonality Constraint, $\gamma = \infty$

In this case, we will impose the constraint $B^\top A = 0$, and the optimization problem can be reformulated as:

$$< A, B, E > = \arg\min_{A,B,E} \|A\|_* + \alpha\|E\|_0$$
$$+ \beta\|B(I - Q)\|_F^2 \quad \text{s.t.} \quad X = A + B, \quad B^\top A = 0$$

This is a problem with two constraints, and finding the optimal solution is more involved than in [(6) in the main paper]. To meet both constraints we leverage two augmented Lagrangian multipli-

Figure 3.18: **Uncorrupted data $\alpha = \infty$ decomposition on Extended-Yale Face dataset** This figure illustrates decomposition results using objective function in 3.15 on six different subjects, first row is the original downsampled image($X$), second row is the low rank matrix ($A$), third row is the sparse noise ($E$), and last row is the identity subspace($B$). It is noteworthy that, all images are rescaled to [0 255] in all channel colors for better visualization of decomposition.
\[< A, B, E > = \arg \min_{A,B,E} \|A\|_* + \alpha \|E\|_1 + (\mu/2)\|B^\top A\|^2_F \\
+ \beta \|B(I - Q)\|^2_F + < \lambda_1, X - A - B - E > \\
+ (\mu/2)\|X - A - B - E\|^2_F + < \lambda_2, B^\top A > \]

\[= \arg \min_{A,B,E} \|A\|_* + \alpha \|E\|_1 + \beta \|B(I - Q)\|^2_F + \]

\[q(X, A, B, E, \lambda_1, \lambda_2) - \mu^{-\frac{1}{2}}(\|\lambda_1\|^2_F + \|\lambda_2\|^2_F) \]

where:

\[q(X, A, B, E, \lambda_1, \lambda_2) = (\mu/2)\|B^\top A + \mu^{-\frac{1}{2}}\lambda_2\|^2_F \\
+ (\|X - A - B - E + \mu^{-\frac{1}{2}}\lambda_1\|^2_F).
\]

Figure 3.19: **Decomposition without Orthogonality Constraint \(\gamma = 0\) on Extended-Yale Face dataset** Illustrates decomposition results using objective function on 3.16 six different subjects, first row is the original downsampled image(X), second row is the low rank matrix(A), third row is the sparse noise(E), and last row is the identity subspace(Y). It is noteworthy that, all images are rescaled to [0 255] in all channel colors for better visualization of decomposition.
Where \( \mu \) is the regularizer for similarity and error terms. Eq. (3.18) can be solved using an iterative minimization method, similar to the one in the paper. In the following we derive the updating scheme for the parameters \( A, E \) and \( B \):

\[
A_{k+1} = \arg \min_A \| A \|_* + < \nabla_A q(A_k, E_k, B_k, \lambda_k; \mu_k),
\]

\[
A - A_k > + (\eta \mu_k/2) \| A - A_k \|_F^2
\]

\[
A_{k+1} = \arg \min_A \| A \|_* + (\eta \mu_k/2) \| A - A_k + [- (X - A_k
\]

\[
- E_k - B_k + \mu^{-1} \lambda_{1,k}/\mu_k] + B_k (B_k^\top A_k + \mu^{-1} \lambda_{2,k}))/\eta \|_F^2
\]

\[
= S_{(\eta \mu_k)^{-1}} (X - B - E + \mu^{-1} \lambda_{1,k} - B_k (B_k^\top A_k
\]

\[
+ \mu^{-1} \lambda_{2,k})
\]

(3.19)

Figure 3.20: **Decomposition with Exact Orthogonality Constraint** \( \gamma = \infty \) on Extended-Yale Face dataset Illustrates decomposition results using our objective function 3.17 on six different subjects, first row is the original downsampled image(X), second row is the low rank matrix(A), third row is the sparse noise(E), and last row is the identity subspace(B). It is noteworthy that, all images are rescaled to [0 255] in all channel colors for better visualization of decomposition.
Where $\nabla_A q$ is the gradient of $q$ with respect to $A$.

**Updating $E_{k+1}$**:

\[
E_{k+1} = \arg\min_E \alpha \|E\|_1 + \langle X - A_{k+1} - E - B_k, \lambda_{1,k} \rangle \\
+ (\mu_k/2)\|X - A_{k+1} - E - B_k\| \\
= (\alpha / \mu_k)\|E\|_1 + (1/2)\|E - X + A_{k+1} + B_k - \lambda_{1,k}/\mu_k\| \\
= \mathcal{S}(\alpha \mu_k^{-1})(X - A_{k+1} - B_k + \lambda_{1,k}/\mu_k) \\
\tag{3.20}
\]

**Updating $B_{k+1}$**:

\[
B_{k+1} = \arg\min_B (\mu_k/2)(\|X - A_{k+1} - E_{k+1} - B_k\| \\
+ \lambda_1/\mu_k)^2 + \beta \|B(I - Q)\|_F^2 + (\mu_k/2)\|B^\top A + \lambda_2/\mu_k\|_F^2) \\
\tag{3.21}
\]

Since Eq. (3.21) is a quadratic function, in order to find the optimum solution we set its gradient to zero, therefore we have:

\[
\frac{\partial L(B)}{\partial (B)} = -\mu_k(X - A - B - E + \lambda_1/\mu) \\
+ 2\beta B(I - 2Q - QQ^\top) + \mu_k A(A^\top B + \lambda_2/\mu_k) = 0 \\
\mu A A^\top B + 2\beta B((1 + \mu_k/2\beta)I - 2Q - QQ^\top) = \\
\mu_k(X - A_{k+1} - E_{k+1} + \lambda_1/\mu - A\lambda_2/\mu_k) \\
\tag{3.22}
\]

which is the famous Sylvester equation Fig. 3.20 shows the results obtained with this decomposition.

### 3.8.4 Unsupervised Decomposition, $\beta = 0$

$\beta = 0$ corresponds to removing the invariance constraint. This also means removing the labeling information, giving the unsupervised problem:

\[
< A, B, E > = \arg\min_{A,B,E} \|A\|_* + \alpha \|E\|_1 + \gamma \|B^\top A\|_F^2 \\
\text{s.t. } X = A + B + E \\
\tag{3.23}
\]

This problem has a solution similar to [(6) in the main paper] and will be solved using ALM, however, the step used for updating $B$ doesn’t require solving a Sylvester equation. For updating $A$ and $E$ we can follow the same steps described in [(6) in the main paper]. For updating $B$ we
Figure 3.21: **Unsupervised decomposition on Extended-Yale** $\beta = 0$ Illustrates decomposition results using Eq.(3.23) our objective function on six different subjects, first row is the original downsampled image(X), second row is the low rank matrix(A), third row is the sparse noise(E), and last row is the identity subspace(B). It is noteworthy that, all images are rescaled to $[0, 255]$ in all channel colors for better visualization of decomposition.
Figure 3.22: **Visual results of parameter selection**, $\alpha = .5, \gamma = 0$. This images illustrates visualization results using our main model (Eq. 3.6) using parameters alpha=.5 gamma=0 and obtained rank of Rank=7

Proceed as follows:

$$\frac{\partial L(B)}{\partial (B)} = 2\mu_k (X - A - B - E + \lambda/\mu) - 2\gamma AA^\top B = 0$$

$$2(\gamma AA^\top + \mu_k)B = \mu_k (X - A_{k+1} - E_{k+1} + \lambda/\mu)$$

Which gives a simple linear equation with exact solution given by:

$$B = \frac{1}{2}(\gamma AA^\top + \mu_k)^{-1}\mu_k (X - A_{k+1} - E_{k+1} + \lambda/\mu_k)$$

Fig. 3.21 illustrates the decomposition results using Eq. (3.23). It should be noted that the absence of the invariant constraint causes the invariant components of a given subject to be significantly different.

Fig. 3.22 through Fig. 3.27 illustrates decomposition output along with final parameters such as rank, alpha and gamma. Low-rank component(second row) captures different lighting variations across different subjects and are visually similar across different subject decompositions.

**Discussion**: In this chapter, we introduced a new low-rank matrix decomposition method, for learning simultaneously a nuisance factor space, and an identity space orthogonal to it. Based on our purpose of recognition, we formed an objective function to decompose the data matrix into meaningful components, where the low rank matrix models nuisance data, the sparse matrix models noise of individual data that is not captured by the common nuisance subspace, and most impor-
Figure 3.23: **Visual results of parameter selection,** $\alpha = .8, \gamma = 0$. This images illustrates visualization results using our main model (Eq. 3.6) using parameters alpha=.8 gamma=0 and obtained rank of Rank=26

Figure 3.24: **Visual results of parameter selection,** $\alpha = .5, \gamma = 0$. This images illustrates visualization results using our main model (Eq. 3.6) using parameters alpha=.3 gamma=0 and obtained rank of Rank=2
Figure 3.25: **Visual results of parameter selection, \( \alpha = 0.5, \gamma = 0.01 \).** This images illustrates visualization results using our main model (Eq. 3.6) using parameters \( \alpha = 0.5 \) \( \gamma = 0.01 \) and obtained rank of Rank\( = 7 \)

Figure 3.26: **Visual results of parameter selection, \( \alpha = 0.8, \gamma = 0.01 \).** This images illustrates visualization results using our main model (Eq. 3.6) using parameters \( \alpha = 0.8 \) \( \gamma = 0.01 \) and obtained rank of Rank\( = 27 \)
Figure 3.27: Visual results of parameter selection, $\alpha = 0.8, \gamma = 0.01$. This image illustrates visualization results using our main model (Eq. 3.6) using parameters alpha=.3 gamma=0.01 and obtained rank of Rank=2

...tantly, we learn a new matrix for identity representation, $B$, which is one of the main contributions of the chapter. We solve the corresponding objective function with an optimization procedure that builds on ALM and a Sylvester equation. The experimental section confirms the validity of the theoretical proposal of the chapter, in face recognition and even in a more general spectrum like person identification. We propose to represent data by their invariant components. By leveraging recent advances in low-rank matrix recovery, we develop a framework for the supervised learning of invariant components. This representation leads to a simple and efficient testing rule and promotes class separation. We have empirically verified the convergence of the training algorithm, and we have applied the model to the face recognition problem with highly corrupted training and testing data. The performance is very promising since they are on par or better than state-of-the-art, with significant gains in time complexity at testing time, and in classification accuracy at higher fractions of corrupted training data, and with small-size and corrupted training datasets. A preliminary test of the approach on the challenging person re-identification problem further assesses the validity of the invariant components as a discriminative data representation. While we have done all necessary to converge to the best possible framework in recognition using low rank decomposition and subspace learning, still promising results obtained from our proposed method opens a new path for the low rank decomposition learning methods. So far all the approaches based on low rank methods confined themselves to modeling the structure of modalities by a low rank matrix, however as we showed in this work, as this might not be helpful in many cases, doing the opposite which is modeling the structured noise by a low rank matrix is much beneficiary. Then
this can introduce a new concept for the low rank modeling of the various samples in computer
vision, although we investigated thoroughly all possible decomposition objective functions with
varying parameters, but we haven’t tried minor alterations that might improve the performance
of recognition, still there might be more efficient and better ways to do this modeling which one
might be interested in investigating, although we tested our method on current benchmark chal-
lenges and it is working properly however for datasets with large number of objects and samples
the complexity of solving Sylvester equation increase with $O(N^2)$ therefore it is prohibitive to use
our proposed method on large scale data sets, therefore it is also one of the areas that need to be
invested to improve the performance of the proposed solution for the objective function 3.28. Here
as an Example we will propose two new optimization problems that are intended to reach the same
objective that we were seeking in Chapter 3, but however with different formulation which may
lead in better result either computationally or from the accuracy aspect. One alteration that can
be made on 3.28 is as following:

\[(p6) : \langle A, E, Y \rangle = \arg \min_{A,E,Y} \text{rank}(A) + \alpha \|E\|_0 + \gamma \|(I - Q)Y^\top A\|_F^2 + \tau \|D - A - E - Y\|_F^2 \]  

(3.26)

where we call it a unified objective function where unlike 3.28 that has two different terms over
$Y$, in 3.26 both constraints on $Y$ are combined, and finding a solution for optimization of such a
problem may yield more impressive results.

On the other hand we have observed in some cases that the noise projected onto the orthogonal
complement sometimes is too high and causes high projection of sparse nuisance factors in identity
space. Therefore one of the ideas that worth investing in future is if we impose orthogonality for
the sparse factors as well, this might lead to better results in case of large amount of sparse noise,
therefore by imposing the constraint $Y^\top (A + E) = 0$, when the noise is projected to identity space
it would basically be equal to zero. over objective function can be formulated as

\[(p7) : \langle A, E, Y \rangle = \arg \min_{A,E,Y} \|A\|_* + \alpha \|E\|_1 + \gamma \|(A + E)^\top Y\|_F^2 + \beta \|Y(I - Q)\|_F^2 \quad \text{s.t.} \quad D = A + E + Y \]  

(3.27)
3.9 Derivative of B

Problem:
\[ \partial (\| B^T (BB^T)^{-1} B - Q \|_F^2 ) / \partial (B) \]

\[ \| B^T (BB^T)^{-1} B - Q \|_F^2 = Tr \{ (B^T (BB^T)^{-1} B - Q)(B^T (BB^T)^{-1} B - Q)^T \} \]

\[ \partial (\| B^T (BB^T)^{-1} B - Q \|_F^2 ) / \partial (B) = \partial \{ Tr (B^T (BB^T)^{-1} B - Q) \} / \partial (B) \]

\[ = \partial \{ Tr \{ (B^T (BB^T)^{-1} BB^T (BB^T)^{-1} B) - QB^T (BB^T)^{-1} B \}
\]
\[ - B^T (BB^T)^{-1} BQ^T + QQ^T \} \} / \partial (B) \]

\[ = \partial \{ Tr \{ (B^T (BB^T)^{-1} BB^T (BB^T)^{-1} B) \} + Tr \{ QB^T (BB^T)^{-1} B \} +
\]
\[ Tr \{ B^T (BB^T)^{-1} BQ^T \} + Tr \{ QQ^T \} \} / \partial (B) \]  (3.28)

\[ \partial (Tr \{ (B^T (BB^T)^{-1} B) \} / \partial (B) \} = (BB^T)^{-1} B - 2(BB^T)^{-1} BB^T (BB^T)^{-1} B
\]
\[ + (BB^T)^{-1} B = 2(BB^T)^{-1} B - 2(BB^T)^{-1} BB^T (BB^T)^{-1} B = 0 \]

\[ \partial (Tr \{ QB^T (BB^T)^{-1} B \} / \partial (B) \} = (BB^T)^{-1} BQ - 2(BB^T)^{-1} BQ^T B^T (BB^T)^{-1} B
\]
\[ + (BB^T)^{-1} BQ^T = 2(BB^T)^{-1} BQ - 2(BB^T)^{-1} BQ^T B^T (BB^T)^{-1} B \]

\[ \partial (Tr \{ B^T (BB^T)^{-1} BQ \} / \partial (B) \} = (BB^T)^{-1} BQ - 2(BB^T)^{-1} BQ^T B^T (BB^T)^{-1} B
\]
\[ + (BB^T)^{-1} BQ^T = 2(BB^T)^{-1} BQ - 2(BB^T)^{-1} BQ^T B^T (BB^T)^{-1} B \]

\[ \partial (\| B^T (BB^T)^{-1} B - Q \|_F^2 ) / \partial (B) = 4(BB^T)^{-1} B(Q)B^T (BB^T)^{-1} B
\]
\[ - 4(BB^T)^{-1} BQ \]
Chapter 4

Maximal Margin Invariant Subspace

4.1 Introduction

Domain Adaptation (DA) and Generalization (DG) have gained a lot of attention during the last couple of years. Domain Adaptation aims to relate the knowledge in source domain (training dataset) to the target domain (testing dataset), while we have some knowledge about the target domain, for example, unlabeled target domain samples are provided along with source domain samples. Domain Generalization (DG) is the task of learning best knowledge from the source domain that can be used on any target domain while we don’t have any information from the target domain, despite DA that unlabeled samples from target domain are provided. Therefore the purpose of DG is to learn a domain-invariant classifier. Domain generalization and adaptation have been introduced to address the problem of domain shift and distribution mismatch between target and source domains in visual recognition which is also called dataset bias[47]. Existing techniques are dealing with this issue either by finding invariant domain representation or introduce robust classifiers to overcome this distribution mismatch. However, most of these methods are not considering these tasks simultaneously; Therefore their solutions are not optimal. In this chapter, we are introducing a novel domain generalization method that will solve this problem with a global function where it will simultaneously find the invariant representation and the optimum classifier over this representation.

Domain adaptation problem has been approached in two different scenarios of unsupervised and semi or weakly supervised settings. The unsupervised setting is the case in which category label is only and only provided for data from the source domain, [142, 143, 144, 145, 146, 147, 148] are some of the recent works proposed unsupervised methods. The semi or weakly supervised setting is the scenario where some labeled samples from target domain are provided in addition to labeled source samples [121, 149, 150, 151, 152, 153, 154]. There is a great discussion
about the nature of the bias in existing databases in computer vision community in [47]. In this paper, Torralba etc. argues that no matter how perfectly the dataset collection and labeling is done, Unfortunately, all the datasets labeled by human are subject to bias. This dataset bias precludes generalization of learned methods to other datasets. To tackle this problem, there have been two parallel efforts independent of each other that address this issue, domain generalization approaches are proposed to learn robust classifiers without exposing any data from target domain to the classifier[146, 149, 155, 156, 157, 158, 159]. On the other front, new techniques and implementations of convolutional neural networks are introduced and practiced that attempt to learn visual categories from millions of web-based sample images, which although successful in learning robust representations and improve the situation where an increasing number of source data exposed to learning algorithms decreases the probability of the bias. However, recent experiments have proved that it doesn’t solve the problem completely and still there is a need for robust methods to attenuate dataset bias effect[155, 156, 160, 161].

Several methods have introduced intermediary representations to relate the target and the source domains [144, 147, 150, 151, 161, 162, 163, 164]. Although useful but they are heavily dependent to biased representations, which may make their solution sub-optimal for classification. On the other hand, sample selection or re-weighting methods [143, 165] fail to address the fact that domain shift may have distorted image features themselves. Also, many of the recent works attempt to solve the problem of feature adaptation and learning a classifier separately, which leads to a sub-optimal solution [145]. So to address this issue in this work, we propose to tackle domain shift by extracting the information that is discriminative across all domains and leave out the nuisance factors (i.e. dataset bias, background features,...). As we will discuss later, these nuisance factors belong to a particular domain and are considered to be nuisance with respect to data of other domains and are orthogonal to max margin decision boundary. We accomplish this with a unified objective function that finds the optimal representation (or shift) and max marginal decision boundary on the new representation.

In this chapter, we are addressing domain generalization problem without any prior constraint on training data. We are proposing a new method for learning unbiased max margin classifiers which are robust against nuisance factors or domain biases given samples from multiple source domains. Our proposed method doesn’t require domain labels or necessarily target domain data for adapting the domain shift and learning the max margin support vectors, so it is an entirely unsupervised adaptation method, despite most of the works proposed that need the target domain to find the intermediary subspace. It has been shown that existence of nuisance features in machine learning during training, not only increases the complexity and convergence time of the learning algorithm, it deteriorates the learn-
ing accuracy and eventually drops the classification performance. Even more with the explosion of the online image and video dataset the importance of robust learning methods to overcome the challenge of nuisance factors in the object manifestation (i.e. overcoming features extracted from background across different categories) in this pool of images and videos is felt more than ever. These common nuisance factors naturally can be divided into low-level factors such as occlusion, clutter, viewpoint changes, image saturation or background nuisance features common across different categories and geometric or photometric variations; On the other hand, high-level factors are such as dataset bias or label bias in scalable applications. Numerous publications discuss intuitively and empirically illustrate that almost all attempts to come up with an unbiased dataset have failed[47, 48], and best approach to handle dataset bias is to model it during learning and equip the learning machine to model the dataset bias and eliminate it from the equation. Naturally, some of these variations are inherently linear and of additive nature. In this work we propose a new method in which we explore and subtract the nuisance subspace or in a more conventional paradigm the current bias in training data and learn large margin hyperplane on the remanent of the data. Usually, both of high-level and low-level nuisance factors appear in structured elements that can be modeled as linear subspaces Developments in the area of the low-rank minimization have led to emergence of powerful tools for the rank minimization problems that suites for various applications, It is well known that robust PCA can capture dominant global data distribution information and PCA is used widely for dimension reduction in classification applications, there are numerous applications of the robust principle component analysis in video or image processing that successfully disintegrate the background and foreground or the bias from the real object of interest[12]. In this work, we use matrix factorization to obtained the bias over data and since the shared subspace (bias) is predominantly orthogonal to separating hyperplane we add the orthogonality constraint to distinguish the bias better. Therefore the learned hyperplane would be a robust decision boundary that represents the actual classification hyperplane. One might raise the question that non-linear SVM is introduced to address these variations and using a custom kernel can resolve the problem mentioned above, but still, these methods fall short in situations that there is a structured drift in data as well as there is not a deterministic solution for kernel selection or custom kernel design. Our method can automatically analyze the input data, detect the bias and nuisance factors and eliminate them in learning procedure. Therefore the learned pair of bias and decision boundary can be used in testing time to map the data into the expected margin. In this method using an unsupervised max margin learning we are learning a decision boundary, however, since this decision boundary is prone to learn the database bias we use an unsupervised robust principle component analysis to subtract the bias factor from discriminatory data. Eventually, each learned support vector will have a corresponding bias factor which will be used in testing time to eliminate
the data bias.
The rest of the paper is organized as in the following. Section 4.2 is devoted to the review of the previous works on domain generalization or adaptation. 4.3 describes our low-rank domain generalization method, the formulates the solution and optimization method used for the introduced method. Eventually section 4.4 is dedicated to experimental results performed on domain adaptation benchmarks using state of the representation, and its comparison to the state of the art methods on DA.

4.2 Related Works

During last decade there has been an enormous amount of work to improve the performance of large margin classifiers. [166] proposed a new objective function, in which incorporates both LDA and SVM in one problem. LDA increases the discriminative power of the projection by considering the global distribution of the data, and SVM minimizes the decision boundary error. They accomplish this by defining a new objective function unlike RMM [167], which adds new constraints to the SVM problem. [167] added a new limiting constraint over decision hyperplane, incorporating this additional constraint yields a significant margin while normalizing the solution based on the spread of the data. [168] combined SVM with NDA pretty much using the same structure used in [166]. Following these methods, we propose a new method to increase the margin for data acquired in different domains and obtain the optimum subspace to generalize the obtained support vectors to domains that are never exposed.

There has been an enormous amount of work on the bias of existing databases in the computer vision society, and it sounds like, despite the great effort to create unbiased and generalizable datasets, dataset bias is an inevitable part of the existing computer vision datasets [47]. However, there have been some efforts to devise new models to learn from dataset bias and generalize existing tools to be used with amalgamate of images from different datasets. [48] introduced a new additive linear bias term for each dataset and in a single objective function attempted to find the max margin hyperplane and bias for each dataset. [169] learns two linear transfer functions that map input space into a common space, which is supposed to model non-dataset specific information and an additive orthogonal linear projection which models the dataset bias. Domain adoption using subspace methods has proven to produce effective solutions with less computational complexity [144, 147, 161]. [144] proposed a new intermediary subspace along the geodesic path of source and target domain on Grassman manifold; then they learned a classifier on the projection of the data from source domain on this subspace. [142] extended method proposed by Gopalan et al to deal with polynomial curves on Grassman manifold so can handle multiple source domains. [147]
also uses a geodesic path, where they model incremental changes between source and target domain on a flow kernel. [161] takes a different approach where instead of learning an intermediary subspace, it learns a transformation to map source domain to target domain, this mapping is learned through an optimization to minimize the discrepancy between the source and target domain. [164] extended Fernando’s work by adding a selection of landmarks between source and target domains. Even though most of the methods proposed for subspace modeling are linear they can be easily extended through feature mapping [170]; these methods used different criteria for subspace modeling such as Canonical Correlation Analysis (CCA) [151], Transfer Component Analysis (TCA) [150] Transfer Subspace Learning [163]. The problem with domain adoption methods is that they are not solving the problem of generalizability and are only resolving the bias of the source domains and are adopting it for the target domain. In our approach, we are learning the bias over support vectors from various source domains and by eliminating the bias of each dataset collectivity converge to a most discriminative corner of the space, and converges to unbiased large margin subspace. During the last couple of years, there have been enormous attention for Domain adoption and plenty of methods have been proposed for learning on source domain and test on the target domain. Even though this alleviates the concerns about generalizability and overfitting on an eccentric dataset bias, still there is the concern that proposed methods are learned for transforming from one dataset to another dataset. However, in order to address this concern we propose our method to learn on data from multiple datasets and test them on multiple datasets. Our work relates to another thread of works that attempt to find latent domains, where they use clustering with different criteria like MMD [149, 153] and then a different classifier is learned for each one of domains [156, 171]. [156] built on the Exemplar-SVM method [172], where they assumed likelihood probability function of samples from the same latent domain would reside in the same subspace. The ensemble matrix of these likelihood functions of exemplar weight vectors will yield a low-rank matrix, and they utilized an alternating solution for this nonlinear optimization problem. The problem with these methods is that they are sensitive to noisy data and their solution is suboptimal since they don’t define a global optimization problem. Moreover, our work is related to [173, 174], [173] is proposing a joint feature representation and transfer knowledge using a structured low-rank constraint. [174] added a subspace mapping function for source domain data in the max margin learning problem. It is while our method, is learning a Max-margin decision boundary jointly by considering the low-rank structure of the bias.
4.3 Unsupervised Low Rank Domain Generalization

In this section, we propose our unsupervised low-rank Domain generalization (ULRDG) method, and then we will describe the optimization procedure. For ease of presentation a vector/matrix is denoted by lowercase/uppercase letters in boldface. $\mathbf{0}_n, \mathbf{1}_n$ denotes n-dimensional column vectors of zeros and ones, when the dimension is obvious we use $\mathbf{0}, \mathbf{1}$ respectively. $A \odot B$ is the Kronecker or element-wise product of two matrices with the same size. Suppose $X$ is an aggregation of data points from different domains, since our method is fully unsupervised we don’t require any information from domains of even source data. So suppose N training samples belonging to $C$ classes from various domains are given as $\{(x_1, l_1), \ldots, (x_N, l_N)\}$, where $x_i$ is the $i$-the data sample, and $l_i \in \{1, \ldots, C\}$ is the corresponding class label of $x_i$. In the following, we first introduce our risk minimization problem and consequently we will introduce components of each function for binary and linear classification and eventually we will present the multi-class and non-linear extension of the proposed method.

4.3.1 Problem Setup

Our goal in this work is to separate additive dataset bias from $X$ and find the portions of the data points that yield less marginal classification error. We use support vectors of each category for finding the best decision boundary; however, the decision boundary is not based on the raw input data. First, we subtract additive dataset bias from input features. Given that domain data mismatch is due to distribution shift or bias, and it is of homogeneous additive drift nature, and naturally as discussed in the previous section it is low rank, using an additive low-rank decomposition, we would be able to model this distribution shift or bias. Therefore we introduce the following risk

![Figure 4.1: Regular SVM vs. proposed method.](image-url)

In this figure, we illustrate the main idea of the paper, where while SVM learned on a single domain act entirely to separate class labels, It fails when new datasets are introduced. Our proposed method attempts to increase the marginal distance of datasets and find the best subspace (a) that is inclusive for all datasets and captures the best invariant subspace for a category that is generic among various domains.
minimization problem:

$$\min_{\omega, a, b} \mathcal{L}(w, a) + \nu \Theta(w, b)$$

s.t.  \( x = a + b \)  \( (4.1) \)

Where \( \mathcal{L}(w, a) \) is the loss function for miss-classification over training data where we use well know max margin decision boundary and \( \Theta(w, b) \) is the bias regularizer in which minimizes the rank of obtained bias matrix and keeps the bias in orthogonal space of the decision boundary(since bias stands for the data drift which obscures a large margin decision boundary). As it can be seen this a joint minimization problem that simultaneously aims to find the best feature space \( a \in \mathbb{R}^m \), and the best max marginal decision boundary \( \omega \in \mathbb{R}^m \) over these new feature space. In the following section, we will introduce the risk function’s elements and will discuss the optimization method in detail.

### 4.3.2 Robust SVM

We assume that a set of \( N \) training data samples from \( K \) different entities or object classes (e.g. features extracted from images of multiple object categories from various datasets) are given, where each class \( i \) has \( k_i \) samples (these data might come from various domains, we don’t differentiate between them in the formulation). We concatenate the data into a matrix \( X = [x_1, x_2, \cdots, x_N] \in \mathbb{R}^{m \times N} \), where \( X \) is the training data matrix obtained by lining up all the training samples. We introduce following objective function as our boundary learning problem.

$$\min_{\omega, \xi, c, A, B} \frac{1}{2} \|\omega\|_2 + \frac{\alpha}{2} \|\omega^T B\|_2 + \beta \|B\|_* + C \sum_{i=1}^{N} \xi_i$$

s.t.  \( y_i((\omega, a_i) + c) \geq 1 - \xi_i \)  \( \forall i = 1, \ldots, N \),

\( X = A + B, \quad \xi_i \geq 0 \)  \( (4.2) \)

Here \( B = [b_1, b_2, \cdots, b_N] \in \mathbb{R}^{m \times N} \) is a low rank matrix, in which models dataset bias in the training data matrix \( X \). \( A = [a_1, a_2, \cdots, a_N] \in \mathbb{R}^{m \times N} \) is the sufficient subspace in which the support vector machine is learnt based upon. \( y_i \in \{-1, 1\} \) is the class label for each data category in the binary classification. In order to deploy this formulation for multi-class classification we simply learn \( C \) classifiers for each class. For the sake of better representation, we keep the formulation simple for binary case, however in the following we will describe the formulation for multi-class case.
$\|\omega\|_2$ is to make sure that learned decision boundary has the maximum margin from the classes and the boundary doesn’t converge to infinity. $\|\omega^T B\|_2$ is the orthogonality constraint that ensures the dataset bias and nuisance factors are orthogonal to the decision boundary, in which by using this term we ensure that we are not losing any discriminative information only bias and nuisance factors. $\|B\|_*$ is the nuclear norm of the matrix $B$, It is shown that nuclear norm is the best convex approximation of the rank function; therefore the nuclear norm is used to minimize the rank of the bias matrix, then the shared subspace across all of the data is obtained. $y_i((\omega, a_i) + c) \geq 1 - \xi_i$ is the decision rule and defines the classification decision boundary and finally $X = A + B$ ensures data fidelity, $\alpha$ and $\beta$ are tradeoff parameter that we obtain using a grid search. Since this is a convex problem we will solve it using two Lagrangian multipliers, we use the regular Lagrangian multiplier for support vectors and augmented Lagrangian multiplier for data fidelity term. Therefore the optimization problem after adding Lagrangian multipliers will be as following:

$$
\mathcal{L}(\omega, \xi, A, B, c) = \frac{1}{2} \|\omega\|_2^2 + \frac{\alpha}{2} \|\omega^T B\|_2^2 + \beta \|B\|_* + C \sum_{i=1}^N \xi_i - \sum_{i=1}^N \lambda_{1i} \{y_i((\omega, a_i) + c) - 1 + \xi_i\} + \langle \lambda_2, X - A - B \rangle + \frac{\mu}{2} \|X - A - B\|_2^2
$$

Where $\lambda_1 = [\lambda_{11}, \lambda_{12}, \ldots, \lambda_{1N}] \in \mathbb{R}^N$ and $\lambda_2 \in \mathbb{R}^{m \times N}$. In here, we use sequential iterative optimization method for solving the problem for all the parameters, where we find the optimum value for one of the variables while the other variables are fixed and update the variable by the rules that we extract. In the following we will rewrite the lagrangian objective function for each variable and then introduce the optimization procedure that we use to update each variable.

**Updating $\omega$ and $c$:** In the following we rewrite the Eq. 4.3 for the parameter $\omega$:

$$
\mathcal{L}(\omega, c) = \frac{1}{2} \|\omega\|_2^2 + \frac{\alpha}{2} \|\omega^T B\|_2^2 - \sum_{i=1}^N \lambda_{1i} \{y_i((\omega, a_i) + c) - 1 + \xi_i\}
$$

Since this is a convex problem with respect to $\omega$ the solution of $\frac{\partial \mathcal{L}(\omega)}{\partial \omega} = 0$ is as below:

$$
\dot{\omega} = (I + \alpha BB^T)^{-1} \sum_{i=1}^N \lambda_{1i} y_i a_i
$$

80
Algorithm 2 Robust Support Vector Machine Learning via Augmented Lagrangian and the Exact ALM Method

Require: Observation matrix $X$, labels $Y$, and penalty weights $\alpha, \beta$, $\rho > 1$; $\mu_0 > 0$; $\eta = \|X\|_F^2$; $\lambda_0 = \frac{\text{sgn}(X)}{\max(\|\text{sgn}(X)\|_F^p, \alpha^{-1} \|\text{sgn}(X)\|_\infty)}$; $B_0 = (I - \omega_0 \omega_0^T)D$; $A_0 = XQ$;

1: $Q = Y^T Y / \text{normcol}(Y^T Y)$;
2: while not converged do
3: $j = 0$; $A_{j+1} = A_j$; $B_{j+1} = B_j$;
4: while not converged do
5: $(U, \Sigma, V) = \text{svd}(X - A_j + \mu_k^{-1} \lambda_2 - \frac{\alpha}{2 \rho \mu \lambda_2 \omega_k \omega_k^T} \omega_k \omega_k^T B_k^T)$; $B_{j+1} = U \Sigma (\eta \mu_k - 1) V^T$
6: Update $A_k$ using eq. (4.12) with $B_{j+1}^T$
7: $j \leftarrow j + 1$
8: end while
9: $B_{k+1} = B_{j+1}^T$; $A_{k+1} = A_{j+1}^T$;
10: $\mu_{k+1} = \rho \mu_k$; $\lambda_{2(k+1)} = \lambda_{2k} + \mu_k (X - B_{k+1} - A_{k+1})$
11: Update $\lambda_1$ by solving the (4.8)
12: Update $\omega$ using (4.5) and $B_{K+1}$ and $A_{K+1}$
13: Update $c$ using (4.9)
14: $k \leftarrow k + 1$
15: end while

Ensure: $\omega_k$, $c_k$

On the other hand, we can compute the following rule for the bias term $c$ following the same rule used for $\omega$:

$$
\mathcal{L}(c) = -\sum_{i=1}^{N} \lambda_{1i} y_i ((\omega, a_i) + c) \rightarrow \sum_{i=1}^{N} \lambda_{ij} y_i = 0
$$

(4.6)

Given equations 4.5,4.6 independent of the rest of the optimization problem, the KKT conditions for updating $\omega$ and $c$ will change according to these equations, therefore we can write the Wolfe Lagrangian dual function for solving $\omega$ as in the following:

$$
\min_{\lambda_1} \sum_{i=1}^{N} \lambda_{1i} - \frac{1}{2} \sum_{i,j=1}^{N} \lambda_{1i} \lambda_{1j} y_i y_j a_i^T (I + \alpha BB^T)^{-T} a_j
$$

s.t. $0 \leq \lambda_{1i} \leq C$ $\forall i = 1, ..., N$.

(4.7)

The above equation can be written in the following Matrix-Vector format:

$$
\min_{\lambda_1} \frac{1}{2} \lambda_1^T K_b \lambda_1 - 1^T \lambda_1
$$

where $K_b = y_i y_j a_i^T (I + \alpha BB^T)^{-T} a_j$

(4.8)

Where $\lambda_1 = [\lambda_{11} \lambda_{12} \ldots \lambda_{1N}]^T \in \mathbb{R}^N$, it is very well known that optimum solution for the $\lambda_1$ in the Eq.4.8 for large set of data is the iterative solution of a system of linear equations $K_b \lambda_1 = 1$
subject to $0 \leq \lambda_1 \leq C[175]$.

The motivation for using an explicit bias term is to decrease the number of Support vector machines learned, and consequently reduce the learning time and increase the efficiency. For updating the bias term, since it is independent of the other parameters, we can follow steps in [176] or [177], therefore following steps in [176], we augment the cost function of (4.2) with $1/2k\alpha^2$ and this will lead to following constraint instead of $\sum_{i=1}^{N} \lambda_1 y_i = 0$ in (4.6):

$$\frac{\partial L(\hat{c})}{\partial c} = 0 \quad \Rightarrow \quad \hat{c} = \frac{1}{k} \sum_{i=1}^{N} \lambda_1 y_i$$ (4.9)

**Update rule for B:** rewriting the equation in (4.3) for the variable $B$, we will get the following:

$$L(B) = \alpha \|\omega^T B\|_2 + \beta \|B\|_* + \lambda_2, X - A - B > + \frac{\mu}{2} \|X - A - B\|_2$$

$$L(B) = \|B\|_* + h(\omega, A, B, \lambda_2, \mu) - \frac{1}{2\beta\mu} \|\lambda_2\|_2$$ (4.10)

Where $h(\omega, A, B, \lambda_2, \mu) = \alpha \|\omega^T B\|_2 + \frac{\mu}{2\beta} \|X - A - B + \frac{\lambda_2}{\mu}\|_2$, therefore since $h$ is a convex problem in addition to a convex nuclear norm, it is convenient to use the linearization technique of the LBDMBP method [120], very effectively used also by other approaches [45, 103, 121], and replace the quadratic term $h$ with its first order approximation, computed at iteration $k$, and add a proximal term, giving the following update

$$B_{k+1} = \arg\min_B \|B\|_* + \frac{\eta\mu}{2\beta} \|B - B_k\|_F + B - B_k, \nabla_B h(\omega_k, A_k, B_k, \lambda_k, \mu) >$$

$$= \arg\min_B \|B\|_* + \frac{\eta\mu_k}{2\beta} \|B - B_k - (X_k - A_k - B_k) - \frac{\alpha}{\mu_k} \omega^T B_k + \frac{\lambda_k}{\mu_k}\|/\eta_2, \quad (4.11)$$

The optimization problem in this equation is well studied in [12] and it is solved by an iterative approach where a singular value thresholding function is used as it is shown in the algorithm (2). The above formulation is valid for the case of binary classification and for multi class framework we simply use $\Omega\Omega^T$ instead of $\omega\omega^T$, where $\Omega = [\omega_1 \omega_2 ... \omega_\mathcal{C}] \in \mathbb{R}^{m \times \mathcal{C}}$ and $\omega_i$ is the decision boundary learned for classification of class $i$ against rest of the classes.

**Update rule for A:** Rewriting the Eq. in (4.3) for A, we’ll get the following:

$$L(A) = \frac{\mu}{2\beta} \|X - A - B + \frac{\lambda_2}{\mu}\|_2 - \sum_{i=1}^{N} \lambda_1 y_i ((\omega, a_i) + c)$$ (4.12)
Since the obtained equation for $A$ is a convex equation with respect to $A$, we can easily compute the following closed form solution $\frac{\partial L(A)}{\partial A} = 0$, for updating $A$ (for better representation, sum is converted to vector-matrix format):

$$A_{k+1} = \frac{\beta}{\mu_k} \omega_k (\lambda_1 \odot y)^\top + X - B_{k+1} + \frac{\lambda_{2k}}{\mu_k}$$

(4.13)

Where $\lambda_1 = [\lambda_{11} \lambda_{12} \ldots \lambda_{1N}]^\top \in \mathbb{R}^N$ and $y = [y_1 y_2 \ldots y_N]^\top \in \mathbb{R}^N$ and $\odot$ is the Kronecker product of two vectors and $y_i \subset \{-1, 1\}$, please note that the term $\frac{\beta}{\mu_k} \omega_k (\lambda_1 \odot y)^\top$ insures that matrix $A$ will preserve discriminative content of input feature, and will move the data point in a direction that increases the marginal distance of each category, the rest of the terms are to insure the data fidelity. For the multi-class formulation, this term in the update rule of $A$ (one-vs-all scheme) can be written as $\sum_{j=1}^N \frac{\beta}{N \mu_k} \omega_j^i (\lambda_1 \odot y)^\top$, where $\omega_j^i$ corresponds to decision boundary learned for class $j$.

and the solution, reported in algorithm 2, is still obtained with an instance of the singular value thresholding algorithm [61].

And eventually $\lambda_2$ is updated using following equation assure the data fidelity. For detailed information on the optimization please refer to algorithm 2.

$$\lambda_{2(k+1)} = \lambda_{2(k+1)} + \mu_k (X - A_{k+1} - B_{k+1});$$

(4.14)

### 4.3.3 Multi-class Learning and Testing

The formulation presented in the previous section is for binary classification; however, most of the real world computer vision problems include more than two classes. To generalize our method and experiment for multi-class, we simply follow the well-known one-vs-all framework. Using one-vs-all framework will produce $K$ decision boundaries which in a matrix stack will be denoted as $\Omega = \{\omega_1 \omega_2 \ldots \omega_C\} \in \mathbb{R}^{m \times C}$, to carry out our experiments we simply use a voting scheme for making a decision, our additive component segregation model is aimed to separate the optimal marginal complement for classification along with corresponding additive drift in the dataset, as it can be seen in equation (4.5) $\omega$ is formed of unbiased support vectors $a$. Then basically we compare new test data points against our learned unbiased points in a max margin formulation. Then following the regular formulation for max-margin the decision function would be as following:

$$\hat{y}_j = \arg\max_k f_k(x_j) = \arg\max_k \omega_k x_j + c_k$$

(4.15)

where $\omega$ is obtained using Eq. (4.5).
4.4 Experiments:

Here in this section, we evaluate our domain generalization method on several visual recognition benchmarks. There are enough amount of work that show current learning methods including regular max-margin methods are not able to generalize to new datasets, and classifier’s performance dropped when learned on source domain and tested on target domain [47, 48, 178]. We don’t see a reason to repeat these experiments solely to show that dataset bias exists and we urge interested readers to read these works.

We divide our experimental section into two main sets of experiments. In the beginning, we test and compare the recognition accuracy of our proposed model against the recent solutions proposed for domain generalization and adaptation. Please note that we don’t restrain ourselves to domain generalization methods and we compare the accuracy of our method even against domain adaptation methods that need data from target domain (unlike our method). Next, to further challenge the broader applications of the proposed model, we test the ability of the proposed method for learning from disguised training data (such as glass or scarf occlusion in the face dataset i.e. AR face dataset). In the following, we will describe each one of our experiments more in depth.
### Table 4.2: Domain adaptation on office-caltech using DeCAF features.
Comparison of recognition accuracy of unsupervised DA methods against our proposed method

<table>
<thead>
<tr>
<th>Source</th>
<th>Target</th>
<th>A, C</th>
<th>W, D</th>
<th>C, D, W</th>
<th>A, D, W</th>
</tr>
</thead>
<tbody>
<tr>
<td>L-SVM</td>
<td>W, D</td>
<td>82.68</td>
<td>76.06</td>
<td>90.73</td>
<td>84.51</td>
</tr>
<tr>
<td>1HNN</td>
<td>A, C</td>
<td>83.41</td>
<td>76.49</td>
<td>92.13</td>
<td>85.89</td>
</tr>
<tr>
<td>Sub-C[180]</td>
<td></td>
<td>82.61</td>
<td>78.65</td>
<td>90.75</td>
<td>85.21</td>
</tr>
<tr>
<td>[171][Ensemble)</td>
<td>[171][Match)</td>
<td>84.01</td>
<td>77.11</td>
<td>91.65</td>
<td>87.06</td>
</tr>
<tr>
<td>[153][Ensemble]</td>
<td>[153][Match)</td>
<td>79.23</td>
<td>68.06</td>
<td>80.75</td>
<td>80.32</td>
</tr>
<tr>
<td>UML[178]</td>
<td></td>
<td>82.29</td>
<td>79.54</td>
<td>91.02</td>
<td>84.59</td>
</tr>
<tr>
<td>LRE-SVMs[172]</td>
<td></td>
<td>84.59</td>
<td>81.17</td>
<td>91.87</td>
<td>86.38</td>
</tr>
<tr>
<td>D-MTAE+1HNN[155]</td>
<td></td>
<td>85.35</td>
<td>80.52</td>
<td>93.13</td>
<td>86.15</td>
</tr>
<tr>
<td>Our Method</td>
<td></td>
<td>86.34</td>
<td>80.97</td>
<td>92.26</td>
<td>88.56</td>
</tr>
</tbody>
</table>

In this section, we compare domain generalization performance of our proposed method against the state-of-the-art DA methods on the benchmark dataset of office+caltech10 [152]. This dataset is composed of images from four different domains: Amazon, DSLR, Webcam and Caltech. Office dataset consists of 3 domains named as Amazon, DSLR and Webcam images. Each one of these domains is composed of 31 common categories in the total of 4652 images. Amazon is a dataset acquired in a strictly controlled lighting and studio conditions downloaded from online merchants. DSLR images are high resolution images taken using a digital SLR camera under home lighting conditions. Webcam images are taken under similar circumstances to DSLR but using a low-resolution webcam. These datasets capture a good intra-class variation of 31 categories. To add more variety to the domains following footsteps of [147] caltech256 is added to these domains. Following [147] we utilize 10 common classes among all categories available in four domains, this yields 2533 images in total. Fig. 4.2 illustrates sample images from these four domains. As it is shown in [161] z-normalization on features in general boosts the classification accuracy then we perform z-normalization on image representations, Also since comparison of non-sub space DA and subspace DA methods in [161] indicates that subspace methods in general outperforms other methods on DA benchmarks, we mostly compare to outperforming subspace methods such as GFK[147], SA[161, 179], TCA[150], JCSL[174], UML[178], E-SVMs[156].
Table 4.3: Domain adaptation on office-caltech using landmark features. Comparison of recognition accuracy of unsupervised DA methods against our proposed method

<table>
<thead>
<tr>
<th>Domain</th>
<th>A → C</th>
<th>A → D</th>
<th>A → W</th>
<th>C → A</th>
<th>C → D</th>
<th>C → W</th>
<th>W → A</th>
<th>W → C</th>
<th>W → D</th>
</tr>
</thead>
<tbody>
<tr>
<td>NO ADAPT-SVM</td>
<td>41.7</td>
<td>41.4</td>
<td>34.2</td>
<td>51.8</td>
<td>54.1</td>
<td>46.8</td>
<td>31.1</td>
<td>31.5</td>
<td>70.7</td>
</tr>
<tr>
<td>NO ADAPT-INN</td>
<td>26.2</td>
<td>25.5</td>
<td>29.8</td>
<td>23.7</td>
<td>25.5</td>
<td>25.8</td>
<td>23.0</td>
<td>20.0</td>
<td>59.2</td>
</tr>
<tr>
<td>GFK-SVM[147]</td>
<td>42.2</td>
<td>42.7</td>
<td>40.7</td>
<td>44.5</td>
<td>43.3</td>
<td>44.7</td>
<td>31.8</td>
<td>30.8</td>
<td>75.6</td>
</tr>
<tr>
<td>JCSL[174]</td>
<td>42.6</td>
<td>42.5</td>
<td>47.6</td>
<td>44.3</td>
<td>46.5</td>
<td>46.4</td>
<td>41.3</td>
<td>35.1</td>
<td>74.2</td>
</tr>
<tr>
<td>SA[161, 179]</td>
<td>39.9</td>
<td>38.8</td>
<td>39.6</td>
<td>39.4</td>
<td>38.9</td>
<td>39.3</td>
<td>31.8</td>
<td>31.8</td>
<td>77.9</td>
</tr>
<tr>
<td>Our Method</td>
<td><strong>43.8</strong></td>
<td><strong>43.6</strong></td>
<td>46.8</td>
<td><strong>52.1</strong></td>
<td>49.6</td>
<td><strong>48.2</strong></td>
<td>37.4</td>
<td><strong>35.4</strong></td>
<td><strong>86.62</strong></td>
</tr>
</tbody>
</table>

DeCAF features:

To evaluate the strength of our method in conjunction with state-of-the-art feature extraction methods, we examine our method on Office-Caltech with DeCAF$_6$ features proposed in [7, 181]. It is illustrated empirically in [160] that even using convolutional neural networks which learns thousands of non-linear parameters over millions of training samples, is not enough to address dataset bias problem. More recent works on domain generalization have tested their method using this state-of-the-art features[156]. And [182] has shown that convolutional network layers provide generic mid-level image representations that can be used for other tasks. So we follow the same footsteps in [156] to conduct our experimental section here. DeCAF is obtained by inputting warped (256x256) RGB raw image values to CNN architecture implemented in$^1$ [181]. 4096 activation values of neurons from the layer 6 (subscript indicates layer) are selected to represent the whole image. Tables 4.1 and 4.2 illustrate the results obtained using the DeCAF$_6$ features on recent domain adaptation methods and its comparison to our method. Comparing results obtained with DeCAF and SURF features(next section), clearly indicates that DeCAF$_6$ features outperforms the SURF features with a large margin, and DeCAF$_b$etter represents the objects in general. DeCAF$_b$ is aimed to learn from million of images, therefore since during learning, training algorithms are exposed to million of images from the same category with different variations and from different domains, therefore many domain adaptation methods either deteriorate the accuracy of these representations or slightly improves the accuracy which is not significant. Table 4.1 illustrates the accuracy results obtained from learning on source domain and testing it on the target domain. We compare our method to unsupervised domain adaptation methods even though our method doesn’t need the data from the target domain. Results in table 4.1 clearly illustrate that our method outperforms domain adaptation methods and SVM without adaptation with a large margin. This clearly indicates that there is an additive bias in layer 6 of DeCAF features that our method is able to successfully separate these nuisance factors and give a boost to the accuracy of the soft margin classifier with the newly adopted decision boundaries. Also in order to illustrate the performance of our method using DeCAFF features against recent domain generalization methods such

as exemplar SVMs (E-SVMs) [172], low rank exemplar-SVMs (LRE-SVMs) [156], discriminative sub-categorization (Sub-C) [180], unbiased metric learning (UML) [178] and methods proposed in [171] and [153]. We also compare against two baseline methods of linear SVM (L-SVM) and nearest neighbour (1NN) methods. We follow the experimental setup used in [156, 171], the results obtained illustrate that our method outperforms in 3 out of 4 experiments performed using a different combination of source and target domains. The results illustrate that only one method outperforms our method in one experiment and this method is based on nearest neighbor classification, and our method outperformed all the methods that use max margin classification method.

**Land mark Surf features:**

In order to yield a fair assessment of the accuracy performance of our proposed method with well-studied domain adaptation methods, we follow an experimental setup used in [145, 147, 152, 179] using SURF features. In this experiment features are extracted from rescaled grayscale images, features are encoded into an 800-bin histogram using a codebook computed via a K-means clustering on a random subset. Table 4.3 illustrates the obtained results using SURF features on office-caltech10 dataset, despite the fact that all the methods compared against in this table use unlabeled data from target domain, we don’t use any data from target domain, still our method outperforms most of these DA methods.

### 4.4.2 Action Recognition

In this section, we further examine the performance of the proposed method on activity recognition across different domains, which is referred to a camera set in this application. In this experiment, we learn the max margin classifiers using data acquired a particular set of cameras, and we use learned boundaries to classify data acquired from same actions using a different set of cameras. For this purpose we select publicly available dataset of IXMAS [183] as used for examining domain generalization performance in [156, 171]. IXMAS contains the videos acquired using five cameras installed at different viewpoints from 12 actors performing 11 different actions. Each action is performed 3 times by each actor. As suggested in [171] to exclude irregularly performed actions we keep five actions (check watch, cross arms, scratch head, sit down, get up) performed by first 6 actors (Alba, Andreas, Daniel, Hedlena, Julien and Nicolas). We extract dense trajectory features and use HOF, HOG, MHBx and MHBx descriptors in a 396-dimensional feature vector. We use these features to compute 2000 codewords using K-means algorithms to represent each action on each camera.

http://www1.icsi.berkeley.edu/ saenko/projects.html
<table>
<thead>
<tr>
<th>Source</th>
<th>Cam 0,1</th>
<th>Cam 2,3,4</th>
<th>Cam 0,1,2,3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target</td>
<td>Cam 2,3,4</td>
<td>Cam 0,1</td>
<td>Cam 4</td>
</tr>
<tr>
<td>SVM - No Ada</td>
<td>71.70</td>
<td>63.83</td>
<td>56.61</td>
</tr>
<tr>
<td>KMM</td>
<td>73.92</td>
<td>42.22</td>
<td>52.57</td>
</tr>
<tr>
<td>SGF</td>
<td>60.37</td>
<td>69.04</td>
<td>28.66</td>
</tr>
<tr>
<td>GFK</td>
<td>64.87</td>
<td>55.53</td>
<td>42.16</td>
</tr>
<tr>
<td>STM</td>
<td>68.69</td>
<td>70.53</td>
<td>51.05</td>
</tr>
<tr>
<td>DIP</td>
<td>65.20</td>
<td>70.03</td>
<td>62.92</td>
</tr>
<tr>
<td>SA</td>
<td>73.35</td>
<td>77.92</td>
<td>49.59</td>
</tr>
<tr>
<td>GFK(latent)[171]</td>
<td>69.12</td>
<td>68.87</td>
<td>51.30</td>
</tr>
<tr>
<td>SA(latent)[171]</td>
<td>71.04</td>
<td>76.64</td>
<td>72.26</td>
</tr>
<tr>
<td>DAM(latent)[171]</td>
<td>77.32</td>
<td>73.94</td>
<td>63.47</td>
</tr>
<tr>
<td>DAM(latent)[153]</td>
<td>77.92</td>
<td>76.99</td>
<td>53.76</td>
</tr>
<tr>
<td>LRE-SVMs-DA</td>
<td>81.79</td>
<td>82.43</td>
<td>75.26</td>
</tr>
<tr>
<td>Our method</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
</tbody>
</table>

Table 4.4: IXMAS Domain adaptation on IXMAS using dense trajectory features, comparison of recognition accuracy of unsupervised DA methods against our proposed method.

![Figure 4.3: AR dataset. Recognition rates versus different numbers \(p\) of corrupted training images per class for the three scenarios: sunglasses (left), scarf (center), sunglasses and scarf (right).](image)

**4.4.3 Face Recognition:**

Since we got promising results using our model on domain adaptation methods, this motivated us to challenge our method further on classification problems where training data are disguised with structured nuisance factors. We chose AR dataset (a benchmark dataset for face recognition) for learning our max margin decision boundary on face images which are captured under different occlusions, Fig 4.3 illustrates the results obtained on this dataset. For detailed description of the experiment please refer to (Section 3.7.2). As you can see for various number of occluded images in the training set this method is outperforming all matrix factorization methods. Extensive experiments using various features, proves the effectiveness of the proposed method and increases the accuracy of recognition on domain adaptation and generalization benchmark datasets.
Chapter 5

Sparse Coding and Dictionary Learning for Recognition

5.1 Introduction

It is well known that contextual information boosts object recognition performance [184, 185, 186, 187, 188, 189, 190]. (e.g. it is more likely to find a bottle over a table than over an airplane). Modeling contextual information is even more important when an image has a large number of objects. This is because context is better learned and leveraged [184, 191]. Also, context works more effectively when big objects are easily detected first (e.g. a table), while smaller ones (e.g. a bottle) are detected last [191, 192]. Careful modeling also leads to an improved detection of objects embedded in unconventional context [193] (e.g. a sofa on the street). Finally, there is a growing belief that context is very effective for extracting higher-level semantic information. For instance, modeling the interaction between objects leads to the interpretation of actions [184, 190] (e.g. a man playing tennis requires to detect a person and a tennis racket in a suitable relative position).

Sparse Representation primarily was introduced for face recognition in computer vision. [12] demonstrate that how face images lay in same subspace and a given test face image can be reconstructed using images of face samples in the dictionary from the same subject and ruling out noise factors like occlusion in the reconstructed image. Sparse coding or in other words linear decomposition or reconstruction of data points using elements from a dictionary has shown a great potential and during recent years has found to be useful in many applications such as signal processing and information theory [65], then it found broad applications in Denoising [49], restoration [50], super resolution [194], classification and face recognition [12], clustering [101]. Producing outstanding results has intrigued us to devote this work to build on sparse representation and improve existing detection and recognition methods based on sparse representation.
In this chapter, we propose to extend the K-SVD algorithm further to learn an over-complete dictionary from a set of labeled training images, so that spatial contextual information describing the presence of other objects becomes predictable by the dictionary (see Figure 5.1).

We present a supervised algorithm that for training starts from a dataset of images where objects and their positions (bounding boxes) are known, and incorporates these labels directly into the dictionary learning stage. The resulting objective function is optimized with the K-SVD algorithm. Therefore, the learned dictionary is still compact, reconstructive, and discriminative, and not just representational like traditional approaches [12, 50, 55, 56]. In particular, like [52, 57] our approach learns a simple multiclass linear classifier, in contrast to approaches that learn multiple binary classifiers [51, 53, 195, 196]. More importantly, a collection of linear regressors is also learned, such that the dictionary can map sparse codes to the space of layout distributions of objects. The dictionary, the classifier, and such layout maps are learned simultaneously, as opposed to approaches that require iterations between subsets of parameters or separate the learning of the dictionary from the rest [53, 59, 195, 196]. This is highly desirable, and given the efficiency of the K-SVD, it leads to solutions that scale well with the number of object categories, and tend to avoid local minima. In order to show how sparse representation can utilize linear reconstruction property of points from same subspace, we will focus here on sparse subspace clustering [101] to introduce the concept of sparse representation. The most straightforward assumption of the reconstruction problem in 1.13 is that we attempt to reconstruct data from same data points, in other words instead of learning a Dictionary, we use $X = D$, then Eq. 1.13 can be reformulated as:

$$
< a_i > = \arg\min_{\|a\|_0} \quad s.t. \quad x_i = Xa_i, \quad a_{ii} = 0
$$

(5.1)
Since we know data points from same subspace can be written as linear combination of data points from same subspace, therefor we expect coefficients of $a$ corresponding to data points from same subspace be non-zero, therefore by solving problem (5.1) one would be able to reveal the subspace structure of existing data points, in fact as we saw in previous section solution of problem (5.1) is NP-hard therefore its convex hull, which is $l_1 - norm$ instead of $l_0 - norm$:

$$< a_i > = \arg \min_{\|a\|_1} \quad s.t. \quad x_i = Xa_i, \quad a_{ii} = 0 \quad (5.2)$$

this problem can be solved efficiently using convex optimization methods, also if we repeat the problem (5.2) for all data point we will have

$$< A > = \arg \min_{\|A\|_1} \quad s.t. \quad X = XA, \quad diag(A) = 0 \quad (5.3)$$

[101] has shown that by converting $C$ matrix into symmetric and non-negative affinities $|A| + |A^T|$, from which the segmentation of subspaces is found using spectral clustering.

In the case of data contaminated by gross noise $E$, and also gaussian noise $G$, It is shown that they can be model in optimization problem as in following:

$$< A > = \arg \min_{A,G,E} \|A\|_1 + \alpha/2\|G\|_2^2 + \gamma\|E\|_1 \quad s.t. \quad X = XA + E + G, \quad diag(A) = 0 \quad (5.4)$$

In this chapter, we will discuss on fundamentals of sparse representation and provide useful information for those readers that are not familiar with sparse representation. Then we will discuss about different dictionary learning methods used for sparse coding, and also we will provide some insight on how sparse representation was transformed from reconstruction to other applications in computer vision. Section 5.3 introduces our context model and structured prediction framework using sparse representation, and Section 5.4 validates the proposed approach.

### 5.2 Sparse Coding and Dictionary Learning

In this chapter for sparse representation, we will use KSVD which is discussed in Sec. 5.3, We will consider the most corresponding optimization problem Equation. 5.5 and will illustrate how it can be generalized to regression problems. Problem (1.18) can be solved efficiently by the K-SVD algorithm [56]. Once the dictionary $D$ is available, the sparse representation $A$ of the signals $X$ can be computed by applying a matching pursuit method, for instance, the orthonormal matching
pursuit (OMG) [197], to solve the following problem

\[ \langle A \rangle = \arg \min_A \|X - DA\|_2^2 \quad \text{s.t.} \quad \|A\|_0 < T. \]  

(5.5)

### 5.2.1 Adding Discriminative Power

The dictionary learned through (1.18) has excellent reconstructive properties. When used for a recognition task it should also have discriminative properties. One way to “endow” the dictionary with those is to use the sparse codes \( A \) as features for classification. This requires training a classifier \( f(a, G) \), where its parameters \( G \) satisfy

\[ \langle G \rangle = \arg \min_G \sum_i L\{y_i, f(a_i, G)\} + \lambda\|G\|_2^2, \]  

(5.6)

and where \( y_i \) is the label associated with the feature \( a_i \), \( L \) is the classification loss function (for which there are several choices [52, 54, 195, 196]), and \( \lambda \) is a parameter that sets the strength of the regularizing term.

The disconnection between dictionary and classifier learning means that the dictionary may originate only suboptimal features for classification. This issue can be significantly mitigated if the learning of the classifier and the dictionary happen simultaneously, by combining the cost functions in (1.18) and (5.6), leading to

\[ \langle D, A, G \rangle = \arg \min_{D,A,G} \|X - DA\|_2^2 \]

\[ + \sum_i L\{y_i, f(a_i, G)\} + \lambda\|G\|_2^2 \quad \text{s.t.} \quad \forall i, \|a_i\|_0 \leq T. \]  

(5.7)

Several approaches use a similar architecture [52, 54, 57, 195, 196]. Most of these algorithms design fairly elaborate procedures to solve (5.7), where subproblems compute temporary estimations of a subset of parameters, increasing the possibility for the optimization not to converge to the right solution.

A substantial difference in approaching problem (5.7) has been introduced by [52]. In particular, they advocate the use of a simple quadratic cost function and a simple linear classifier. The outcome is the possibility to solve (5.7) very elegantly with the K-SVD algorithm [56], inheriting all the computational benefits it comes with.

One concern, not explicitly addressed in [56], is the fact that in object recognition, given the large intra-class and inter-class variation of the observations, the size of the training dataset may significantly increase the number of classes. The result is that maintaining the same discriminative
power as the number of classes grows, forces the dictionary size to increase, which is undesirable. The work in [57] addresses this problem by building on the approach in [56]. They can learn a dictionary, simultaneously with a multi-class linear classifier, that has a size that scales well with the number of classes, while maintaining excellent performance. This is achieved just by adding a clever regularizing term to the cost function to optimize, which is linear in the sparse codes $A$. In addition to that, similarly to [56], [57] solves (5.7) with the K-SVD algorithm, inheriting again all the benefits that come with it.

5.3 Context Model

In this section, we are interested in extending the framework introduced in Section 5.2, and learn a dictionary that is “endowed” with the ability to predict the context of a given signal $x_i$, where $x_i$ might represent an object or a region in an image.

5.3.1 Adding Context Layout information

So far we made the implicit assumption that a signal $x_i$ can be represented by a sparse code $a_i$, according to an over-complete dictionary $D$. We also assumed that the same signal is a sample from a class identified by the label $y_i \in \{1, \cdots, c\}$. If $x_i$ represents an object $i$ of class $y_i$ in an image, predicting its context means being able to provide the likelihood information of whether another object, of a given class $j$, is located in a certain position $p$, with respect to object $i$. We indicate this likelihood with $l_{pi}(j)$. Also, with $l_i(j) \doteq [l_{i1}(j), \cdots, l_{mi}(j)]'$ we indicate the set of likelihoods describing the full spatial layout distribution of where an object of class $j$ might be located around object $i$.

Following the reasoning in Section 5.2, one way to exploit the dictionary to predict the spatial layout distribution of a class $j$ around an object, is to consider the sparse codes $A$ as features for a regression model $g(a, W(j))$, where its parameters $W(j)$ satisfy the following

$$< W(j) >= \arg \min_{W(j)} \sum_i M\{ l_i(j), g(a_i, W(j)) \} + \nu \| W(j) \|_2^2. \quad (5.8)$$

Here $\nu$ is a scalar that sets the strengths of the regularizing term, whereas $M$ is a regression loss function of our choice.

If we were to proceed along the path just highlighted, we would create a disconnection between the dictionary and the regression learning in that the dictionary would originate only suboptimal features for regression. Again, this issue can be mitigated if the learning of the regression and
of the dictionary happen simultaneously by combining the cost functions in (1.18) and in (5.8). However, since we are interested in learning also a discriminative dictionary, we combine the cost functions with the one in (5.6) as well. This leads to the following training problem

\[
<D, A, G, W(\cdot)> = \arg \min_{D, G, A, W(\cdot)} \|X - DA\|_2^2 + \sum_i \mathcal{L}\{y_i, f(a_i, G)\} + \lambda\|G\|_2^2 \\
+ \sum_j \sum_i \mathcal{M}\{l_i(j), g(a_i, W(j))\} + \nu \sum_j \|W(j)\|_2^2 \quad \text{s.t.} \forall i, \|a_i\|_0 \leq T.
\] (5.9)

For estimating the parameters, we are interested in leveraging the lesson learned from [52, 57] and convert (5.9) into a problem approachable with the K-SVD. We proceed by picking quadratic loss functions for \(\mathcal{L}\) and \(\mathcal{M}\), and we chose a linear multi-class classifier parameterized by \(G \in \mathbb{R}^{c \times k}\), i.e. \(f(a, G) = Ga\). Also, we select a linear regression model parameterized by \(W(j) \in \mathbb{R}^{m \times k}\), i.e. \(g(a, W(j)) = W(j)a\). In addition, we define the following variables: \(Y \doteq [y_1, \cdots, y_N] \in \mathbb{R}^{c \times N}\), and \(L(j) \doteq [l_1(j), \cdots, l_N(j)] \in \mathbb{R}^{m \times N}\), where a label \(y_i = b\) now is represented as a vector with \(c-1\) zeros and a 1 in its \(b\)-th component, i.e. \(y_i = [0, \cdots, 0, 1, 0, \cdots 0]'\). Finally, we define the following matrices

\[
L = \begin{pmatrix}
L(1) \\
L(2) \\
\vdots \\
L(c)
\end{pmatrix}, \quad W = \begin{pmatrix}
W(1) \\
W(2) \\
\vdots \\
W(c)
\end{pmatrix}.
\] (5.10)

Given the choices outlined above, problem (5.9) can be expressed as follows

\[
<D, A, G, W> = \arg \min_{D, G, A, W} \|X - DA\|_2^2 + \alpha\|Y - GA\|_2^2 + \chi\|G\|_2^2 \\
+ \beta\|L - WA\|_2^2 + \nu\|W\|_2^2 \quad \text{s.t.} \quad \|A\|_0 \leq T
\]

where \(\alpha\) and \(\beta\) are tuning parameters setting the strength of the classification error versus the regression error versus the reconstruction error. Note that this formulation of the learning problem does not include the contribution proposed by [57]. This is done to keep the exposition of this work more focused on the newly proposed contribution. However, the reader should be warned that the consistent label term proposed in [57], could be added to the cost function in (5.12) as well, producing the beneficial effects discussed in Section 5.2. In the following we will explain how to address (5.12) with the K-SVD.

94
5.3.2 Learning with K-SVD

As a first step, we notice that the cost function in (5.12) can be rewritten as

\[ <D, A, G, W> = \arg \min_{D, G, W, A} \left\| \begin{pmatrix} X \sqrt{\alpha Y} \\ \sqrt{\beta L} \end{pmatrix} - \begin{pmatrix} D \sqrt{\alpha G} \\ \sqrt{\beta W} \end{pmatrix} A \right\|_2^2 + \lambda \|G\|^2_2 + \nu \|W\|^2_2 \text{ s.t. } \|A\|_0 \leq T, \tag{5.12} \]

and a more compact expression is obtained by setting \( X = (X', \sqrt{\alpha}Y', \sqrt{\beta}L') \) and \( D = (D', \sqrt{\alpha}G', \sqrt{\beta}W') \), leading to

\[ <D, A> = \arg \min_{D, A} \|X - DA\|_2^2 + \lambda \|G\|^2_2 + \nu \|W\|^2_2 \text{ s.t. } \|A\|_0 \leq T. \tag{5.13} \]

Finally, since the K-SVD proceeds by iteratively updating each column of \( D \), while maintaining them normalized, the regularizing terms on \( \|G\|_2 \) and \( \|W\|_2 \) are dropped, or equivalently \( \lambda \) and \( \nu \) can be set to 0. To summarize the optimization algorithm we indicate with \( d_k \) and \( a_k \) the \( k \)-th column and row of \( D \) and \( A \), respectively, and define \( E_k = (X - \sum_{j \neq k} d_j a_j) \). Also, \( \tilde{a}^j \) indicates the row vector \( a^j \) after removing the zero entries, and \( \tilde{E}_k \) is the error \( E_k \) after removing the columns in corresponding positions. Following the K-SVD protocol, \( d_k \) and \( \tilde{a}^k \) are updated by solving

\[ <d_k, \tilde{a}^k> = \arg \min_{d_k, \tilde{a}^k} \|\tilde{E}_k - d_k \tilde{a}^k\|_2^2, \tag{5.14} \]

which can be computed in closed form by decomposing the error \( \tilde{E}_k \overset{SVD}{=} U \Sigma V' \), and setting (using Matlab notation) \( d_k = U(:, 1) \), and \( \tilde{a}^k = \Sigma(1, 1) V(:, 1)' \). \( \tilde{a}^k \) is then used to replace the non-zero values in \( a^k \). After a complete update of \( D \), the sparse code matrix \( A \) is updated with a matching pursuit algorithm, and the iteration repeats until convergence. This process maintains the columns of \( D \) normalized. Therefore, similarly to what is done in [52], the columns of \( D \) have to be re-normalized, and the columns of \( G \) and \( W \) have to be re-scaled accordingly.

The iteration requires initializing the values of \( D, G, \) and \( W \). The dictionary can be obtained by just applying the traditional K-SVD algorithm to the input data \( X \), which means solving for problem (1.18). The classifier \( G \) can then be obtained by solving problem (5.6) with a quadratic loss function, and the regression maps \( W \) are computed through (5.8), again with a quadratic loss.
Figure 5.2: **Layout distributions.** A given $5 \times 5$ elements section of the left image shows, for a given class on the horizontal axis, the spatial layout probability to find a given class on the vertical axis. Each $5 \times 5$ elements image is color coded according to the layout distribution. Lighter means higher probability. (a), (b), and (c) are close ups of the layout distributions. It should be noted that to visualize the learned spatial layout better for a particular class with respect to the others, we normalize distribution probability of a row in layout map to span between zero and one. See Section 5.4 for more information.

function. The solution to (5.6) and (5.8) is given in closed form by [198], and more precisely by

$$
G = (AA' + \lambda I)^{-1}AY',
$$

$$
W = (AA' + \nu I)^{-1}AL'.
$$

(5.15)

This learning approach allows to simultaneously estimate all the parameters with a procedure that tends to avoid local minima and is numerically efficient. Through the use of robust and sequential algorithms for computing the SVD [199] it is possible to scale well with the size of the training dataset and with the number of classes to handle, even if the number of parameters to estimate is large. At the same time, it is possible to impose the desired size of the dictionary while maintaining it reconstructive, discriminative, and context aware. As mentioned before, by adding the contribution proposed in [57], it is possible to obtain even more compact dictionaries without the expense of losing discriminative power, which further increases scalability with respect to the number of classes.
5.3.3 Simultaneous Object Localization and Recognition

In this section, we are interested in using the context-aware dictionary introduced in Section 5.3.1 to design a structured prediction framework able to localize and recognize objects in images simultaneously.

All the notation developed so far is still valid, what changes here is that we consider an image $I$ depicting $M$ objects, each of which is represented by a feature vector $x$, so $X = [x_1, \cdots, x_M]$ is now the set of features, or signals describing the objects in the image $I$. Assuming that a dictionary $D$ has already been learned, the sparse codes $A = [a_1, \cdots, a_M]$ representing the objects can be computed by (5.5). The feature $x$ could be a simple histogram of oriented gradients (HOG) descriptor [200], which is indeed what was used in our experiments. The descriptors are computed over the $M$ bounding boxes surrounding the hypothetical $M$ objects. The boxes could be provided by a bank of sliding window detectors[186], or a selective search procedure [201].

For a given object $i$ in image $I$, $l_i(j)$ describes the layout distribution of objects of class $j \in \{1, \cdots, c\}$, around object $i$. Figure 5.3 shows how the space around object $i$ is divided in $m = 11$ regions. The inner circle has a diameter corresponding to the diagonal of the bounding box of the object, and the outer circle has a diameter that is three times the same diagonal. Region 10 is delimited by the size of the image, while region 11 is anywhere outside of the image. When the center location of an object of class $j$ falls into a given region, its presence is recorded into the corresponding bin of $l_i(j)$.

Given the framework above, the dictionary $D$, the classifier $G$, and the layout maps $W$, we want to design a score function $S(Y, A)$ that, given the sparse codes $A$, when optimized with respect to $Y = [y_1, \cdots, y_M]$ produces the correct labels for the set of $M$ boxes in the image. One way to proceed is to make the following two observations about object $i$, represented by the sparse code
a; (a) the label \( y_i \) should be predictable by the classifier \( G a_i \); (b) the layout distribution around object \( i \) of objects of class \( j \), \( l_i(j) \), should be predictable by \( W(j)a_i \). This should hold for every object in the image. Therefore, the score function that we are looking for could be expressed as

\[
S(Y, A) = \sum_i \|y_i - G a_i\|^2_2 + \gamma \sum_i \sum_j \|l_i(j) - W(j)a_i\|^2_2 ,
\]

(5.16)

Where the first term minimizes the classifier error, and the second term the layout map error. The coefficient \( \gamma \) is a parameter to be learned from data and establishes a balance between the terms.

While (5.16) is certainly a good option, we design a simplified model that will provide considerable computational advantages. In particular, we change the way we treat contextual information around object \( i \), and we say that an object \( j \) should be able to predict the layout \( l_j(y_i) \) of the objects of class \( y_i \) around object \( j \), which is given by \( W(y_i)a_j \). This should hold for every object. Therefore, the score function associated with this model is given by

\[
S(Y, A) = \sum_i \|y_i - G a_i\|^2_2 + \frac{\gamma}{M} \sum_i \sum_j \|l_j(y_i) - W(y_i)a_j\|^2_2 ,
\]

(5.17)

where, again, \( \gamma \) is a parameter to be learned from training data, and \( M \) here is necessary for normalization purposes.

A fundamental advantage of model (5.17) versus model (5.16) is the fact that the score function can be written as

\[
S(Y, A) = \sum_i S(y_i, A) ,
\]

(5.18)

where

\[
S(y_i, A) = \|y_i - G a_i\|^2_2 + \frac{\gamma}{M} \|L(y_i) - W(y_i)A\|^2_2 ,
\]

(5.19)

and every label \( y_i \) can be determined individually, and very efficiently with a simple linear search, by optimizing

\[
< y_i >= \arg \min_{y_i} S(y_i, A) \quad \text{for } i = 1, \ldots, M .
\]

(5.20)

Estimating one label at a time is a considerable advantage with respect to model (5.16). In fact, optimizing (5.16) would require a full search over the joint space of the labels of all objects. In the alternative, one could start from an initial guess provided by the classifier, and then iteratively update each label one at a time, with all the risks of converging only to a local minimum.
5.3.4 Balancing the Context

Let us assume that a set of \( P \) training images \( \{ I^p \}_{p=1}^{\ldots P} \) is given, and that each image \( I^p \) has \( M^p \) boxes surrounding the same number of hypothetical objects. Then, during testing, the label \( y^p_i \) of the \( i \)-th box is computed as the argument that maximizes Equation (5.19) or in a more elaborate form the following score function

\[
S(y^p_i, A^p) = \|y^p_i - Ga^p_i\|^2 + \frac{\gamma}{M^p} \|L^p(y^p_i) - W(y^p_i)A^p\|^2.
\] (5.21)

Where \( A^p = [a^p_1, \ldots, a^p_{M^p}] \in \mathbb{R}^{k \times M^p} \) is the sparse representation of the boxes in the training image \( I^p \), and \( L^p(y^p_i) \in \mathbb{R}^{m \times M^p} \) is the spatial layout of the objects of class \( y^p_i \). The score in Equation (5.21) is composed of two terms with relative weight defined by the parameter \( \gamma \).

The weight \( \gamma \) should be such that the importance of the context versus the data fidelity term guarantees that the label assignment is the best possible. This means that a deviation from the correct assignment should correspond to an increase in the score because only the right assignment produces a minimum. This means that the derivative of a score, computed in correspondence of the correct label assignment should be zero or, in the presence of noise, as close to zero as possible. Therefore, given the set of training images \( \{ I^p \} \) and corresponding label annotations \( \{ y^p_i \} \), we estimate \( \gamma \) by solving the following least squares problem:

\[
\hat{\gamma} = \arg \min_{\gamma} \sum_{p=1}^P \sum_{i=1}^{M^p} \| \frac{\partial S(y^p_i, A^p)}{\partial y^p_i} \|^2_2.
\] (5.22)

Problem (5.22) entails the computation of the following derivative (where in order to lighten the notation we drop the superscript \( p \), indicating the particular training image)

\[
\frac{\partial S(y_i, A)}{\partial y_i} = 2(y_i - Ga_i)^\top + \frac{2\gamma}{M} \sum_{j=1}^M (l_j(y_i) - W(y_i)a_j)^\top \left( \frac{\partial l_j(y_i)}{\partial y_i} - \frac{\partial (W(y_i)a_j)}{\partial y_i} \right)
\] (5.23)

where \( l_j(y_i) \in \mathbb{R}^{m \times 1}, \ j = 1, \ldots, M, \) indicates the spatial layout of the objects/boxes of class \( y_i \) around object/box \( j \). The right hand side of Equation (5.23) can be written as \( B_i + \gamma C_i \), where

\[
B_i = 2(y_i - Ga_i)^\top,
\] (5.24)

\[
C_i = \frac{2}{M} \sum_{j=1}^M (l_j(y_i) - W(y_i)a_j)^\top \left( \frac{\partial l_j(y_i)}{\partial y_i} - \frac{\partial (W(y_i)a_j)}{\partial y_i} \right).
\] (5.25)
In order to compute $C_i$ we express $W(y_i) \in \mathbb{R}^{m \times k}$ and $l_j(y_i)$ as follows

$$W(y_i) = (y_i^\top \otimes I_m)W,$$
$$l_j(y_i) = (y_i^\top \otimes I_m)L_j,$$

where $L_j = [l_j(1)^\top, \ldots, l_j(c)^\top]^\top \in \mathbb{R}^{mc \times 1}$ represents the complete layout of object/box $j$, and $W = [W(1)^\top, \ldots, W(c)^\top]^\top \in \mathbb{R}^{mc \times k}$ is the concatenation of the mapping functions for all the possible classes. With this notation, and by relaxing the variable $y_i$ (note that its components are meant to assume only binary values), the computation of $C_i$ proceeds as follows

$$\frac{\partial(W(y_i)a_j)}{\partial y_i} = (a_j^\top \otimes I_m)\frac{\partial W(y_i)}{\partial y_i}, \quad (5.26)$$

where

$$\frac{\partial W(y_i)}{\partial y_i} = \frac{\partial((y_i^\top \otimes I_m)W)}{\partial y_i} = (W^\top \otimes I_m)\frac{\partial(y_i^\top \otimes I_m)}{\partial y_i}, \quad (5.27)$$

and finally

$$\frac{\partial(W(y_i)a_j)}{\partial y_i} = (a_j^\top \otimes I_m)(W^\top \otimes I_m)(I_c \otimes \text{vec}(I_m)). \quad (5.28)$$

In a similar manner we have

$$\frac{\partial l_j(y_i)}{\partial y_i} = \frac{\partial((y_i^\top \otimes I_m)L_j)}{\partial y_i} = (L_j^\top \otimes I_m)\frac{\partial(y_i^\top \otimes I_m)}{\partial y_i}, \quad (5.29)$$

and the final expression becomes

$$\frac{\partial l_j(y_i)}{\partial y_i} = (L_j^\top \otimes I_m)(I_c \otimes \text{vec}(I_m)), \quad (5.30)$$

Finally, the quantities $B_i \in \mathbb{R}^{1 \times c}$ and $C_i \in \mathbb{R}^{1 \times c}$ are expressed as below:

$$B_i = 2(y_i - Ga_i)^\top, \quad (5.31)$$

$$C_i = \frac{2}{M} \sum_{j=1}^{M} (l_j(y_i) - W(y_i)a_j)^\top \left[(L_j^\top \otimes I_m)(I_c \otimes \text{vec}(I_m)) - (a_j^\top \otimes I_m)(W^\top \otimes I_m)(I_c \otimes \text{vec}(I_m))\right]. \quad (5.32)$$

At this point if we concatenate all $B_i^p$ and $C_i^p$, obtained from $p = \{1, \ldots, P\}$ training images,
into two row vectors $B = [B_1, \ldots, B_p]$, and $C = [C_1, \ldots, C_p]$, then problem (5.22) can be rewritten as follows

$$\hat{\gamma} = \arg \min_{\gamma} \|B + \gamma C\|^2_2,$$  

(5.33)

for which the least squares solution becomes

$$\hat{\gamma} = -\frac{BC^\top}{CC^\top}.$$  

(5.34)

Then using estimated optimum $\hat{\gamma}$ we can rewrite score function as in below:

$$S(y_i, A) = \|y_i - Ga_i\|^2_2 + \frac{\hat{\gamma}_j}{M} \|L(y_i) - W(y_i)A\|^2_2,$$  

(5.35)

This regularizing term assures us that we are working at the optimum point of class fidelity and context fidelity amalgam. To keep the flow of the report consistent, we leave the evaluation of the obtained context model for the experiment section.

### 5.4 Experiments

We evaluate the proposed method on two most challenging datasets of Pascal Visual Object Challenge (VOC2007) [60] and SUN09 [193]. The VOC2007 consist of about 10,000 images of 20 different classes. About half of them are used for training and validation and the other half is used for testing, in average each image in data set contains about 2.5 objects. SUN09 is known to be a suitable data set for exploiting contextual information. It contains 12,000 annotated images covering a large number of indoor and outdoor scenes, for more information on this data set reader can refer to [193].

Fig. 5.4 (a) shows the existential concurrency and respective layout of the categories in training set of VOC2007 as a form of probability distribution depicted in grayscale normalized among all categories, Fig. 5.4 (b) is the learned layout maps $W(j)a_i, i, j = \{1, \ldots, M\}$ by our model and Fig. 5.4 (c) is the difference between actual layout map and learned layout map. It can be easily inferred from difference of two maps that our model captures the concurrency and layout of different object categories very well.

Fig. 5.2 illustrates the results from learning the layout maps $W(j)$ on 20 classes of VOC2007. In particular, given an object represented by the sparse code $a_i$, the quantity $W(j)a_i$ represents the layout likelihood to find an object of class $j$ around the object $i$. In Fig. 5.2 the vertical axis represents the object class $j$, whereas the horizontal axis represents the class of the object $i$. Therefore, in the table showing the layout maps, the row corresponding to the class $j$ shows an example
of likelihood distribution $W(j)a_i$, for a sparse code $a_i$ that comes from each of the 20 classes of the dataset. Each of the entry $W(j)a_i$ in the $20 \times 20$ table is represented by a $5 \times 5$ elements image, where each pixel element is color coded with the likelihood of the corresponding layout bin. It should be noted that in Fig. 5.2 for all the bins corresponding to object $i$ which consist of $W(j)a_i$, $j = \{1, 2, ..., M\}$, all the bins are normalized to 0-1 with the same scaling map to produce a better gray scale illustration. Lighter areas mean higher likelihood. Fig. 5.2-(a)(b)(c) shows a close-up view of a few layout distributions. In particular, Fig. 5.2-(b)(c) highlights that in the dataset there certainly are enough examples showing bottles on tables, and people riding horses, as it appears from the layout distributions. Similar interpretations hold for the other examples. More importantly, this is a clear indication that the proposed approach can capture these types of contextual constraints. In addition, Fig. 5.2-(b)(c) shows that inverting the order of the two classes considered (i.e. switching from $W(j)a_i$ to $W(i)a_j$) leads to a flipping of the distribution with respect to the center. This is expected (e.g. if bottles are likely to sit on tables, then tables are likely to hold bottles, and vice-versa), and it gives another confirmation that the model is capturing the correct information.

We cast rest of our experimental section in two folds, primarily we evaluate our algorithm on a synthetical framework on VOC2007, and in the second part, we examine our algorithm on a real application for re-scoring detections on VOC2007 and SUN09 based on contextual information. To test the proposed method, we cre-
ate a supervised framework, which will prove the efficiency of embedding contextual information in sparse feature construction and classification. For the supervised framework, we use Pascal VOC2007 dataset where we use training data for learning dictionary, and for testing, we use test images along with their annotated bounding framed but by discarding their labels, which will create a supervised framework, providing us initial bounding frames of real objects in their existential context layout. Now we will deploy our algorithm to assign category labels to anonymous boxes. Using this framework enables us to compare the accuracy of classification using dictionary learned without context coding, against the dictionary learned with context layout. Non-context dictionary learning is where our dictionary is obtained based on local features and label reconstruction error minimization, and context-aware dictionary learning is the one which In addition to retaining previously mentioned properties it embeds contextual layout as well. Tab. 5.1 shows the average precision obtained from non-context dictionary learning and context aware dictionary learning for each class. Also, Fig. 5.5 illustrate precision-recall curves obtained from running two methods. In fig. 5.5 the precision-recall (PR) curves are obtained following the comp3 protocol of the VOC Challenge. Here we report only the results of four classes, namely bottle, car, chair, and train, in which both Tab. 5.1 and Fig. 5.5 indicates our context aware coding is a legitimate method to incorporate contextual information in object detection and increases the accuracy. The second set of experiments are on real detection confidence rescoreing application and is applied for rescoring detections based on contextual information. In this section of experiments, we used the discriminative part-based models (DPBM) described in [186] as baseline local object detector, which is known to produce reliable detections for generic object categories. Given initial candidate windows and confidence scores, we exploit our context model to rescore the detections and increase the accuracy of detections. Also, we compare our context rescoreing algorithm with the rescoreing methods proposed in [186] and [193]. For both these methods, we use DPBM to produce the initial detections. Context model proposed in [186] denoted as SVM-Context, trains an SVM for each of M object categories to incorporate contextual information, for a given candidate window a feature vector of size (M+5) is formed; which consists the score of most confident detections from each object category, pulse the coordinate information of the candidate window and its own confidence score. In [193] which we denote as tree-based model, a binary tree is learned to capture the concurrency of different object categories and a gaussian variable is linked to learned concurrency tree to model.
the relative spatial location of the objects in the image and also a gist features is used to enrich the model with the gist of image.

5.4.1 Performance on VOC2007

The training procedure illustrated in section 5.3 is affected by the Parameters $\alpha$ and $\beta$, we conducted a grid search for finding best values. In This section instead of using structured predictive model we compute the three quadratic reconstruction loss functions and using an SVM along with the detectors confidence score, we find the best ratio for these loss functions and initial confidence score, and then using a linear combination of them, we rescore the confidence of detection for each detection. Tab. 5.2 shows the corresponding per class average precision (AP) for all the 20 classes. The table includes the AP of the off the shelf baseline algorithm for object detection without using any context model [186], the AP precision obtained using SVM context model proposed in [186], and AP for tree based model proposed in [193]. As it can be seen, the context aware dictionary produces results in line with state of the art methods. Fig. 5.6 shows the improvement in average precision for each object category sorted by the AP improvement over the baseline. Due to the large number of objects in our dataset, many objects benefit in different degrees from context. Fig. 5.7 shows samples from the testing dataset in first column, the second column illustrate the
Table 5.2: Per-class AP scores on the PASCAL VOC 2007 dataset [202]. Baseline is the AP obtained using baseline detector without any context model [186] (Voc-release5 available online has been used to produce baseline results), the results are obtained after Bounding Box Prediction and Non-Maximum Suppression. Also, we confine the maximum number of detected bounding boxes for each category detector to (3 in the table in the right hand and 10 in the table in left hand side) most confident detections. SVM depicts the results obtained by context SVM model proposed by [186]. Tree-based are the AP obtained from Hierarchical Context model in [193]. sparse is the results for proposed method using compact KSVD dictionary learning and the boundry is the maximal bound for AP which could be obtained given initial detections using baseline detector for the current max recall.

<table>
<thead>
<tr>
<th>Class</th>
<th>baseline</th>
<th>SVM</th>
<th>Tree-based</th>
<th>sparse</th>
<th>bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>plane</td>
<td>32.81</td>
<td>35.95</td>
<td>0.3587</td>
<td>0.3523</td>
<td>56.14</td>
</tr>
<tr>
<td>bike</td>
<td>57.35</td>
<td>60.43</td>
<td>0.5246</td>
<td>0.5992</td>
<td>71.81</td>
</tr>
<tr>
<td>bird</td>
<td>10.13</td>
<td>11.88</td>
<td>0.1204</td>
<td>0.1197</td>
<td>30.28</td>
</tr>
<tr>
<td>boat</td>
<td>15.52</td>
<td>17.74</td>
<td>0.1840</td>
<td>0.1935</td>
<td>34.60</td>
</tr>
<tr>
<td>bottle</td>
<td>23.50</td>
<td>25.04</td>
<td>0.2337</td>
<td>0.2520</td>
<td>33.05</td>
</tr>
<tr>
<td>bus</td>
<td>51.31</td>
<td>54.42</td>
<td>0.4815</td>
<td>0.5419</td>
<td>76.53</td>
</tr>
<tr>
<td>car</td>
<td>50.70</td>
<td>57.29</td>
<td>0.5003</td>
<td>0.5694</td>
<td>63.45</td>
</tr>
<tr>
<td>cat</td>
<td>21.91</td>
<td>25.07</td>
<td>0.2413</td>
<td>0.2434</td>
<td>54.75</td>
</tr>
<tr>
<td>chair</td>
<td>19.11</td>
<td>20.17</td>
<td>0.2150</td>
<td>0.2087</td>
<td>38.89</td>
</tr>
<tr>
<td>cow</td>
<td>23.87</td>
<td>25.19</td>
<td>0.2090</td>
<td>0.2629</td>
<td>48.36</td>
</tr>
<tr>
<td>table</td>
<td>26.73</td>
<td>26.56</td>
<td>0.2835</td>
<td>0.2637</td>
<td>64.08</td>
</tr>
<tr>
<td>dog</td>
<td>12.13</td>
<td>13.94</td>
<td>0.1370</td>
<td>0.1359</td>
<td>45.40</td>
</tr>
<tr>
<td>horse</td>
<td>56.30</td>
<td>61.10</td>
<td>0.5681</td>
<td>0.5932</td>
<td>76.72</td>
</tr>
<tr>
<td>motbike</td>
<td>47.32</td>
<td>49.92</td>
<td>0.4635</td>
<td>0.4941</td>
<td>66.77</td>
</tr>
<tr>
<td>person</td>
<td>38.86</td>
<td>40.07</td>
<td>0.3873</td>
<td>0.3953</td>
<td>50.11</td>
</tr>
<tr>
<td>plant</td>
<td>12.51</td>
<td>12.75</td>
<td>0.1392</td>
<td>0.1348</td>
<td>28.96</td>
</tr>
<tr>
<td>sheep</td>
<td>20.94</td>
<td>20.74</td>
<td>0.2333</td>
<td>0.2403</td>
<td>34.71</td>
</tr>
<tr>
<td>sofa</td>
<td>35.54</td>
<td>37.69</td>
<td>0.3489</td>
<td>0.3793</td>
<td>71.97</td>
</tr>
<tr>
<td>train</td>
<td>45.04</td>
<td>49.15</td>
<td>0.4612</td>
<td>0.4772</td>
<td>67.73</td>
</tr>
<tr>
<td>TV</td>
<td>41.59</td>
<td>42.67</td>
<td>0.4215</td>
<td>0.4289</td>
<td>68.51</td>
</tr>
<tr>
<td>Avg</td>
<td>32.16</td>
<td>34.38</td>
<td>32.56</td>
<td>34.43</td>
<td>54.14</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Class</th>
<th>baseline</th>
<th>SVM</th>
<th>Tree-based</th>
<th>sparse</th>
<th>bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>plane</td>
<td>33.04</td>
<td>36.58</td>
<td>35.66</td>
<td>35.34</td>
<td>62.11</td>
</tr>
<tr>
<td>bike</td>
<td>59.32</td>
<td>62.20</td>
<td>53.95</td>
<td>61.59</td>
<td>78.64</td>
</tr>
<tr>
<td>bird</td>
<td>10.23</td>
<td>12.09</td>
<td>11.62</td>
<td>11.70</td>
<td>42.92</td>
</tr>
<tr>
<td>boat</td>
<td>15.64</td>
<td>17.53</td>
<td>18.27</td>
<td>17.39</td>
<td>48.67</td>
</tr>
<tr>
<td>bottle</td>
<td>26.39</td>
<td>28.61</td>
<td>26.74</td>
<td>27.73</td>
<td>46.27</td>
</tr>
<tr>
<td>bus</td>
<td>52.00</td>
<td>54.63</td>
<td>48.51</td>
<td>54.28</td>
<td>79.81</td>
</tr>
<tr>
<td>car</td>
<td>53.69</td>
<td>60.41</td>
<td>52.48</td>
<td>58.06</td>
<td>73.77</td>
</tr>
<tr>
<td>cat</td>
<td>22.44</td>
<td>25.52</td>
<td>24.89</td>
<td>24.75</td>
<td>73.18</td>
</tr>
<tr>
<td>chair</td>
<td>20.15</td>
<td>21.21</td>
<td>22.37</td>
<td>21.20</td>
<td>55.42</td>
</tr>
<tr>
<td>cow</td>
<td>24.27</td>
<td>25.21</td>
<td>21.10</td>
<td>25.48</td>
<td>59.43</td>
</tr>
<tr>
<td>table</td>
<td>26.92</td>
<td>26.58</td>
<td>28.00</td>
<td>27.24</td>
<td>72.33</td>
</tr>
<tr>
<td>dog</td>
<td>12.56</td>
<td>14.70</td>
<td>14.16</td>
<td>14.06</td>
<td>67.08</td>
</tr>
<tr>
<td>horse</td>
<td>56.46</td>
<td>60.87</td>
<td>56.16</td>
<td>60.18</td>
<td>79.89</td>
</tr>
<tr>
<td>motbike</td>
<td>48.47</td>
<td>50.67</td>
<td>46.92</td>
<td>50.45</td>
<td>71.38</td>
</tr>
<tr>
<td>person</td>
<td>43.13</td>
<td>44.64</td>
<td>42.52</td>
<td>44.27</td>
<td>62.94</td>
</tr>
<tr>
<td>plant</td>
<td>13.43</td>
<td>14.38</td>
<td>15.19</td>
<td>13.99</td>
<td>43.54</td>
</tr>
<tr>
<td>sheep</td>
<td>21.24</td>
<td>21.36</td>
<td>24.72</td>
<td>21.12</td>
<td>46.69</td>
</tr>
<tr>
<td>sofa</td>
<td>35.88</td>
<td>37.83</td>
<td>34.79</td>
<td>37.59</td>
<td>85.36</td>
</tr>
<tr>
<td>train</td>
<td>45.17</td>
<td>49.28</td>
<td>46.21</td>
<td>48.05</td>
<td>73.05</td>
</tr>
<tr>
<td>TV</td>
<td>42.14</td>
<td>43.65</td>
<td>42.20</td>
<td>43.21</td>
<td>76.95</td>
</tr>
<tr>
<td>Avg</td>
<td>33.13</td>
<td>35.39</td>
<td>33.32</td>
<td>34.88</td>
<td>64.97</td>
</tr>
</tbody>
</table>

Results of DPM detector with 3 most confident detections from each category overlayed bounding boxes with confidence score written above each box. The third column illustrates only the 11 most confident detections across all classes. With their modified scores using SVM based method and the last column shows the results from sparse model rescoring scheme, visual comparison of modified scores after deploying our context model indicates that using our contextual coding is better able to exploit contextual layout information in compare to [186].

5.4.2 Performance on SUN09

SUN09 is known to be a suitable data set for exploiting contextual information. It contains 12,000 annotated images covering a large number of indoor and outdoor scenes, for more information on this data set reader can refer to [193]. We used about 9,000 of these images with 111 classes as valid categories for our purpose of detection, with 111 object categories the average number of objects in the whole database was 10.5 objects per image. Table 5.3 shows the mean average precision of all object categories obtained from three context models along with the MAP of baseline detector, as the table indicates sparse model method outperforms other context models.

In this section, we saw that even with an unoptimized extraction of the HOG features, our
Figure 5.6: **a) Mean AP across all classes vs. maximum number of initial detections.** For this experiment average precision of three context models are computed using comp3 protocol, AP is computed for varying number of Maximum initial detections from 1 to 10. The Average Precision increases as we increase the number of detections this a behaviour that we expect to occur, since some images might contain more than one object of a certain type by confining detections to lower number of detections we impose ourself to discard some detections and as we increase the number of detections we include some more detection in accuracy computation. As this graph depicts for lower number of initial detection our algorithm outperforms other context models but as the number of detections increases due to intervention of lots of low confident detection in score function the average procession drops in compare to SVM based context model. **b) Sorted Improvement.** class by class improvement comparison of sparse context model with the results of other methods on VOC2007 (in this experiment maximum 10 initial detections form each baseline detectors are used) the results illustrate that while some methods like tree base method can deteriorate the detection accuracy at certain classes SVM-based and Sparse methods are proof to deteriorate the detection accuracy while context rescoring.

Method is on par with similar state-of-the-art approaches. In particular, on the VOC07 dataset, exploiting the context aware dictionary has led to a 7% increase of the per-class mean average precision, with respect to not using contextual information. These results are a clear indication that the basic idea of the approach is valid and worth exploiting in future developments.

Using K-SVD we learned a context aware dictionary for sparse representation of initial hypothetical detected boundaries and a model for re-scoring these initial boxes, On the other hand using methods introduced recently for low rank decomposition through shrinkage function of SVD, we developed a novel framework to learn an orthogonal span of each unique identity/object, where learned subspaces are orthogonal to other structured nuisance factors, which enable us with fast and simple test statistics for decision making of new samples. Initially, we presented a technique for learning an over-complete dictionary for sparse representation of signals that is compact, reconstructive, discriminative, and that simultaneously learns a set of linear regression maps. Even
Table 5.3: **Mean AP (Averaged across all object categories on the SUN09).** Baseline is the AP obtained using baseline detector without any context model [186] (Voc-release5 available online has been used to produce baseline results). SVM depicts the results obtained by context SVM model proposed by [186]. Tree-based are the AP obtained from Hierarchical Context model in [193]. Sparse is the results for proposed method using compact KSVD dictionary learning.

<table>
<thead>
<tr>
<th></th>
<th>Baseline</th>
<th>SVM</th>
<th>Tree-based</th>
<th>Sparse</th>
</tr>
</thead>
<tbody>
<tr>
<td>Avg</td>
<td>6.815</td>
<td>8.093</td>
<td>7.910</td>
<td>8.143</td>
</tr>
</tbody>
</table>

Figure 5.7: **Detection rescoring instances** Here we provide couple of rescoring results obtained from SVM-context method and sparse method for comparison purpose, second column illustrate the 3 most confident detections obtained from running 20 object category detectors, in order to better visualise rescoring results on detections we only keep 10 most confident detections among all initial detections, yet again the best comparison could be obtained on digital screen by magnifying images, third column contains rescoring results from svm-context and forth column is the results obtained from our regression method.
though such maps are suitable for modeling and predicting contextual information, their use is not restricted to this scenario. Important properties about this framework include numerical stability, and the ability to scale well with the number of parameters to learn, and with the size of the dataset. This is mainly due to the use of the K-SVD, and robust online algorithms for computing the SVD. We tested this idea by designing a structured prediction framework for simultaneously localizing and recognizing multiple objects from multiple classes, and that exploits the sparse domain representation to perform a very fast assignment of the class labels. We have shown that the approach is able to learn meaningful layout distributions. We have verified that the method is on par with similar state-of-the-art approaches. In particular, on the VOC07 dataset, exploiting the context aware dictionary has led to a 7% increase of the per-class mean average precision, with respect to not using contextual information. These results are a clear indication that the basic idea of the approach is valid and worth exploiting in future developments. Also we would like to point out that by incorporating in our framework the label-consistent contribution proposed in [57], it would be possible to increase the discriminative power and improve the compactness of the dictionary. This would benefit the scalability of the approach with respect to the number of classes. We leave for future work testing this extension with the SUN09 dataset [191], which includes images from many more object classes than the VOC07, and is well suited for testing models of contextual interactions. Figure 5.8 illustrate the rescoring result on VOC07 challenge.

Finally, we would like to point out that by incorporating in our framework the label-consistent contribution proposed in [57], it would be possible to increase the discriminative power and improve the compactness of the dictionary. This would benefit the scalability of the approach with respect to the number of classes. We leave for future work testing this extension with the SUN09 dataset [191], which includes images from many more object classes than the VOC07, and is well suited for testing models of contextual interactions.

Figure 5.8: Object localization and recognition. Samples from the PASCAL VOC 2007 dataset showing the labeling of the bounding boxes hypothesis with top confidence rank, computed by the proposed structured prediction framework.
Chapter 6

Conclusions and Future Work

In this dissertation, we introduced two new concepts in the linear modeling of data points, using linear additive model and noise subspaces. *Parity Space* is the orthogonal complement of model subspace, representing a subspace that can be used to measure the distance of an input sample from the model subspace. Similarly, *Invariant Space* is the orthogonal complement of the structured noise subspace, representing a unique subspace pertinent to the model. We introduced a well-grounded framework to learn a metric on parity subspace by combining RKHS geometry and the concept of parity subspace. With extensive experiments, we proved the efficiency of parity subspace for online human activity detection applications.

On the other hand, exploiting the idea of the invariant subspace, we introduced two different frameworks to leverage this subspace for visual recognition and domain generalization of visual data. We used recent advancements in the area of low-rank decomposition and Augmented Lagrangian Multiplier (ALM) optimization to define the formulation and solve the optimization problem.

In chapter (2), we introduced a new test statistics on *parity subspace* using Gram matrix of two different sequences of data, where enables us to map data sequences using non-linear transforms to a Hilbert space and detect any changes in the structure of Gram Matrix. Using our models and test statistics, we gained 3% improvement in the accuracy and up to 9% improvement in the timeliness of online activity detection compared to state of the art online detection methods. As a result of the successful implementation of new statistics defined on parity subspace, in conjunction with nonlinear mapping functions on RKHS and obtaining promising results for online human activity recognition using new metrics, we are motivated to test the idea of new statistics on invariant subspace for online human activity recognition. As we described in this chapter, new metrics on parity subspace were in introduced to measure the distance of new samples from the model subspace; However for future work, we are proposing that we consider learning the nui-
sance dynamics of the video instead and define new statistics based on invariant subspace which is an orthogonal complement to nuisance dynamics. Using these new dynamics will empower us to do the online activity detection even without the need for learning of the model in real time. Also another direction that we would like to investigate for the future work is utilization of linear low-rank decomposition approaches in Hilbert space, Since as we saw in Chapter 2 linear models tend to produce promising results in Hilbert space such as our new test statistics, therefore such linear techniques when combined with non-linear projection operators, is expected to outperform many approaches proposed for detection and recognition of low-rank behaviours expected to happen in any sequence of data.

Following the main idea of this dissertation, in the chapter (3), we introduce a new formulation for implementing the metrics on the invariant space. In this section, we present a mathematically well-grounded formulation for learning the variation subspace and illustrate how this idea can be used for visual recognition tasks like face recognition. We introduce an optimization problem to learn the variation and invariant subspace (identity subspaces), where we used a low-rank matrix to model the variation subspace and we used a Euclidean distance metric for measuring the distance from invariant subspace. For solving this optimization problem, we use Augmented Lagrangian Multiplier, and we use recursive methods to learn the proposed subspaces. Using extensive experiments, we proved the effectiveness of the proposed method in compare to the state of the art methods. In chapter (4), we introduced a different framework using the same decomposition model for domain generalization application. In this case, the structured variation to model is the shift in dataset bias and model space are shared between data sets. Therefore by modeling the dataset shift as variation subspace and learning the classifier on invariant subspace, we eliminate the effect of domain shift on the classifier.

We presented a new domain generalization approach using max margin formulation, we used low-rank matrix decomposition to subtract the dataset bias and nuisance factors in learning process which are orthogonal to the max margin classifier, while we are sure the soft margin formulation will take care of noisy data samples, our model is able to handle both data distribution shift and outlier noisy data from the training function simultaneously, and consequently converge to combination of robust support vectors and decision boundary that has a good generalization property. Experimental section of this chapter proves the higher performance of this generalization method in compare to state of the art domain adaptation and generalization methods. Extensive experiments using features obtained by deep learning methods (DeCAF) with the state of the art performance in visual applications, proves the effectiveness of the proposed method and increases the accuracy of recognition on domain adaptation and generalization benchmark datasets. The successful implementation of these two visual recognition application introduced in chapters (3, 4) proves the
effectiveness of utilization of the invariant subspace in the visual recognition applications.

In continuation of our studies over subspace analysis to improve the recognition accuracy by exploitation of sparse and low-rank methods, in another attempt, we introduced a new low-rank matrix decomposition method, for learning a nuisance factor space, and an identity space orthogonal to it simultaneously. Based on our purpose of recognition, we formed an objective function to decompose the data matrix into significant components, where the low-rank matrix models conjoint nuisance subspace, the sparse matrix models noise of individual data that is not captured by the conjoint nuisance subspace. Most importantly, we learn a new matrix for identity representation $Y$, which is one of the main contributions in chapter 3. We solve the corresponding objective function with an optimization procedure that builds on ALM and a Sylvester equation. The experimental section confirms the validity of the theoretical proposal of the chapter 3, in face recognition and even in a more general spectrum like person identification. We study any possible situation which may occur while setting the parameters for proposed objective function, We thoroughly examine any possible situation in proposed model and illustrate that the optimum objective function proposed in Eq. 3.28 produce the best recognition performance among state of the art methods proposed for modeling the gross noise. The experimental section confirms the validity of the proposed theory in face recognition and even in more general spectrum like person identification.

This work has presented a technique for learning an over-complete dictionary for sparse representation of signals that is compact, reconstructive, discriminative, and that simultaneously learns a set of linear regression maps. Even though such maps are suitable for modeling and predicting contextual information, their use is not restricted to this scenario. Important properties about this framework include numerical stability, and the ability to scale well with the number of parameters to learn, and with the size of the dataset. This is mainly due to the use of the K-SVD, and robust online algorithms for computing the SVD. We tested this idea by designing a structured prediction framework for simultaneously localizing and recognizing multiple objects from multiple classes, and that exploits the sparse domain representation to perform a very fast assignment of the class labels. We have shown that the approach is able to learn meaningful layout distributions. Moreover, even with an unoptimized extraction of the HOG features, we have verified that the method is on par with similar state-of-the-art approaches. In particular, on the VOC07 dataset, exploiting the context aware dictionary has led to a 7% increase of the per-class mean average precision, with respect to not using contextual information. These results are a clear indication that the basic idea of the approach is valid and worth exploiting in future developments. Finally, we would like to point out that by incorporating in our framework the label-consistent contribution proposed in [57], it would be possible to increase the discriminative power and improve the compactness of the dictionary. This would benefit the scalability of the approach with respect to the number of classes. We
leave for future work testing this extension with the SUN09 dataset [191], which includes images from many more object classes than the VOC07, and is well suited for testing models of contextual interactions.
Bibliography


[12] John Wright, Arvind Ganesh, Shankar Rao, Yigang Peng, and Yi Ma. Robust principal component analysis: Exact recovery of corrupted low-rank matrices via convex optimiza-
tion. In *Advances in neural information processing systems*, pages 2080–2088, 2009. 1.2, 1.2, 1.3.3, 3.1, 3.2, 3.4, 3.7.2, 3.1, 3.7.2, 3.7.4, 3.7.6, 4.1, 4.3.2, 5.1, 5.1


[25] Zechao Li, Jing Liu, Jinhui Tang, and Hanqing Lu. Robust structured subspace learning


[41] René Vidal and Paolo Favaro. Low rank subspace clustering (lrsc). Pattern Recognition Letters, 2013. 1.2, 1.3.3, 3.2


[43] Chih-Fan Chen, Chia-Po Wei, and Y.-C.F. Wang. Low-rank matrix recovery with structural incoherence for robust face recognition. In Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, pages 2618–2625, June 2012. 1.2, 3.1, 3.2, 3.4, 3.5.1, 3.7.2, 3.7.2, 3.1, 3.7.4


[45] Yangmuzi Zhang, Zhuolin Jiang, and Larry S Davis. Learning structured low-rank representations for image classification. In IEEE CVPR, pages 676–683, 2013. 1.2, 3.1, 3.2, 3.5.1, 3.5.3, 3.7.2, 3.1, 3.7.4, 4.3.2


[52] Qiang Zhang and Baoxin Li. Discriminative k-svd for dictionary learning in face recognition. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2691–2698, 2010. 1.2, 5.1, 5.2.1, 5.2.1, 5.3.1, 5.3.2


[57] Zhuolin Jiang, Zhe Lin, and L. S. Davis. Learning a discriminative dictionary for sparse coding via label consistent k-svd. In *IEEE CVPR*, pages 1697–1704, 2011. 1.2, 5.1, 5.2.1, 5.3.1, 5.3.2, 5.4.2, 6


[62] E. Candés, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *Journal of the ACM*, 58(3), 2011. 1.3.2, 1.3.2, 1.3.2, 1.3.2, 3.1, 3.4, 3.5.1, 3.5.1


[68] S. Motiian, K. Feng, H. Bharthavarapu, S. Sharlemin, and G. Doretto. Pairwise kernels for human interaction recognition. In *Advances in Visual Computing*, volume 8034, pages 210–221, 2013. 2.1, 2.2, 2.5.1, 2.5.1, 2.6, 2.6


[75] C. Brunner, A. Fischer, K. Luig, and T. Thies. Pairwise support vector machines and their application to large scale problems. *jmlr*, 13:2279–2292, August 2012. 2.5.1, 2.5.1


[77] W. Li and N. Vasconcelos. Recognizing activities by attribute dynamics. In *NIPS*, 2012. 2.6

[78] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines.


3.2, 3.5.1, 3.5.3


[99] Xin Zhang, Zhouchen Lin, Fuchun Sun, and Yi Ma. Rectification of optical characters as transform invariant low-rank textures. In *ICDAR*, pages 393–397, Aug 2013. 3.2


[125] Martin Hirzer, Peter M Roth, Martin Köstinger, and Horst Bischof. Relaxed pairwise learned metric for person re-identification. In ECCV, pages 780–793, 2012. 3.6.1

[126] Zhen Cui, Wen Li, Dong Xu, Shiguang Shan, and Xilin Chen. Fusing robust face region descriptors via multiple metric learning for face recognition in the wild. In IEEE CVPR, pages 3554–3561, June 2013. 3.6.1


[140] Kilian Q Weinberger and Lawrence K Saul. Distance metric learning for large margin nearest neighbor classification. JMLR, 10:207–244, 2009. 3.7.7


[142] Rui Caseiro, João F Henriques, Pedro Martins, and Jorge Batista. Beyond the shortest path: Unsupervised domain adaptation by sampling subspaces along the spline flow. In
Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 3846–3854, 2015. 4.1, 4.2


[153] Judy Hoffman, Brian Kulis, Trevor Darrell, and Kate Saenko. Discovering latent domains


[163] Si Si, Dacheng Tao, and Bo Geng. Bregman divergence-based regularization for transfer subspace learning. Knowledge and Data Engineering, IEEE Transactions on, 22(7):929–942, 2010. 4.1, 4.2

[164] Rahaf Aljundi, Rémi Emonet, Damien Muselet, and Marc Sebban. Landmarks-based kernelized subspace alignment for unsupervised domain adaptation. In Computer Vision and Pattern Recognition (CVPR’2015), 2015. 4.1, 4.2


and fuzzy logic models. MIT press, 2001. 4.3.2


127


network.org/challenges/VOC/voc2007/workshop/index.html, 2007. 5.1, 5.2