Sparse visual signal representations and selected applications

Hung, Tzu-Yi

2015

Hung, T.-Y. (2015). Sparse visual signal representations and selected applications. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/65048>

<https://doi.org/10.32657/10356/65048>

SPARSE VISUAL SIGNAL REPRESENTATIONS AND SELECTED APPLICATIONS

HUNG TZU-YI

School of Electrical and Electronic Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfillment of the requirement for the degree of
Doctor of Philosophy

2015

Acknowledgments

I would like to express my sincere thanks and appreciation to my supervisor, Dr. Yap-Peng Tan for his invaluable guidance, support and suggestions. His knowledge, suggestions and discussions help me to become a capable researcher.

I would also like to thank my senior, Dr. Jiwen Lu, who has always provided valuable comments in discussion, and seniors Dr. Gao Yang and Dr. Likun Huang for their kind support. I am thankful to Seong-Ping Chuah, Maodong Li, Tao Wang, Qiang Li, Wenwen Wang, Ye Luo, Shan Luo, Yi Jin, Dongye Zheng, Lu Wang, Renliang Weng, Junlin Hu and Dr. Chia-Wen Lin for their friendship and encouragement. My sincere thanks to the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, for providing me the full research scholarship and support to complete my Ph.D study.

Last but not least, I would like to dedicate this thesis to my family, my father Fu-Lin, mother Ling-Hui, husband Wei-Ting and sisters, Irene, Nikki and Mandy, and thank for their unconditional love, support, encouragement, patience and understanding throughout all these years of my study. Without them, this would have never been possible.

Tzu-Yi Hung

Singapore, 2015

Contents

Acknowledgments	i
Summary	xi
List of Figures	xv
List of Tables	xxii
List of Abbreviations	xxv
List of Notation	xxvii
1 Introduction	1
1.1 Background	1
1.2 Motivation	5
1.3 Contributions of the Thesis	8
1.4 Organization of the Thesis	10
2 Literature Review	15

2.1	General Overview on Sparse Visual Signal Representation	15
2.2	Sparse Representation Models	17
2.2.1	K-means	17
2.2.2	Fuzzy C-means	18
2.2.3	Sparse Coding	19
2.2.4	Laplacian Sparse Coding and Graph Regularized Sparse Coding	21
2.2.5	Smooth Sparse Coding	22
2.2.6	Summary	23
2.3	Sparsity Estimation Algorithms	24
2.3.1	Feature-sign Search	24
2.3.2	Marginal Regression	26
2.3.3	Summary	27
2.4	Dictionary Learning Methods	29
2.4.1	Conjugate Gradient Descent	29
2.4.2	Gradient Projection Descent	30
2.4.3	Summary	31
2.5	Selected Applications	31
2.5.1	Activity-based Human Identification	32

2.5.2	Handwritten Digit Recognition	33
2.5.3	Image Classification	33
2.5.4	Action Recognition	34
3	Graph-based Sparse Coding and Embedding	37
3.1	Introduction	37
3.2	Graph-based Sparse Coding and Embedding	40
3.2.1	Motivation	40
3.2.2	Proposed Approach	41
3.2.3	Proposed Algorithms	43
3.3	Activity-based Human Identification	49
3.3.1	Feature Extraction	50
3.3.2	Feature Coding	51
3.3.3	Video Representation	52
3.3.4	Human Identification	53
3.3.5	Data Sets	54
3.3.6	Parameter Settings	54
3.3.7	Comparisons with Existing Activity-based Human Identifica- tion Methods	55

3.3.8	Comparisons with Existing Sparse Coding Methods	56
3.3.9	Parameter Analysis	56
3.3.10	Confusion Matrix	58
3.3.11	Discussion	61
3.4	Summary	62
4	Sparse Manifold Coding	63
4.1	Introduction	63
4.2	Unsupervised Sparse Manifold Coding	65
4.2.1	Motivation	67
4.2.2	Unsupervised Sparse Manifold Coding	68
4.2.3	Sparsity Estimation	70
4.2.4	Dictionary Learning	74
4.3	Supervised Sparse Manifold Coding	76
4.4	Activity-based Human Identification	78
4.4.1	Data Sets	79
4.4.2	Comparisons with Existing Activity-based Human Identifica- tion Methods	81
4.4.3	Comparisons with Existing Sparse Coding Methods	82

4.4.4	Speed Comparison	83
4.4.5	Confusion Matrix	85
4.4.6	Analysis of Weighting Parameters	87
4.4.7	Discussion	89
4.5	Summary	90
5	Marginal-Lasso Coding	93
5.1	Introduction	93
5.2	Marginal Regression for Feature Quantization	97
5.3	Marginal-Lasso Coding	99
5.3.1	Motivation	99
5.3.2	Marginal-Lasso Coding	101
5.3.3	Sparsity Estimation and Parameter Selection	104
5.3.4	Dictionary Learning	106
5.4	Activity-based Human Identification	109
5.4.1	Comparisons with Existing Marginal-based Sparsity Estimation Methods	112
5.4.2	Comparisons with Existing Activity-based Human Identification Methods	113

5.4.3	Comparisons with Existing Sparse Coding Methods	115
5.4.4	Confusion Matrix	115
5.5	Handwritten Digit Recognition	118
5.6	Image Classification	120
5.6.1	Experimental Settings	121
5.6.2	Scene 15 Dataset	122
5.6.3	Caltech 101 Dataset	122
5.6.4	Caltech 256 Dataset	124
5.6.5	Speed Comparison of Sparsity Estimation Methods	125
5.6.6	Quantization Error of Sparsity Estimation Methods	125
5.6.7	Parameter Selection	126
5.7	Action Recognition	127
5.7.1	Experimental Settings	128
5.7.2	Recognition Accuracy	130
5.7.3	Sparsity Analysis	132
5.7.4	Speed Comparison	138
5.7.5	Feature Quantization Error	139
5.7.6	Discussion	140

5.8	Summary	143
6	Conclusions and Future Work	145
6.1	Conclusions	145
6.1.1	A Lasso-based Method for Feature Coding — the High-dimensional Case	146
6.1.2	A Marginal Regression-based Method for Feature Coding — an Unsupervised Solution	148
6.1.3	A Marginal Regression-based Method for Feature Coding — a Supervised Solution	149
6.1.4	A Marginal-lasso-based Method for Feature Coding — an Efficient and Robust Solution	151
6.2	Future Work	152
6.2.1	Activity-based Human Identification	152
6.2.2	Sparse Visual Signal Representation	154
	Publications	157
	References	159

Summary

Sparse representation has been well investigated and discussed over the past decade due to its ability in visual signal discrimination for various applications such as face recognition, image classification and video clustering. It has attracted more and more interest in the recent years because of the increasing demands for developing real world systems with large-scale image and video collections. While a large number of sparse representation algorithms have been proposed in the literature and some encouraging results have been obtained, there is still a need for further improvement. This thesis aims to address various issues of sparse representation, including feature quantization models, sparsity estimation methods and dictionary learning techniques for sparse visual signal representation over different computer vision and pattern recognition tasks such as image classification, action recognition and activity-based human identification to demonstrate their efficacy and superiority over state-of-the-art methods. More specifically, we focus our work on two directions: 1) An application-oriented problem: we investigate the problem of activity-based person identification which will be elaborated in the thesis; and 2) A model-oriented problem: we improve the existing sparse coding approaches in a more efficient and effective way and evaluate the performance of the proposed method on several visual tasks.

Motivated by the problems of human action recognition and gait recognition,

we investigate the problem of activity-based human identification which aims to identify people from their activities. There are many potential applications such as intelligence surveillance and public safety, which may demand the recognition of people from their activities at a distance without a clear view of their faces. Inspired by the fact that most existing methods usually quantize features independently and ignore the geometrical structure of the data space, and further, the number of training images is usually much smaller than the dimension of each human body feature, we propose a graph-based sparse coding and embedding (GSCE) method to quantize these dynamic patterns of multiple activity movements for recognition. We consider the geometrical structure between features in a compact low-dimensional feature space such that the intrinsic characteristics of the original data can be well-preserved and redundancy information can be removed.

While some of sparse coding methods have considered the geometrical structure of the data space for activity-based human identification, there is little work on exploiting the dependency of features in/between video manifolds which consist of a series of human movements. From the fact that features lying on the same manifolds are extracted from the same human movements, we propose a Sparse Manifold Coding (SMC) approach to explicitly utilize the local dynamic information in the manifold and simultaneously consider the global geometrical structure between multiple manifolds for discriminative feature quantization. We also propose a marginal regression approach to solve the optimization problem. To encode new data, we propose a NN-based sparse code approximation method which can easily trace the geometrical structure in the data space. Specifically, we present a Supervised Sparse Manifold Coding (SSMC) approach to integrate hierarchical manifold structure with class information.

Since most existing sparse coding solutions are computational intensive, especially when a large over-complete dictionary is used, the scalability for large-scale image/video collections remains a challenge. While some marginal regression solutions are proposed to expedite the quantization procedure, they work well when the columns of the dictionary have low correlation. In general, we have an over-complete dictionary with highly correlated atoms, and thus it may cause a large quantization error. To address this problem, we propose a Marginal-Lasso Coding (MLC) approach which incorporates marginal regression coefficients with an l_1 penalty for feature quantization and sparsity estimation. MLC obtains each sparse code by an adaptive regularization parameter, and simultaneously shrinks coefficients to alleviate the bias of sparse codes caused by the highly correlated visual words to further improve the performance.

The above-mentioned methods have been used to address several challenges of computer vision applications, such as image classification, action recognition, and activity-based human identification. Experimental results are presented to demonstrate the efficacy of the proposed methods.

List of Figures

1.1	An example showing the processes of image and video representation.	3
1.2	An example showing the concept of dictionary, visual words, hard assignment and soft assignment mechanisms.	4
1.3	The structure of the models discussed in the thesis. Blocks in orange color are our work.	11
1.4	The structure of activity-based human identification discussed in the thesis. Blocks in orange color are our work.	12
3.1	Flowchart of the proposed activity-based human identification method.	50
3.2	The steps of feature extraction. From left to right, given a sample image, background subtraction and thresholding is used to obtain a binary mask. Then, we align the binary mask to reduce the effect caused by the distance between camera and the subject. By doing so, we can obtain the region of interest (ROI). Finally, we resize the ROI into 64x48 to mitigate the computational complexity.	51

3.3	Two series of sample ROI frames from two video sequences performed drinking behavior by two different persons. From left to right are the binary ROI masks of movement performed by the same person. The rows show different persons performing the same activity.	51
3.4	Sample frames from 9 different activity video clips performed by the same person in the Weizmann database. From left to right and top to bottom are the activities of jumping jack, jump, jump in place, run, gallop-sideways, skip, walk, one-hand wave, and two-hands wave, respectively.	53
3.5	Sample frames from two activity video sequences in the MOBISERV-AIIA database. We use two activities, drinking with a cup and eating with a fork, and two clothing scenarios, one wearing long sleeves and one wearing short sleeves, respectively. From left to right are the person drinking with a cup with long sleeves, the person drinking with a cup with short sleeves, the person eating with a fork with long sleeves and the person eating with a fork with short sleeves, respectively.	55
3.6	Average recognition rate (%) of our GSCE versus different feature dimension on the Weizmann dataset	57
3.7	Average computational cost (second) of our GSCE versus different feature dimension on the Weizmann dataset	58
3.8	Sample codebook pictures of the Weizmann dataset.	59
3.9	Sample codebook pictures of the MOBISERV-AIIA dataset.	59

4.1	Flowchart of the proposed sparse manifold coding framework for activity-based human identification.	66
4.2	The basic idea of our sparse manifold coding approach, which emphasizes the dependency of features within each movement manifold and simultaneously maintains the global structure between features in the data space, so that similar features, especially for those extracted from the same movement sequences may obtain similar sparse codes.	67
4.3	Unsupervised sparse manifold coding (USMC) model. For this example, each video form a video-based manifold denoted by $[\mathcal{V}_1, \dots, \mathcal{V}_8]$ with its local features $[x_1, \dots, x_n]$ denoted by circles. Note that n is number of features which varies according to each clip. We aim to minimize the quantization error and simultaneously maintain the geometrical structure within and between manifolds. To achieve this goal, we construct two similarity matrices denoted by S^v and S^u with different thresholding parameters w^v and w^u , respectively. Therefore, the red dot feature is quantized respect to its selected neighborhood within and between manifolds.	68
4.4	NN-based sparsity approximation. In this example, the new feature denotes by the black circle x_t . $W^t(x_t, x_i) = x_i^T x_t$ calculates the correlation between new sample and every sample x_i in the training dataset. By finding the most correlated sample x_o which denotes by the red circle, we assign sparse codes b_o^* to x_t to maintain the geometrical structure.	73

4.5	Supervised sparse manifold coding (SSMC) with video-based and person-based manifolds structure. For this example, we form two layers of manifold. Each video form a first layer video-based manifold denoted by $[\mathcal{V}_1, \dots, \mathcal{V}_8]$ with its local features $[x_1, \dots, x_n]$ denoted by circles. Note that n is number of features which varies according to each video clip. Further, we construct a person-based manifold for each class. As shown, $\mathcal{V}_1, \mathcal{V}_2, \mathcal{V}_5, \mathcal{V}_6$ build a person-based manifold and $\mathcal{V}_3, \mathcal{V}_4, \mathcal{V}_7, \mathcal{V}_8$ build another so that each feature with the corresponding person-based manifold shares the same person information.	76
4.6	Three activity examples of five persons from the Weizmann dataset. Every two columns refer to successive frames of the jump, side, and skip activities from left to right, respectively. Each row represents a person performing three activities.	80
4.7	There are sample frames for the MOBISERV-AIIA dataset. The first raw is the activity of drinking with a cup, and the second raw is the activity of eating with a fork. Different columns mean different clothing scenarios for a person.	81
4.8	Sample codebook pictures of the Weizmann dataset.	86
4.9	Sample codebook pictures of the MOBISERV-AIIA dataset.	86
4.10	Color map of correct recognition rate of our USMC versus different parameter settings on the Weizmann dataset. The brighter block obtains higher recognition accuracy.	92

4.11	Color map of correct recognition rate of our SSMC versus different parameter settings on the Weizmann dataset. The brighter block obtains higher recognition accuracy.	92
5.1	Distributions of two sampled marginal regression coefficients, s_A and s_B . (a) With a fixed tuning parameter $t = 0.75$, some of the coefficients may shrink too much such as s_B ; instead, with a fixed regularization parameter $t = 0.55$, some of the coefficients may include too much noise such as s_A . (b) With an l_1 energy constraint $\ s\ _1 \leq E$, each feature can be represented by similar sparsity level.	95
5.2	Sample frames of the Weizmann dataset. From left to right are sample image frames of handwaving, jumping-forward-on-two-legs, jump-in-place and gallop-sideways.	111
5.3	Sample frames of the AIIA dataset. From left to right, every two samples are the same clothing scenario with two activities, eating with a fork and drinking with a cup.	112
5.4	Sample codebook pictures of the Weizmann dataset.	116
5.5	Sample codebook pictures of the MOBISERV-AIIA dataset.	116
5.6	Sample digits from 0 to 9 of the USPS dataset.	119
5.7	Speed comparison between different sparsity estimation methods on Caltech 101 data set (in Seconds).	126
5.8	Quantization error comparison between different sparsity estimation methods on Caltech 101 data set.	127

5.9	Performance comparison between different values of sparsity and energy self-ratio on Caltech 101 data set.	129
5.10	Performance comparison between different values of energy bound on Caltech 101 data set.	130
5.11	Sample frames of the KTH data set. From left to right are four scenarios: outdoors, outdoors with scale variation, outdoors with different clothes and indoors. From top to down are three example activities: boxing, handclapping and handwaving.	131
5.12	The average amount of non-zero elements of sparse codes quantized from the HOF feature descriptors using marginal regression with simple hard thresholding approach from the KTH action recognition training set as t varies. The dictionary size is set as 4000 in this example.	135
5.13	Comparison of the soft and hard-thresholding schemes. (a) The average l_1 norm energy of sparse codes under different t . As shown, using the soft-thresholding scheme is a more effective way for variable selection and coefficient shrinkage. (b) The average standard deviation of the l_1 energy between sparse codes as t changes. Sparse codes obtained via the soft-thresholding scheme are much stable than those obtained via the hard-thresholding scheme under different t . . .	136

5.14	The results in terms of recognition accuracy using the soft and hard thresholding schemes under three feature descriptors, HOG, HOF, and HOGHOF as t varies, respectively. As shown, the accuracy of the soft-thresholding scheme is much better than that of the MR-Hard method.	137
5.15	Recognition accuracy under different self-ratios. We obtain around 90% recognition accuracy from $e = 5\%$ to $e = 35\%$, and obtain poor recognition accuracy (below 10%) from $e = 75\%$ and upwards.	137

List of Tables

2.1	Categorization of sparse representation methods. (“*”the proposed methods)	23
2.2	Categorization of sparse coding methods according to different data structures. (“*”the proposed methods)	24
2.3	Summary of the different sparsity estimation approaches in terms of the quantization error and the computational complexity. (“*”the proposed methods)	27
2.4	Categorization of sparse coding methods according to different sparsity estimation approaches. (“*”the proposed methods)	29
2.5	Categorization of sparse representation methods according to the dictionary learning approach. (“*”the proposed methods)	31
3.1	CRR comparison (%) with the existing activity-based human identification methods.	56
3.2	CRR comparison (%) with the existing sparse representation methods.	57
3.3	Confusion matrix of our GSCE method on the Weizmann dataset. . .	60

3.4	Confusion matrix of our GSCE method on the MOBISERV-AIIA dataset.	60
4.1	Comparison with existing methods for the Weizmann dataset.	82
4.2	Comparison with existing methods on the MOBISERV-AIIA dataset.	82
4.3	Comparison with existing sparse coding methods on the Weizmann dataset.	83
4.4	Comparison with existing sparse coding methods on the MOBISERV-AIIA dataset.	83
4.5	Speed comparison between lasso solution and marginal regression.	84
4.6	Speed comparison between GSCE and SMC on the Weizmann dataset.	84
4.7	Confusion matrix of our USMC method on the Weizmann dataset.	85
4.8	Confusion matrix of our USMC method on the MOBISERV-AIIA dataset.	87
4.9	Confusion matrix of our SSMC method on the Weizmann dataset.	88
4.10	Confusion matrix of our SSMC method on the MOBISERV-AIIA dataset.	88
5.1	Recognition accuracy (%) of different marginal regression methods on the Weizmann and the MOBISERV-AIIA datasets.	114
5.2	Recognition accuracy (%) of different methods on the Weizmann and MOBISERV-AIIA datasets.	114

5.3	Comparison with existing sparse coding methods.	115
5.4	Confusion matrix of our MLC method on the Weizmann dataset. . .	117
5.5	Confusion matrix of our MLC method on the MOBISERV-AIIA dataset.	117
5.6	Recognition accuracy (%) of different methods on the USPS dataset.	118
5.7	Recognition accuracy (%) of different methods on the Scene 15 dataset.	120
5.8	Recognition accuracy (%) of different methods on the Caltech 101 dataset.	123
5.9	Recognition accuracy (%) of different methods on the Caltech 256 dataset.	124
5.10	Quantization error of different sparsity estimation methods	128
5.11	Recognition accuracy (%) of different methods using HOG, HOF, HOGHOF feature descriptors, respectively, on the KTH data set. . .	133
5.12	Computational complexity of different methods on the KTH dataset. The results are shown in terms of speed. Notice that here we randomly sampled 1,258 features from the training set.	141
5.13	Quantization error of different methods on the KTH dataset. The results are shown in terms of quantization error. Notice that here we randomly sampled 1,258 features from the training set.	142

List of Abbreviation

ABBREVIATION	FULL EXPRESSION
AFS	Approximated Feature-sign Search
BoF	Bag-of-Features
Bow	Bag-of-Words
BP	Basis Pursuit
DicSize	Dictionary Size
DicSpar	Dictionary Sparsity
FS	Feature-sign Search
FCM	Fuzzy C-means
GraphSC	Graph Regularized Sparse Coding
GSCE	Graph-based Sparse Coding and Embedding
HOG	Histogram of Oriented Gradient
HOF	Histogram of Optical Flow
LAR	Least Angle Regression
Lasso	Least Absolute Shrinkage and Selection Operator
LDA	Linear Discriminant Analysis
LSC	Laplacian Sparse Coding
LScSPM	Laplacian Sparse Coding using spatial pyramid matching
MLC	Marginal-lasso Coding

MLcSPM	Marginal-lasso Coding using spatial pyramid matching
MR	Marginal Regression
MR-Hard	Marginal Regression wit Hard-thresholding
MR-Soft	Marginal Regression wit Soft-thresholding
OR	Original representation
ROI	Region of Interest
SC	Sparse Coding
ScSPM	Sparse Coding using spatial pyramid matching
SIFT	Scale-Invariant Feature Transform
SIS	Sure Independent Screening
SMC	Sparse Manifold Coding
SSMC	Supervised Sparse Manifold Coding
SPM	spatial pyramid matching
SR	Sparse Representation
SSC	Smooth Sparse Coding
SVD	Singular Value Decomposition
USMC	Unsupervised Sparse Manifold Coding
VQ	Vector Quantization

List of Notation

NOTATIONS	EXPLANATION
x	Vector
X	Matrix
$\ X\ _F$	Frobenius norm of X defined as $\ X\ _F = \sqrt{\sum_{i,j} X_{ij}^2}$
$\ X\ _F^2$	Power of Frobenius norm defined as $\ X\ _F^2 = \sum_{i,j} X_{ij}^2$
$\ x\ _0$	l_0 -norm of x which counts the number of nonzero elements
$\ x\ _1$	l_1 -norm of x defined as $\ x\ _1 = \sum_{i=1}^d x_d $
$\ x\ _2$	l_2 -norm of x defined as $\ x\ _2 = \sqrt{\sum_{i=1}^d x_i^2}$
$x \succeq 0$	All the elements of x are nonnegative
$(\cdot)^T$	Matrix or vector transpose
$Card(x)$	Cardinality constraint meaning that x has only one nonzero element
$tr(X)$	Trace operator defined as $\sum_{i=1}^n x_{ii}$
$N_k(x)$	A set of k nearest neighbors of x

Chapter 1

Introduction

1.1 Background

With the rapid development of Internet technology and increasing popularity of digital devices in recent years, the amount of images and video clips on the internet has increased dramatically. For instance, 100 hours of video are uploaded to YouTube every minute [1], while 60 millions photos on Instagram are shared per day [2]. For such a huge data pool, the need of visual signal applications, such as image/video content analysis, image/video classification, image annotation, video event detection, action recognition, object recognition, scene categorization and face verification etc., has led to extensive research on machine learning, pattern recognition, computer vision, data mining and information search. To realize these applications, one is often confronted with problems of visual signal representation. While a number of representation algorithms have been proposed in the literature and encouraging results have been obtained, how to describe and represent a set of visual signals efficiently still remain a challenge. In particular, it is much desired to further reduce computational complexity and improve the classification performance of visual signal classifiers.

One of the most effective techniques to address such problems is the bag-of-features (BoF) model or so-called Bag-of-words (BoW) model, named by considering an image/video as a collection of unordered visual words, which has been widely used in many real world applications [3–50]. The general process can be summarized into four steps: feature extraction and description, followed by feature clustering and polling. More specifically, an interest point feature detector is used to extract salient local regions and each region is described by a local descriptor. Then, these descriptors are clustered and quantized into basic patterns, termed as visual words. Finally, distribution histograms are computed by considering spatial or temporal information such as patches from an image or frames from a video clip to represent each image or video, respectively. Figure 1.1 shows an example: 2D-Harris detector [51] and 3D-Harris detector [52] described by the Scale-Invariant Feature Transform (SIFT) [53] descriptor and Histogram of Oriented Gradient (HOG) [54]/Histogram of Optical Flow (HOF) descriptors [8], respectively, are two of the state-of-the-art feature extraction methods for image and video signals, accordingly. With a set of descriptors, an image or a video can be quantized into visual words and be represented by spatial/temporal pooling methods [4].

Among four steps discussed above, the most challenge step is feature clustering which governs the quality of visual signal representation. This is a quantization step including dictionary construction and visual words assignments. A classical example is k -means clustering [4, 21, 28–32, 55] which builds a dictionary by k centers, and assign each local feature to its nearest center. However, this kind of hard assignment mechanisms, which assign a local feature to a visual word, may result in severe information loss, especially when those features are the linear combination of several visual words. Therefore, soft assignment techniques [33, 34] such as fuzzy c -means clustering [56, 57] are introduced to reduce quantization error by assigning each

Figure 1.1: An example showing the processes of image and video representation.

feature to more than one visual word so that each feature has a weighting associated with a particular visual word. However, the number of clusters and the way of assignments need to be determined. The illustrations of visual words, dictionary, hard assignment and soft assignment are shown in Figure 1.2.

More recently, sparse coding (SC) has attracted much interest, and it has been successfully applied to many machine learning and computer vision tasks [58–75]. Sparse coding refers to a feature quantization mechanism which encodes each input signal as a sparse histogram feature based on a linear combination of a small number of visual words from an over-complete dictionary. In other word, the number of the nonzero elements of the coefficient vector for the linear combination is larger than one but much smaller than the number of visual words in the dictionary. Due to its soft assignment principal, it can make the quantization error much smaller, and moreover, it is more robust to the noise and easily incorporating into the BoF framework. The main challenge of sparse coding techniques is variable selection

Figure 1.2: An example showing the concept of dictionary, visual words, hard assignment and soft assignment mechanisms.

with corresponding coefficients from an over-complete dictionary to minimize the reconstruction error which has been shown to be an NP-hard problem [76]. The most representative and popular sparsity estimation model in statistics signal processing is l_1 -norm based method: Least Absolute Shrinkage and Selection Operator called Lasso in short [77, 78] which is a convex optimization problem minimizing the reconstruction error between the original signal and the reconstructed one with an l_1 -norm penalty. The Lasso problem can be solved by Least Angle Regression (LARS) [79], Homotopy [80] and gradient descent algorithms [81–83]. Other l_1 -norm based methods such as penalized least-squares (PLS) [84], adaptive Lasso [85] and relaxed Lasso [86] are proposed to have more discussion on the regularization of l_1 -norm penalty. Besides the Lasso model and its extensions, recently, marginal regression based methods [87, 88] which calculate and screen marginal coefficients for each input signal, have shown its efficiency for sparsity estimation from a large over-complete dictionary. In computer vision and pattern recognition literature, Wright *et al.* [58] proposed the first attempt using sparse coding as visual signal representation for face recognition and obtained reasonable results. At the mean

time, a large number of visual signal representation frameworks [59–62, 64, 89–104], dictionary learning algorithms [105–116], sparse subspace learning approaches [117–123] have been proposed to improve performance of visual signal applications. Yang *et al.* [89] proposed a SIFT-based feature quantization method using sparse coding (SC), instead of the k -means vector quantization (VQ), with linear spatial pyramid matching (SPM) [4], called ScSPM, for image classification. As general sparse coding do not consider the relationship between local features causing similar local features may be quantized into different sparse codes, more recent work has included local geometrical information into the formulation and derived many feature coding algorithms, such as Laplacian Sparse Coding (LSC) [90, 91], Graph regularized Sparse Coding (GraphSC) [59] and Smooth Sparse Coding (SSC) [87], to enhance the recognition performance.

1.2 Motivation

While the above sparse representation methods and sparsity estimation algorithms have achieved reasonable success in many visual signal tasks, there are still several representation and application issues to be addressed, as explained below.

- **From the viewpoint of sparse representation:**

- (1) Most existing sparse coding algorithms assume that there exists an over-complete dictionary so that sparse representation can be achieved by linearly combining a small number of the visual words chosen from the over-complete dictionary. However, they ignore the fact that original feature data may be of high dimension, which gives rise to the difficulty

of learning a large over-complete dictionary due to the high computational complexity. Therefore, is it possible to learn a compact feature subspace to preserve intrinsic local geometrical structure of the original data? Furthermore, is it possible to learn a low-dimensional over-complete dictionary to encode features into sparse representation maintaining their locality relationship simultaneously?

- (2) Conventional sparse coding methods aim to minimize the quantization error via lasso-based approaches which imposes an l_1 penalty with a fixed regularization parameter λ to control the sparsity of the coefficients. While the optimized solutions can be obtained, finding the lasso-based solutions remain a computational task. Hence, is it possible to have an efficient sparsity estimation solution and further to have an adaptive and automatic mechanism for parameter selection?
- (3) Marginal regression-based methods have been shown to be efficient for visual word selection and sparsity estimation. However, it may cause a large quantization error since marginal regression works well when visual words have low correlation in the dictionary. Unfortunately, in general, they are highly correlated and thus may fail to capture the relationship between visual words and input features. Therefore, is it possible to have an efficient feature coding method?

- **From the viewpoint of visual task:**

- (1) The problem of human activity recognition has become an active research topic in computer vision and pattern recognition over the past two decades. A large number of human activity recognition algorithms have been proposed [124–128], and the common aim of these methods is to

model dynamic action patterns for activity representation. Different from the conventional human activity recognition, which recognizes different types of activities such as walking, running and so on, in many potential applications such as intelligence surveillance, public safety and video annotation, it is desirable to recognize people performing activity at a distance without a clear view of their faces. To address this, gait-based human recognition identifies people in the way they walk [129–140], and some encouraging results have been obtained. However, considering only the walking activity in human gait analysis may not be sufficient in some real world applications. Moreover, these methods are not likely to work well when people perform other activities. Different from face and gait recognition [136, 141], how to identify people from their activities? How to encode these dynamic patterns for recognition?

- (2) Conventional activity-based human identification methods usually assume that there exists only a global geometrical structure available in the feature space. Is this assumption reasonable? The intrinsic structure of a video, e.g., the local geometrical structure of image frames within a video, is ignored. This is very important for the activity-based person identification task since it aims to characterize dynamic patterns performed by the same person. Is it possible to exploit the intrinsic structure of a video? How to build video manifolds for an activity performed by a person and how to encode features into sparse representation considering other features within and between manifolds? How to construct hierarchical manifolds for a supervised model?
- (3) Besides the specific task above, is it possible to have a generic feature quantization framework with an efficient solution for common visual tasks

such as image classification, scene classification, action recognition and so on?

These are the questions that the thesis aims to address.

1.3 Contributions of the Thesis

Aiming at addressing the aforementioned problems and designing more robust and efficient algorithms for feature coding, we propose in this thesis several enhanced methods such as graph-based sparse coding and embedding (GSCE), sparse manifold coding (SMC) and marginal-lasso coding (MLC). The improvement comes from efficient sparsity estimation algorithms, adaptive and automatic regularization parameter selection, subspace feature coding, manifold feature coding and dictionary learning. The main contributions of the thesis are summarized below.

- (1) We propose a novel graph-based sparse coding and embedding (GSCE) method by integrating dimension reduction and graph-based sparse coding into a framework for activity-based human identification. We learn a compact feature subspace and a low-dimensional dictionary to better preserve the similarity information between features and expedite the training procedure. Therefore, visual signals can be mapped to the new subspace via the projection matrix and then be quantized into sparse representation via the learned low-dimensional dictionary such that better discriminative information is exploited for classification. Experimental results on three databases are presented to show the efficacy of the proposed method.

- (2) We shall refer to the local geometrical structure of image frames within a video as the intrinsic structure of a video because features extracted from these successive frames usually lie in a manifold. This is very important for activity-based person identification since it aims to characterize dynamic patterns performed by a person so that his/her identity can be recognized via his/her behavior. While a number of sparse coding methods have been proposed over the past few years and some of them consider the geometrical structure of the data space, there is little work considering the dependency of local features in/between the manifolds. To address this, we propose a sparse manifold coding (SMC) model to explicitly utilize the dynamic information of a person in the video-specific manifold, and simultaneously consider the global geometrical structure of the data space between multiple manifolds. We also propose a NN-based sparsity approximation method to encode new data which can easily trace the geometrical structure in the training space to speed up the feature quantization procedure. In addition, we present a supervised sparse manifold coding (SSMC) model by constructing an additional class-based manifold structure. Experiments on two publicly available databases and the comparison with the state-of-the-art methods show the efficacy of the proposed approach.
- (3) While there is a lot of work focusing on sparse coding and dictionary learning, none of them has exploited the advantages of the marginal regression and the lasso simultaneously to provide more efficient and effective solutions. In our work, we provide such an approach with a sound theoretical grounding. We present a generic optimization framework for efficient feature quantization using sparse coding which can be applied to many computer vision tasks such as image classification, action recognition, activity-based human identification,

etc. We also evaluate the performance of the proposed framework on several datasets including Scene-15, Caltech-101, Caltech-256, USPS, KTH, Weizmann and MOBISERV-AIHA. Furthermore, the computational complexity of the proposed method can be two orders faster than that of the lasso with acceptable quantization error. On the other hand, the proposed method is more robust than the conventional marginal regression based methods. We also provide an adaptive and automatic regularization parameter selection scheme and a dictionary learning method incorporated with the proposed sparsity estimation algorithm. Experimental results and detailed model analysis are presented to demonstrate the efficacy of our proposed methods.

1.4 Organization of the Thesis

This thesis is organized as follows: Chapter 1 presents the background, motivations and the contributions of this thesis. Chapter 2 reviews the literature and introduce the background knowledge closely related to this thesis. Chapters 3-5 present the graph-based sparse coding and embedding (GSCE) method, the sparse manifold coding (SMC) algorithm and the marginal-lasso coding (MLC) method, respectively, for image classification, action recognition and activity-based human identification. Chapter 6 presents conclusions and future work.

Figure 1.3 shows the structure of the sparse representation techniques in the thesis. The proposed methods in this thesis purely fall into the sparse coding family. We mainly focus our work on the improvements and the extensions of the general sparse coding in terms of modeling and sparsity estimation. In Chapter 3, we provide a lasso-based solution which learns a compact feature

Figure 1.3: The structure of the models discussed in the thesis. Blocks in orange color are our work.

Figure 1.4: The structure of activity-based human identification discussed in the thesis. Blocks in orange color are our work.

subspace and a low-dimensional over-complete dictionary to encode features into sparse representation maintaining their locality relationship. In Chapter 4, we provide marginal regression-based solutions to unsupervised and supervised cases, respectively, which consider in/between manifolds with unequal weights for features quantization. In Chapter 5, we have an efficient sparsity estimation solution and further have an adaptive and automatic mechanism for regularization parameter selection

The main line of the tasks is activity-based human identification. Figure 1.4 shows the structure of activity-based human identification in the thesis. In Chapter 3, we map human movement features into a low-dimensional space and encode these dynamic patterns considering the geometrical structure for recognition. In Chapter 4, we utilize the temporal information from video clips for feature coding to further improve the performance. In Chapter 5, we provide a generic optimization framework for sparse visual signal representation and apply to several visual tasks to examine its effectiveness including activity-based human identification, handwritten digit recognition, image classification, action recognition.

Chapter 2

Literature Review

In this chapter, we review some representative models related to sparse representation on visual signal applications. We first give a general overview on sparse representation problems. Then we review the state-of-the-art sparse representation models, sparsity estimation algorithms as well as dictionary learning methods. Last we briefly introduce the selected applications and review their corresponding algorithms.

2.1 General Overview on Sparse Visual Signal Representation

The general sparse representation problems include model formulation, sparsity estimation and dictionary learning. Consider a general regression model $x = Us + z$ with a column-wisely normalized dictionary $U = [u_1, \dots, u_p] \in \mathbb{R}^{d \times p}$, a coefficient vector $s \in \mathbb{R}^p$, an input feature $x \in \mathbb{R}^d$, and a noise vector $z \in \mathbb{R}^d$, where d is the feature dimension and p is the number of visual words, respectively. We consider the case that $p \gg d$ which refers to an over-complete dictionary U . The feature

quantization task is to quantize an input feature x into a feature histogram s with a few non-zero elements so that the linear combination of U and s can minimize the quantization error. The mathematical expression can be shown as follows:

$$\min_s \|x - Us\|_2^2 \quad (2.1)$$

Eq. (2.1) is a simple representation model without sparsity constraint and considers a single input x and its corresponding coefficients s . To achieve sparsity, this optimization problem can be solved via sparsity estimation algorithms such as k -means, marginal regression based techniques or the lasso based methods. When considering multiple inputs, the matrix form of Eq. (2.1) can be reformulated as follows:

$$\min_S \|X - US\|_F^2 \quad (2.2)$$

Given a set of features $X = [x_1, x_2, \dots, x_n] \in R^{d \times n}$ where n is the number of samples, we can cluster these features into groups based on a suitable codebook U . $S = [s_1, s_2, \dots, s_n] \in R^{p \times n}$ refers to the set of the corresponding sparse codes for the feature set X .

In the training phase, we need to learn an over-complete dictionary U shown in Eq. (2.1) and Eq. (2.2), which consists of a set of visual words so that each input signal can be linearly represented by a small subset of bases with corresponding weights to minimize the quantization error. Therefore, with a set of training samples, the mathematical expression can be formulated as follows:

$$\begin{aligned} \min_{U, S} \|X - US\|_F^2 \\ \text{subject to } \|u_m\|_2^2 \leq 1, \forall m \end{aligned} \quad (2.3)$$

Each u_m is a unit vector normalized to have a length of one to avoid the scaling problem. In this thesis, we focus on three challenges: sparse representation models,

sparsity estimation and dictionary learning. In the following, we introduce in Chapter 2.2 several well-cited sparse representation models, describe in Chapter 2.3 some state-of-the-art sparsity estimation algorithms and review in Chapter 2.4 dictionary learning methods.

2.2 Sparse Representation Models

Sparse representation model is a feature coding mechanism which decides the way of feature quantization to minimize the quantization error or increase feature discriminative power for final classification and recognition performance. Existing popular sparse representation models can be divided into two categories: region-based models such as k -means and fuzzy c -means, and sparse coding related frameworks.

2.2.1 K-means

k -means [55] is a simple and popular methods for sparse representation. For each input signal x_i , k -means aims to find the nearest cluster center u_m from the dictionary U to minimize the quantization error. The formulation can be defined as follows:

$$\sum_{i=1}^n \min_{m=1,\dots,p} \|x_i - u_m\|_2^2 \quad (2.4)$$

This optimization problem can be reformulated as follows:

$$\begin{aligned} \min_S \sum_{i=1}^n \|x_i - U s_i\|_2^2 \\ \text{subject to } \text{Card}(s_i) = 1, \|s_i\|_1 = 1, s_i \succeq 0, \forall i \end{aligned} \quad (2.5)$$

The constraint means that, for each sparse code s_i , only one element is nonzero and the value of the nonzero element equals to one. In other words, s_i is an indicator showing the index of cluster assignment. k -means is a hard-assignment method that each data point can only be assigned to one cluster. However, this may result in a large quantization error when the input signal lies around the boundary between two clusters or can be represented by more than one visual word. The dictionary U is learned in the training phase with the consideration of the following optimization problem:

$$\begin{aligned} \min_{U, S} \quad & \sum_{i=1}^n \|x_i - U s_i\|_2^2 \\ \text{subject to} \quad & \text{Card}(s_i) = 1, \|s_i\|_1 = 1, s_i \succeq 0, \forall i \\ & \|u_m\|_2^2 \leq 1, \forall m \end{aligned} \tag{2.6}$$

Eq. (2.6) is not convex for U and S simultaneously, but it is convex when one of them is fixed. Therefore, we can iteratively update U and S by averaging intra-class signals as new cluster centers and finding the nearest centers for each signal via Euclidean distance, respectively. For k -means, we need to decide the number of clusters and randomly select an initial set of cluster centers which may lead to different results.

2.2.2 Fuzzy C-means

Like k -means, Fuzzy c-means (FCM) [56, 57] is also a region-based clustering method, but it relaxes the hard assignment condition to soft assignment. Each input signal can be represented by more than one visual word according to the

distance from each cluster center to the signal. The sparse representation model can be formulated as follows:

$$\sum_{i=1}^n \sum_{m=1}^c s_{im} \|x_i - u_m\|_2^2 \quad (2.7)$$

where c is the number of clusters and $s_i = [s_{i1}, \dots, s_{ic}]$ is a set of weights that indicate the probability of signal x_i belonging to a certain cluster m . We can calculate s_{im} using the following equation:

$$s_{im} = \frac{1}{\sum_{j=1}^c \left(\frac{\|x_i - u_m\|_2^2}{\|x_i - u_j\|_2^2} \right)^{\frac{2}{q-1}}} \quad (2.8)$$

where $q > 1$ is a fuzzication parameter controlling the sparsity of vector s_i . The same as k -means, we can learn U and S iteratively by calculating the weight centers and updating sparse codes. We also need to predefine the number of clusters and select an initial set of cluster centers which may result in different performance.

2.2.3 Sparse Coding

Different from Eq. (2.3) which does not consider the sparse constraint directly in the equation, Tibshirani [77] proposed a Least Absolute Shrinkage and Selection Operator method, called Lasso in short, which exploits an l_1 -norm penalty for the sparsity control so that coefficients can be shrunk towards zero automatically and can be defined by the following optimization problem:

$$\begin{aligned} \min_{U, S} \quad & \sum_{i=1}^n \|x_i - U s_i\|_F^2 + \lambda \|s_i\|_1 \\ \text{subject to} \quad & \|u_m\|_2^2 \leq 1, \forall m \end{aligned}$$

where $\lambda \geq 0$ is a regularization parameter which controls the sparsity of the coefficients s_i . In other words, λ controls the amount of shrinkage: the larger the

value of λ , the greater the amount of shrinkage. That is, the coefficients of s_i are shrunk toward zero so that there exist a few nonzero elements. An equivalent way to write the lasso problem is:

$$\begin{aligned} \min_{U, S} \sum_{i=1}^n \|x_i - U s_i\|_F^2 \\ \text{subject to } \|u_m\|_2^2 \leq 1, \forall m \\ \|s_i\|_1 \leq E, \forall i \end{aligned} \quad (2.9)$$

where $E \geq 0$ is an explicit energy constraint on the parameters. The matrix form of Eq. (2.9) can be written as follows:

$$\begin{aligned} \min_{U, S} \|X - US\|_F^2 + \lambda \sum_{i=1}^n \|s_i\|_1 \\ \text{subject to } \|u_m\|_2^2 \leq 1, \forall m \end{aligned} \quad (2.10)$$

In order to minimize an estimate of expected reconstruction or quantization error, λ and E should be chosen adaptively such as $\lambda_1, \dots, \lambda_n$ and $E_1, \dots, E_n$ for each individual signal $x_1, \dots, x_n$, respectively. However, in practice, they usually set as a fixed value for all inputs. While Eq. (2.6) is not convex for U and S simultaneously, it is convex when one of them is fixed. Therefore, we can learn the dictionary by iteratively updating the learning information using the two following steps:

Step I: Obtain sparse codes via the latest updated dictionary

$$\min_S \|X - US\|_F^2 + \lambda \sum_{i=1}^n \|s_i\|_1 \quad (2.11)$$

Step II: Learn a new dictionary via sparse codes obtained from Step I

$$\begin{aligned} \min_U \|X - US\|_F^2 \\ \text{subject to } \|u_m\|_2^2 \leq 1, \forall m \end{aligned} \quad (2.12)$$

The optimization in Eq. (2.11) can be solved by least angle regression method [79] or feature-sign search algorithm [83]. The dictionary U in Eq. (2.12) can be optimized using conjugate gradient descent method. We will review the sparsity estimation approaches and dictionary learning algorithms in Chapter 2.3 and Chapter 2.4, respectively.

Different from region-based approaches such as k -means and fuzzy c -means, The basic idea of SC is to find a small set of bases so that the linear combination of these selected bases can quantize the original signal with a minimum quantization error.

2.2.4 Laplacian Sparse Coding and Graph Regularized Sparse Coding

Since the quantization process of SC only considers individual signal, it may cause similar features to be quantized into different visual words due to the sensitivity of the visual word selection from the over-complete dictionary. To overcome this shortcoming, some researchers have proposed graph-based sparse coding approaches such as Laplacian sparse coding (LSC) [90, 91] and graph regularized sparse coding (GraphSC) [59] to further improve the representation performance. The aim of graph-based sparse coding method is to impose local structure into the sparse coding optimization framework so that similar local features can be quantized into similar visual words to obtain similar sparse codes. The objective function of LSC and GraphSC is defined as follows:

$$\begin{aligned} \min_{U, S} \|X - US\|_F^2 + \lambda \sum_{i=1}^n \|s_i\|_1 + \frac{\gamma}{2} \sum_{i=1}^n \sum_{j=1}^n \|s_i - s_j\|^2 W_{ij} \quad (2.13) \\ \text{subject to} \quad \|u_m\|_2^2 \leq 1, \forall m = 1, \dots, k \end{aligned}$$

where $\gamma \geq 0$ is a weighting parameter and W is a similarity matrix indicating the local geometrical structure. The similarity matrix can be defined in different ways

such as k nearest neighbors used in GraphSC or histogram intersection used in LSC. Since the third term can be rewritten by defining the Laplacian matrix as $L = D - W$ [142–144], Eq. (2.13) can be reformulated as:

$$\min_{U, S} \|X - US\|_F^2 + \lambda \sum_{i=1}^n \|s_i\|_1 + \gamma \text{tr}(SL S^T) \quad (2.14)$$

where $D = \text{diag}(d_1, \dots, d_n)$ is a degree matrix that $d_i = \sum_{j=1}^n W_{ij}$. Like SC, the objective function is convex when U or S is fixed. Therefore, the solution can be obtained by the iterative learning procedure. To learn sparse codes, the problem in Eq. (2.13) becomes

$$\min_S \|X - US\|_F^2 + \lambda \sum_{i=1}^n \|s_i\|_1 + \gamma \text{tr}(SL S^T) \quad (2.15)$$

The solution can be obtained by following the feature-sign search [83] method. The dictionary optimization problem becomes as follows and can be solved using conjugate gradient descent method [83]:

$$\min_U \|X - US\|_F^2$$

2.2.5 Smooth Sparse Coding

While LSC or GraphSC is one of the most effective feature quantization approach for image classification, time complexity remains a problem since the training procedure falls under a convex lasso sub-problem. Similar to LSC and GraphSC, the basic idea of smooth sparse coding (SSC) is to preserve the intrinsic geometry structure of original data by incorporating similarity function in feature space into sparse coding. The objective function of SSC can be formulated as follows:

$$\begin{aligned} \min_{U, S} \sum_{i=1}^n \sum_{j=1}^n w(x_j, x_i) \|x_j - U s_i\|_2^2 \\ \text{subject to } \|u_m\|_2^2 \leq 1, \forall m = 1, \dots, k \\ \|s_i\|_1 \leq E, \forall i = 1, \dots, n \end{aligned} \quad (2.16)$$

Table 2.1: Categorization of sparse representation methods. (“*”the proposed methods)

Methods	Sparse representation algorithms
Region-based	k -means, FCM
Sparse coding	SC, LSC, GraphSC, SSC, *GSCE, *SMC, *MLC

Different from LSC or GraphSC, SSC do not deal with the lasso optimization problem for sparse code generation. Instead, SSC encodes features using the marginal regression approach [87] which leads to a dramatic speedup for large dictionary sizes. Eq. (2.16) can be solved based on alternating procedures of sparse coding and dictionary learning which use the marginal regression solution and the gradient projection descent method [87] to obtain S and U , respectively.

2.2.6 Summary

We summarize the aforementioned sparse representation models in Table 2.1, and include the proposed methods with star marks to show their similarity and difference. According to the sparse representation methods, we can divide them into two categories: region-based methods and sparse coding approaches. k -means and FCM are region-based methods clustering features based on the distances between clusters to each feature. On the other hand, SC, LSC, GraphSC and SSC are in the sparse coding family. Such kind of methods usually exploit a linear combination of basic patterns or so-called visual words to quantize input features based on the goal of minimizing the quantization error. We propose in this thesis three major models such as GSCE (Chapter 3), SMC (Chapter 4) and MLC (Chapter 5) fall into the category of sparse coding family as well.

Table 2.2: Categorization of sparse coding methods according to different data structures. (“*” the proposed methods)

Methods	Sparse representation algorithms
Independent	SC, *MLC
Geometric structure	LSC, GraphSC, SSC, *GSCE, *SMC

In the sparse coding family, we can further classify these methods into two types based on the data structure used in the sparse representation models. Table 2.2 shows that LSC, GraphSC, SSC, GSCE and SMC exploit the geometrical structure between features, but SC and MLC do not.

2.3 Sparsity Estimation Algorithms

We present two state-of-the-art sparsity estimation algorithms, feature-sign search method [59, 83, 89, 90] and marginal regression approach [87, 88, 145] used in the aforementioned models.

2.3.1 Feature-sign Search

The feature-sign search method is a popular and efficient solution for sparsity estimation which solves the general sparse coding equation of Eq. (2.11). It continuously selects and updates potential candidates in an active set to maintain the nonzero coefficients and their corresponding signs until the constraint is met. The detailed algorithm is listed in Algorithm 1. However, with a large dictionary size, the active set would become large and the feature-sign searching process may

be hard to terminate [145, 146]. While the feature-sign search aims to obtain a small quantization error, it has a large computational cost [87, 88].

Algorithm 1 Feature-sign search algorithm

Require: Given a set of feature points $X = [x_1, \dots, x_n] \in R^{d \times n}$, a dictionary $U = [u_1, \dots, u_p] \in \mathbb{R}^{d \times p}$ and a predefined regularization parameter λ .

Ensure: A set of sparse code $S = [s_1, \dots, s_n] \in \mathbb{R}^{p \times n}$ for features $X = [x_1, \dots, x_n]$

- 1: Initialization: For $x_i \forall i$, zero coefficients of s_i and an empty active set A
- 2: Select $m = \arg \max_m \left| \frac{\partial \|x_i - U s_i\|_2^2}{\partial s_i^{(m)}} \right|$ from zero coefficients of s_i

$$\text{if } \left| \frac{\partial \|x_i - U s_i\|_2^2}{\partial s_i^{(m)}} \right| > \lambda, \text{ then set } \vartheta_m = -1, A = \{m\} \bigcup A \quad (2.17)$$

$$\text{if } \left| \frac{\partial \|x_i - U s_i\|_2^2}{\partial s_i^{(m)}} \right| < -\lambda, \text{ then set } \vartheta_m = 1, A = \{m\} \bigcup A \quad (2.18)$$

- 3: Feature-sign step

Let $\hat{U}$ be a sub-matrix of U containing the corresponding columns in A .
 Let $\hat{x}$ and $\hat{\vartheta}$ be sub-vectors of x and ϑ , respectively, corresponding to A .
 Compute the solution to $\min_{\hat{s}_i} \|x_i - U \hat{s}_i\|_2^2 + \lambda \hat{\vartheta}^T \hat{s}_i$

$$s_i^{\hat{new}} = (\hat{U}^T \hat{U})^{-1} (\hat{U}^T x_i - \lambda \frac{\hat{\vartheta}}{2}) \quad (2.19)$$

Perform a discrete line search from the closed line segment $\hat{s}_i$ to $s_i^{\hat{new}}$.
 Check the objective value at $s_i^{\hat{new}}$ and all coefficients where any sign change.
 Update $\hat{s}_i$ to the point with the lowest objective value.
 Remove zero coefficients of $\hat{s}_i$ from A
 Update $\vartheta = \text{sign}(s_i)$ where $\vartheta_m = \{-1, 0, 1\}$ denotes $\text{sign}(s_i^{(m)})$

- 4: Check the optimality condition

(a) For nonzero coefficients: $\frac{\partial \|x_i - U s_i\|_2^2}{\partial s_i^{(m)}} + \lambda \text{sign}(s_i^{(m)}) = 0, \forall s_i^{(m)} \neq 0$

Go to Step 3 if condition (a) is not satisfied; else check condition (b).

(b) For zero coefficients: $\left| \frac{\partial \|x_i - U s_i\|_2^2}{\partial s_i^{(m)}} \right| \leq \lambda, \forall s_i^{(m)} = 0$

Go to Step 2 if condition (b) is not satisfied; otherwise return s_i as the solution.

2.3.2 Marginal Regression

Traditional sparsity estimation techniques such as the feature-sign search algorithm estimate the sparse code by continuously selecting variables to the active set based on the searching criteria. While low quantization error can be achieved, it is computational expensive. Thus, an old method, marginal regression approach has been revisited recently due to its efficiency. Marginal regression considers the following equation:

$$\min_s \|x - Us\|_2^2 \quad (2.20)$$

To estimate the sparse code s , we first compute the component-wise marginal regression coefficients $\hat{a}$:

$$\hat{a} \equiv U^T x \quad (2.21)$$

where $\hat{a}^{(m)} = u_m^T x$ is the m th element of $\hat{a}$. Then, we threshold the coefficients in terms of their absolute values using a tuning parameter $t > 0$ [88, 147] so that the coefficients whose absolute values are larger than t are kept and others are set to zero. The mathematical expression can be written as follows:

$$\hat{s}^{(m)} = \begin{cases} \hat{a}^{(m)} & \text{if } |\hat{a}^{(m)}| > t \\ 0 & \text{otherwise} \end{cases} \quad (2.22)$$

According to Eq. (2.22), however, utilizing a fixed tuning parameter may cause a large quantization error since it is not always a good threshold to fit each feature well. Instead of threshold selection, the sure independence screening (SIS) [87] method selects the top k large coefficients in terms of their absolute values whose l_1 -norm is bounded by a constraint E [87, 145]. The equation can be reformulated as follows:

$$\hat{s}^{(m)} = \begin{cases} \hat{a}^{(m)} & \text{if } m \in Q \\ 0 & \text{otherwise} \end{cases} \quad (2.23)$$

where $Q = \{J_1, \dots, J_r : r \leq p : \sum_{z=1}^r |\hat{a}_i^{(J_z)}| \leq E\}$. Therefore, each sparse code can be represented under the sparsity energy bound. The detail is summarized in Algorithm 2. It is worth mentioning that Eq. (2.22) and Eq. (2.23) are hard-thresholding-type methods without a theoretical grounding. While marginal regression approaches such as SIS or purely hard-thresholding technique have shown their efficiency, they work well when the columns of the dictionary have low correlation [145, 146]. However, in general, the columns of the over-complete dictionaries are highly correlated, and thus, in a linear regression model, their coefficients can become poorly determined and exhibit high variance.

2.3.3 Summary

We summarize in Table 2.3 the aforementioned sparsity estimation approaches. The computational complexity of the marginal regression methods can be two orders faster than that of the lasso methods by allowing sacrificing the inevitable quantization error. Our marginal-lasso coding method can achieve a smaller quantization error without sacrificing computational complexity. We will introduce the marginal-lasso framework in Chapter 5.

Table 2.3: Summary of the different sparsity estimation approaches in terms of the quantization error and the computational complexity. (“*” the proposed methods)

Methods	Quantization error	Computational complexity
Lasso-based method	Small	$O(p^3 + dp^2)$ [88]
Marginal regression	Large	$O(dp)$ [88]
*Marginal-Lasso	Medium-Small	$O(dp)$

In Table 2.4, we classify aforementioned sparse representation models according

Algorithm 2 Sure independence screening algorithm

Require: Given a set of feature points $X = [x_1, \dots, x_n] \in R^{d \times n}$, a dictionary $U = [u_1, \dots, u_p] \in \mathbb{R}^{d \times p}$, and a predefined regularization parameter t

Ensure: A set of sparse code $S = [s_1, \dots, s_n] \in \mathbb{R}^{p \times n}$ for features $X = [x_1, \dots, x_n]$

1: For $x_i \forall i$, compute marginal regression coefficients

$$\hat{a}_i^{(m)} = \frac{u_m^T x_i}{\|u_m\|_2^2} \quad \forall m = 1, \dots, p \quad (2.24)$$

2: Sorting: Sort the elements of $|\hat{a}_i|$ in a descending order.

$$\tilde{a}_i = [\hat{a}_i^{(J_1)}, \dots, \hat{a}_i^{(J_p)}], \text{ where } |\hat{a}_i^{(J_1)}| > |\hat{a}_i^{(J_2)}|, \dots, > |\hat{a}_i^{(J_p)}| \quad (2.25)$$

where $|\hat{a}_i^{(J_z)}|$ is the z th largest coefficients among $|\hat{a}_i|$.

3: Selecting: Select the top j coefficients whose l_1 norm is bounded by E

$$Q = \{J_1, \dots, J_r : r \leq p : \sum_{z=1}^r |\hat{a}_i^{(J_z)}| \leq E\} \quad (2.26)$$

4: Thresholding

$$s_i^{(m)} = \begin{cases} \hat{a}_i^{(m)} & \text{if } m \in Q \\ 0 & \text{otherwise} \end{cases} \quad (2.27)$$

to different sparsity estimation approaches. For example, SC, LSC, GraphSC and *GSCE estimate sparse codes by solving the lasso problem, while SSC and *SMC use marginal regression to obtain sparse codes. *MLC is the proposed model which can be solved using the proposed marginal-lasso coding framework.

Table 2.4: Categorization of sparse coding methods according to different sparsity estimation approaches. (“*” the proposed methods)

Methods	Sparse representation algorithms
Lasso-based method	SC, LSC, GraphSC, *GSCE
Marginal regression	SSC, *SMC
*Marginal-Lasso	*MLC

2.4 Dictionary Learning Methods

Dictionary learning plays an important role in the training phase. According to the aforementioned sparse coding methods, we review two popular dictionary learning approaches conjugate gradient descent and gradient projection descent.

2.4.1 Conjugate Gradient Descent

When the sparse coefficients S is fixed, the general sparse coding equation with respect to U can be written as follows:

$$\begin{aligned} \min_U \|X - US\|_F^2 \\ \text{subject to } \|u_m\|_2^2 \leq 1, \forall m \end{aligned} \tag{2.28}$$

This is a convex optimization problem and can be solved by the conjugate gradient descent based on its Lagrange dual [83]. Let $\delta = [\delta_1, \dots, \delta_p]$ and $\delta_m \geq 0$ be the

Lagrange multiplier of the m th inequality constraint $\|u_m\|_2^2 - 1 \leq 0$. The Lagrange function of Eq. (3.12) can be defined as follows:

$$\begin{aligned}
 L(U, \delta) &= \|X - US\|_F^2 + \sum_{m=1}^p \delta_m (\|u_m\|_2^2 - 1) \\
 &= \|X - US\|_F^2 + \text{tr}(U^T U \delta) - \text{tr}(\delta) \\
 &= \text{tr}(X^T X) - 2\text{tr}(U^T X S^T) + \text{tr}(S^T U^T U S) + \text{tr}(U^T U \delta) - \text{tr}(\delta)
 \end{aligned} \tag{2.29}$$

Therefore, we can have the optimal solution U^* using conjugate gradient as follows:

$$U^* = X S^T (S S^T + \delta)^{-1} \tag{2.30}$$

2.4.2 Gradient Projection Descent

The dictionary U is a set of normalized visual words denoting by each column u_m . To learn the dictionary, we iteratively optimize U and S by obtaining S with a fixed U and updating U based on a given S . The optimization problem of the dictionary learning can be formulated as follows:

$$\begin{aligned}
 \min_U F(U) &= \min_U \|X - US\|_F^2 \\
 \text{subject to } &\|u_m\|_2^2 \leq 1, \forall m
 \end{aligned} \tag{2.31}$$

This optimization problem can be solved via the first-order gradient descent method [145, 146]:

$$U_{q+1} = \Pi_U \{U_q - \beta \nabla F(U_q)\} \tag{2.32}$$

where

$$\nabla F(U_q) = 2(U_q S S^T - X S^T) \tag{2.33}$$

$\nabla F(U_q)$ is the gradient of $F(U)$ with respect to U at each iteration q , parameter β is the step size, and Π_U is the projection function which maps each column u_k

to the L_2 -norm unit ball. More specifically, the dictionary is learned iteratively by finding the local minimum along the gradient direction based on the small step size and normalizing each column vector with length smaller or equal to one until convergence.

2.4.3 Summary

We summarize in Table 2.5 the aforementioned dictionary learning approaches. To learn a suitable dictionary in the training phase, SC, LSC and GraphSC use the conjugate gradient descent method and SSC use the gradient projection descent approach. On the other hand, some researchers [148] have pointed out that using sparse coding as an encoder, the choice of visual words of a dictionary has little effect on performance. Therefore, we adopt online dictionary learning algorithm [146] to obtain an initial dictionary as the warm start since it has demonstrated its efficiency in terms of speed and is faster than batch alternatives methods such as the feature-sign search method [83].

Table 2.5: Categorization of sparse representation methods according to the dictionary learning approach. (“*” the proposed methods)

Methods	Sparse representation algorithms
Conjugate Gradient Descent	SC, LSC, GraphSC, *GSCE
Gradient Projection Descent	SSC, *SMC, *MLC

2.5 Selected Applications

We focus on feature coding methods in this thesis and apply the proposed algorithms to four visual tasks in the computer vision and pattern recognition area.

They are activity-based human identification, handwritten digit recognition, image classification and action recognition. We briefly review each of them in the following.

2.5.1 Activity-based Human Identification

The problem of activity-based human identification is relative new and has been investigated in the literature [149–151] in recent year. The task aims to identify people when they are performing some activities such as running, handwaving, eating, drinking and so on. Hence, gait recognition, which aims to recognize the identity of people by the way they walk, can be considered as a subproblem of activity-based human identification. How to encode dynamic patterns for recognition is the main challenge of the problem. Gkalelis *et al.* [149] and Iosifidis *et al.* [150] are the pioneers working on this problem. They encoded dynamic information by utilizing a fuzzy c-means (FCM) as vector quantization which is a region-based soft assignment. Then each video clip containing an activity performed by a person was represented using a mean pooling method followed by a linear discriminant analysis (LDA) based algorithm. They adopted two-step classification. The first classifier is an action classifier for movement recognition and then a movement specific person classifier was used for human identification. The work in [151] argued that discriminative information may be lost due to the quantization error caused by FCM. Hence, they proposed using sparse coding to encode dynamic information, followed by discriminative metric learning to further improve the discriminative power. While some encouraging results have been obtained, there are still some limitations which can be further improved for feature coding and dictionary learning for activity-based human identification. There are two more relevant works [152, 153] published in 2013 and 2014, respectively. One

[152] adopted a self-organizing Neural Network to determine a topographic map of representative human body poses, used fuzzy vector for quantization, and trained two feedforward Artificial Neural Networks for classification. The other work [153] used discriminative sparse projections and learned an orthogonal ensemble metric for recognition. This application can be found in Chapter 3, 4 and 5.

2.5.2 Handwritten Digit Recognition

The handwritten digit recognition problem has been investigated in the past decades [59, 154–159]. Hidden Markov Model (HMM) is one of the popular approach for handwritten pattern recognition systems [155–157]. Besides HMM, Labusch *et al.* [158] learned feature basis using sparse coding and applied a local maximum operation for local shift invariance. Zheng *et al.* [59] used graph regularized sparse coding method for handwritten digit representation. In Chapter 5, we propose a new feature coding algorithm, and test our algorithm on the benchmark USPS handwritten digits dataset [59, 159] for handwritten digit recognition.

2.5.3 Image Classification

One of the critical challenges of image classification is image representation. Bag-of-features (BoF) methods [3, 5, 21, 22, 31, 32, 38, 45], which consider an image as a collection of unordered visual words, have shown their promising performance in this task. Since the BoF model do not utilize spatial information of an image, Lazebnik *et al.* [4] proposed a spatial pyramid matching (SPM) method to recognize natural scene categories by repeatedly partitioning an image into sub-regions and computing histograms of local features over the resulting sub-regions, and has obtained exciting

results. Then, Yang *et al.* [89] proposed to encode SIFT from k-means vector quantization (VQ) in the conventional SPM to sparse coding (SC) denoted by ScSPM, and adopt max spatial pooling method for final image representation to further improve the classification performance. As general sparse coding do not consider the relationship between local features causing similar local features may be quantized into different sparse codes, more recent work has considered geometrical structure to encode local features into the formulation and derived many feature coding algorithms, such as Laplacian Sparse Coding (LSC) [90, 91], Graph regularized Sparse Coding (GraphSC) [59] and Smooth Sparse Coding (SSC) [87], to enhance image classification performance. In Chapter 5, we examine our proposed method on three public datasets, Scene 15, Caltech 101 and Caltech 256 for image classification. More, we provide two comparisons in terms of speed and quantization error between different sparsity estimation methods, and further discuss parameter selection of the proposed method on Caltech 101 dataset.

2.5.4 Action Recognition

Action recognition is a very active research topic in computer vision and pattern recognition [12, 160–174]. Cuntoor *et al.* [160] proposed to use the hidden Markov model (HMM) framework to represent events with significant changes in time and space, and represent activities as a sequence of such events. Yao *et al.* [161] proposed to learn sparse bases of action attributes and parts for human action recognition in still images. Ji *et al.* [167] developed a 3D convolution neural networks for action recognition (CNN). Wang *et al.* [12, 164–166] evaluated the effectiveness of a set of local features for action recognition and proposed dense trajectories and improved dense trajectories methods for action representation. Wu *et al.* [168] learned a global

Gaussian mixture model (GMM) as a universal background model, represented each video as a video-specific GMM, and then used multiple GMMs to represent the distributions of local spatio-temporal context between interest points at different scales.

While many features are proposed to describe dynamic action patterns and represent action events, the work [12,172] has similar conclusions that MBH features are enough for basic action recognition for state-of-the-art performance. On the other hands, SIFT and MFCC features provide complementary cues for complex events. Therefore, in Chapter 5, we compare our proposed feature coding method with other existing coding methods such as k-means, feature-sign search and marginal regression based approaches using space-time interest points (STIP) as the feature detector and using three different feature descriptors, histogram of gradient orientations (HOG), histogram of optical flow (HOF) and HOG/HOF, which is a histogram concatenating HOG and HOF together, on the KTH action recognition dataset. We provide two comparisons in terms of speed and quantization error between different sparsity estimation methods, and further discuss parameter selection of the proposed method on Caltech 101 dataset. We not only provide recognition accuracy but also discuss sparsity analysis, speed comparison and quantization error between different methods.

Chapter 3

Graph-based Sparse Coding and Embedding

3.1 Introduction

Activity-based human identification is a relative new problem investigated in the literature [149–151] which aims to identify people from their activities such as running, handwaving, eating and drinking where walking is one of the cases, and thus, gait recognition can be seen as a subproblem of activity-based human identification. The challenge of the problem is to find an effective representation of the dynamic motion patterns from multiple activities performed by an individual for human recognition. Gkalelis *et al.* [149] and Iosifidis *et al.* [150] are the pioneers working on this problem. They encoded dynamic information by utilizing a fuzzy c-means (FCM) vector quantization which is a region-based soft assignment. Then each video clip containing an activity performed by a person was represented using a mean pooling method followed by a linear discriminant analysis (LDA) based algorithm. They adopted two-step classification. The first classifier is an action classifier for movement recognition and then a movement specific person classifier

was used for human identification. The work in [151] argued that discriminative information may be lost due to the quantization error caused by FCM. Hence, it proposed using the sparse coding method to encode dynamic information, followed by discriminative metric learning to further improve the discriminative power. While some encouraging results have been obtained, there are still some limitations which can be further improved for feature coding and dictionary learning for activity-based human identification:

- (1) Most existing sparse coding algorithms assume that there exists an over-complete dictionary so that sparse representation can be achieved by linearly combining a small number of visual words chosen from the over-complete dictionary. However, original feature data may be of high dimension, which give rise to the difficulty of learning a large over-complete dictionary due to high computational complexity. For example, for the activity-based human identification task, most previous work deals with feature coding in the original data space, but it usually incurs high computational complexity on the dictionary learning process, especially when learning an over-complete dictionary. Hence, it is better to transform these high-dimensional data into a compact low-dimensional feature space such that the dictionary learning process can be expedited in the low-dimensional feature space. Moreover, the intrinsic characteristics of the original data can be well-preserved, and redundancy information can be also removed in the low-dimensional feature space to further improve the performance.
- (2) For feature quantization using sparse coding, some similar patterns may be encoded into different visual words in the encoding process due to the sensitivity of the sparse code generation. This inconsistency of data

representation between features may result in deterioration of recognition performance. Therefore, it is necessary to preserve intrinsic local geometrical structure of the data space so that similar features may obtain similar visual words under the feature coding process.

Motivated by the above observations, we propose a novel graph-based sparse coding and embedding (GSCE) method for activity-based human identification, integrating dimension reduction and graph-based sparse coding into the general sparse coding framework for feature quantization.

Our contributions in this chapter include the following:

- (1) We jointly learn a feature projection matrix and a dictionary, where each frame is projected into a low-dimensional subspace and encoded with the consideration of its neighborhood coding coefficients so that similar features can be obtained similar coefficients in the low-dimensional subspace. By doing so, we simultaneously solve two problems mentioned above, high-dimensionality and inconsistency between coding coefficients.
- (2) We propose a new simultaneous feature and dictionary learning method by considering a subspace learning technique and a graph-based embedding into the general sparse coding framework, and providing an optimized solution for the subspace projection, sparsity estimation and dictionary learning for activity-based human identification.

The rest of this chapter is organized as follows. Section 3.2 details the proposed graph-based sparse coding and embedding approach with an objective function,

optimal solutions and algorithms for activity-based human identification. Section 3.3 demonstrates the effectiveness of the proposed method, and Section 3.4 summarizes the chapter.

3.2 Graph-based Sparse Coding and Embedding

3.2.1 Motivation

To encode dynamic patterns of human activities performed by different persons, Lu *et al.* [151] used sparse coding to quantize input features into a set of visual words as follows:

$$\min_{s_i} \|x_i - U s_i\|_2^2 + \lambda \|s_i\|_1 \quad (3.1)$$

where $x_i \in R^d$ is a d -dim input feature to be encoded into a vector of sparse coefficients $s_i \in R^p$ using a learned dictionary $U = [u_1, u_2, \dots, u_p] \in R^{d \times p}$ by minimizing the quantization error. In Eq. (3.1), the first term is the quantization error, and the second term is the sparsity term with a regularization parameter, $\lambda \geq 0$, which controls the sparsity of s_i . According to sparse coding, we can quantize x_i by selecting a small subset of visual words with corresponding weighting coefficients bounded by l_1 -norm penalty defining as $\|s_i\|_1 = \sum_{m=1}^p |s_i^{(m)}|$ and let others to be zero so that their linear combination can minimize the quantization error and simultaneously improve the discriminative power for recognition. This is a soft assignment technique where each feature can be quantized into several visual words instead of one.

While sparse coding has achieved some encourage results on feature quantization, there are two limitations we state in the following. First, it usually assumes that input features lie in the low-dimensional feature space. However, this may not be

always true, and when input features lie in the high-dimensional feature space, it may cause serious computational problems, especially when learning the dictionary U . Second, Eq. (3.1) solves each s_i one at a time and does not consider the structure of neighborhood which may easily cause similar features with difference coding coefficients due to its over-complete dictionary and deteriorate the performance of recognition. This problem is also mentioned in the work of Laplacian sparse coding and graph regularized sparse coding [59, 90, 91], but they do not take the problem of dimensionality into consideration.

To address the limitations mentioned above, we propose to learn a projection matrix which maps input features from high dimensionality to a lower one and simultaneously learn a dictionary in the lower-dimensional subspace with the consideration of its geometrical structure so that similar features hold their similarity conditions in the subspace, and in the coding space.

3.2.2 Proposed Approach

To encode high dimensional dynamic patterns of human activities performed by different persons for further recognition of the person identity, we propose a graph-based sparse coding and embedding (GSCE) method for feature quantization which jointly learns a projection matrix and a dictionary matrix so that each high-dimensional feature can be projected into a low-dimension subspace and encoded into sparse coefficients with the consideration of its neighborhood. By doing so, the discriminative power can be increased for recognition and the computational cost

can be reduced simultaneously. The objective function can be formulated as follows:

$$\begin{aligned}
 \min_{P, U', S} J &= J_1 + \beta J_2 + \frac{\gamma}{2} J_3 \\
 &= \|PX - U'S\|_F^2 + \lambda \sum_{i=1}^n \|s_i\|_1 \\
 &\quad + \beta \|X - P^T P X\|_F^2 \\
 &\quad + \frac{\gamma}{2} \sum_{i=1}^n \sum_{j=1}^n \|s_i - s_j\|_2^2 w(x_i, x_j)
 \end{aligned} \tag{3.2}$$

$$\text{subject to } \|u'_m\|_2^2 \leq 1, \forall m, P P^T = I$$

where $P \in R^{d' \times d}$ is the feature projection matrix which maps high-dimensional data into a low-dimensional subspace, $d > d'$, $U' \in R^{d' \times p}$ is a codebook generated from low-dimensional feature data, $S = [s_1, s_2, \dots, s_n] \in R^{p \times n}$ is the sparse representation of the training samples in $X = [x_1, x_2, \dots, x_n] \in R^{d \times n}$, s_i is the coefficient vector of x_i , λ , β and γ are three regularization parameters accordingly of each term, and $w(\cdot) \in R^{n \times n}$ is a similarity matrix measuring the similarity of the sparse codes s_i and s_j according to their input pairs x_i and x_j in the feature space, where k -nearest neighbors are adopted, which is defined as follows:

$$w(x_i, x_j) = \begin{cases} \exp(-\frac{\|x_i - x_j\|_2^2}{t}) & \text{if } x_i \in N_k(x_j) \text{ or } x_j \in N_k(x_i) \\ 0 & \text{otherwise} \end{cases} \tag{3.3}$$

where $N_k(x_i)$ denotes the k nearest neighbors of x_i , $N_k(x_j)$ denotes the k nearest neighbors of x_j and t is a regularization parameter. Note that, to distinguish from the dictionary U in the original dimensional feature space in this thesis, we denote here the new dictionary in the low-dimensional feature space by $U' \in R^{d' \times m}$.

Eq. (3.2) aims to minimize the quantization error with a sparse constraint, the energy loss caused by dimension transformation as well as the penalty of the

inconsistency between similar features and their corresponding sparse reconstructed coefficients. The first term J_1 in Eq. (3.2) is the quantization error in the low-dimensional feature space, which is to ensure that for each movement sample x_i in the low-dimensional feature subspace, its quantization error is minimized with l_1 -norm penalty. The second term J_2 in Eq. (3.2) is the reduction error between low-dimensional feature approximation and the original feature data which is to ensure that the feature projection matrix P can preserve the energy of each x_i as much as possible. The third term J_3 in Eq. (3.2) is consistency term which is to ensure that the difference of two sparse codes coming from similar movement samples can be minimized so that their geometrical structure can be kept in the subspace.

3.2.3 Proposed Algorithms

Eq. (3.2) is not convex for P , U' and S , simultaneously, but it is convex when two of them are fixed. Therefore, in the training phase, we can iteratively update each variable one at a time. In other words, when P and U are fixed, we can assign a set of visual words with corresponding weighting coefficients by optimizing the following formulation:

$$\min_S \|PX - U'S\|_F^2 + \lambda \sum_{i=1}^n \|s_i\|_1 + \frac{\gamma}{2} \sum_{i=1}^n \sum_{j=1}^n \|s_i - s_j\|_2^2 w(x_i, x_j) \quad (3.4)$$

While the optimization problem in Eq. (3.4) is convex, it is not differentiable at $s_i^{(m)} = 0$ due to the l_1 -norm penalty term. Therefore, we follow the feature-sign search algorithm introduced by [83] to solve this optimization problem. In feature-sign search, each v_i is updated individually, we can rewrite the formulation as follows:

$$\min_S \sum_{i=1}^n \|Px_i - U's_i\|_F^2 + \lambda \|s_i\|_1 + \gamma \sum_{i=1}^n \sum_{j=1}^n L_{ij} s_j^T s_i \quad (3.5)$$

where

$$\begin{aligned}
 \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \|s_i - s_j\|_2^2 w(x_i, x_j) &= \text{tr}(SLS^T) \\
 &= \text{tr}\left(\sum_{i=1}^n \sum_{j=1}^n L_{ij} s_i s_j^T\right) \\
 &= \sum_{i=1}^n \sum_{j=1}^n L_{ij} s_j^T s_i
 \end{aligned} \tag{3.6}$$

where L is the Laplacian matrix defined as $L = D - W$, W consists of similarity scores of all feature pairs, $W_{ij} = w(x_i, x_j)$, and $D_{ii} = \sum_{j=1}^n W_{ij}$. Thus, the optimization problem can be decomposed into p separate optimization tasks:

$$\begin{aligned}
 \min_{s_i} \|Px_i - U' s_i\|_2^2 + \lambda \|s_i\|_1 + \gamma L_{ii} s_i^T s_i + 2\gamma s_i^T \sum_{i \neq j} L_{ij} s_j \\
 = \min_{s_i} f(s_i) + \lambda \|s_i\|_1
 \end{aligned} \tag{3.7}$$

where $f(s_i) = \|Px_i - U' s_i\|_2^2 + \gamma L_{ii} s_i^T s_i + 2\gamma s_i^T \sum_{i \neq j} L_{ij} s_j$. The details of the proposed solution are listed in Algorithm 3.

We can obtain the dictionary U in the training phase when P and S are fixed. The, the objective function can be formulated as follows:

$$\begin{aligned}
 \min_{U'} \|PX - U'S\|_F^2 \\
 \text{subject to } \|u'_m\|_2^2 \leq 1, \forall m = 1, \dots, p
 \end{aligned} \tag{3.11}$$

This is a convex optimization problem and can be solved by the conjugate gradient decent method based on its Lagrange dual [83]. Let $\delta = [\delta_1, \dots, \delta_p]$ and $\delta_m \geq 0$ be the Lagrange multiplier of the m th inequality constraint $\|u'_m\|_2^2 - 1 \leq 0$. The Lagrange function of Eq. (3.11) can be defined as follows:

$$\begin{aligned}
 L(U', \delta) &= \|PX - U'S\|_F^2 + \sum_{m=1}^p \delta_m (\|u'_m\|_2^2 - 1) \\
 &= \|PX - U'S\|_F^2 + \text{tr}(U'^T U' \delta) - \text{tr}(\delta) \\
 &= \text{tr}(X^T P^T P X) - 2\text{tr}(U'^T P X S^T) + \text{tr}(S^T U'^T U' S) + \text{tr}(U'^T U' \delta) - \text{tr}(\delta)
 \end{aligned} \tag{3.12}$$

Algorithm 3 Feature-sign search algorithm for solving Eq. (3.7)

Require: Given a set of feature points $X = [x_1, \dots, x_n] \in R^{d \times n}$, a dictionary $U' = [u'_1, \dots, u'_p] \in R^{d' \times p}$, a projection matrix $P \in R^{d' \times d}$ and a predefined regularization parameter λ .

Ensure: A set of sparse code $S = [s_1, \dots, s_n] \in R^{p \times n}$ for features $X = [x_1, \dots, x_n]$

- 1: Initialization: For $x_i \forall i$, zero coefficients of s_i and an empty active set A
- 2: Select $m = \arg \max_m |\frac{\partial f(s_i)}{\partial s_i^{(m)}}|$ from zero coefficients of s_i

$$\text{if } |\frac{\partial f(s_i)}{\partial s_i^{(m)}}| > \lambda, \text{ then set } \vartheta_m = -1, A = \{m\} \bigcup A \quad (3.8)$$

$$\text{if } |\frac{\partial f(s_i)}{\partial s_i^{(m)}}| < -\lambda, \text{ then set } \vartheta_m = 1, A = \{m\} \bigcup A \quad (3.9)$$

- 3: Feature-sign step

Let $\hat{U}'$ be a sub-matrix of U' containing the corresponding columns in A.

Let $\hat{s}$ and $\hat{\vartheta}$ be sub-vectors of s and ϑ , respectively, corresponding to A.

Let $\hat{L}$ be sub-matrix of L corresponding to A, active set.

Compute the solution to $\min_{\hat{s}_i} \|Px_i - U'\hat{s}_i\|_2^2 + \lambda \hat{\vartheta}^T \hat{s}_i$

$$s_i^{\hat{new}} = 2(\hat{U}'^T \hat{U}' + \gamma L_{ii} I)^{-1} (\hat{U}'^T Px_i - \gamma \sum_{i \neq j} \hat{L}_{ij} \hat{s}_j - \frac{\lambda \hat{\vartheta}}{2}) \quad (3.10)$$

Perform a discrete line search from the closed line segment $\hat{s}_i$ to $s_i^{\hat{new}}$.

Check the objective value at $s_i^{\hat{new}}$ and all coefficients where any sign change.

Update $\hat{s}_i$ to the point with the lowest objective value.

Remove zero coefficients of $\hat{s}_i$ form A

Update $\vartheta = \text{sign}(s_i)$ where $\vartheta_m = \{-1, 0, 1\}$ denotes $\text{sign}(s_i^{(m)})$

- 4: Check the optimality condition

(a) For nonzero coefficients: $\frac{\partial f(s_i)}{\partial s_i^{(m)}} + \lambda \text{sign}(s_i^{(m)}) = 0, \forall s_i^{(m)} \neq 0$

Go to Step 3 if condition (a) is not satisfied; else check condition (b).

(b) For zero coefficients: $|\frac{\partial f(s_i)}{\partial s_i^{(m)}}| \leq \lambda, \forall s_i^{(m)} = 0$

Go to Step 2 if condition (b) is not satisfied; otherwise return s_i as the solution.

Algorithm 4 Graph-based Sparse Coding and Embedding in the training phase

Require:
Require: Given a set of feature samples $X = [x_1, \dots, x_n] \in R^{d \times n}$, a similarity matrix $W \in R^{n \times n}$, and regularization parameters λ , β and γ .

Ensure: A projection matrix $P \in R^{d' \times d}$, a dictionary $U' \in R^{d' \times p}$ and sparse codes $S = [s_1, \dots, s_n] \in R^{p \times n}$

 1: **repeat**

 2: select a d' -dimensional subspace $P \in R^{d' \times d}$ using Eq. (3.16)

$$\begin{aligned} & \min_P \|X - P^T X'\|_F^2 \\ & \text{subject to } PP^T = I \end{aligned}$$

 3: Initialization: For $x_i \forall i$, zero coefficients of s_i and an initial dictionary $U' \in R^{d' \times p}$.

 4: Calculate similarity matrix W using Eq. (3.3) where $W_{ij} = w(x_i, x_j)$

$$w(x_i, x_j) = \begin{cases} \exp(-\frac{\|x_i - x_j\|_2^2}{t}) & \text{if } x_i \in N_k(x_j) \text{ or } x_j \in N_k(x_i) \\ 0 & \text{otherwise} \end{cases}$$

 5: Compute $L = D - W$ where

$$D_{ii} = \sum_{j=1}^n W_{ij}$$

 6: **repeat**

7: Obtain sparse codes using Eq. (3.4)

$$\min_S \|PX - U'S\|_F^2 + \lambda \sum_{i=1}^n \|s_i\|_1 + \frac{\gamma}{2} \sum_{i=1}^n \sum_{j=1}^n \|s_i - s_j\|_2^2 w(x_i, x_j)$$

8: Update the dictionary using Eq. (3.11)

$$\begin{aligned} & \min_{U'} \|PX - U'S\|_F^2 \\ & \text{subject to } \|u'_m\|_2^2 \leq 1, \forall m = 1, \dots, p \end{aligned}$$

 9: **until** terminal condition is reached

 10: **until** terminal condition is reached

Finally, we can have the optimal solution U'^* using conjugate gradient as follows:

$$U'^* = PX S^T (S S^T + \delta)^{-1} \quad (3.13)$$

For the high dimensional feature data, we aim to seek a compact low-dimensional feature subspace and project these original data into the subspace. In the training phase, the mathematical expression can be formulated as follows:

$$\begin{aligned} \min_P & \|PX - U'V\|_F^2 + \beta \|X - P^T PX\|_F^2 \\ \text{subject to} & \quad PP^T = I \end{aligned} \quad (3.14)$$

Since U' and S are learned from PX in low-dimensional subspace which aim to minimize the quantization error between PX and the linear combination of $U'S$, we can rewrite the equation based on the new input variable X' where $X' = PX$:

$$\begin{aligned} \min_P & \|X' - U'S\|_F^2 + \beta \|X - P^T X'\|_F^2 \\ \text{subject to} & \quad PP^T = I \end{aligned} \quad (3.15)$$

Therefore, the formulation of seeking a projection matrix P can be rewritten as below:

$$\begin{aligned} \min_P & \|X - P^T X'\|_F^2 \\ \text{subject to} & \quad PP^T = I \end{aligned} \quad (3.16)$$

The reconstruction problem in Eq. (3.16) aims to minimize the reconstruction error between signals in the original feature space and those in the low-dimensional feature space. We show in the following the reconstruction error caused by one principal

component which denotes by $p \in R^{1 \times d}$:

$$\begin{aligned}
 R &= \|X - p^T p X\|_F^2 & (3.17) \\
 &= \text{tr}[(X - p^T p X)^T (X - p^T p X)] \\
 &= \text{tr}[(X^T - X^T p^T p)(X - p^T p X)] \\
 &= \text{tr}[X^T X - X^T p^T p X - X^T p^T p X + X^T p^T p p^T p X] \\
 &= \text{tr}[X^T X - X^T p^T p X] \\
 &= \text{tr}(X^T X) - \|p X\|_2^2 \\
 &= \|X^T X\|_F^2 - \|p X\|_2^2
 \end{aligned}$$

In Eq. (3.17) we show that minimizing the reconstruction error is equivalence to maximizing the variance of projected data:

$$\begin{aligned}
 &\max_p \|p X\|_2^2 & (3.18) \\
 &\text{subject to } \|p\| = 1
 \end{aligned}$$

Let $J(p)$ be a Lagrange form of Eq. (3.18) with the Lagrange multiplier δ as follows:

$$\begin{aligned}
 J(p) &= \|p X\|_2^2 - \delta(pp^T - 1) & (3.19) \\
 &= p X X^T p^T - \delta(pp^T - 1)
 \end{aligned}$$

Since Eq. (3.19) is a convex optimization problem, we can calculate the partial derivative of $J(\cdot)$ respect to p and set it to zero:

$$\frac{\partial J(p)}{\partial p} = 2X X^T p^T - 2\delta p^T = 0 \quad (3.20)$$

Then, we can transform the original maximization problem to a eigen-decomposition problem, and can be solved using singular value decomposition (SVD).

$$X X^T p^T = \delta p^T \quad (3.21)$$

Based on Eq. (3.4), Eq. (3.11) and Eq. (3.16), we have a projection matrix P and a learned dictionary U in the low-dimensional subspace. The learning details in the training phase are listed in Algorithm 4. Once we obtain U and P , for a feature x_i^{tr} in the training set, we can fix U and P of Eq. (3.7) and computer the similarity between pairs $w(x_i, x_j)$ where j is the k nearest neighbor of x_i so that we can obtain the corresponding sparse code s_i^{tr} of the feature x_i . For a new coming feature x_i^{ts} , we find its nearest neighbor in the training set and assign the sparse code s_i^{tr} to s_i^{ts} as listed in Algorithm 5.

Algorithm 5 Graph-based Sparse Coding and Embedding for each feature

Require: Given the dictionary $U' \in R^{d' \times p}$, the projection matrix $P \in R^{d' \times d}$ and the similarity matrix W .

Ensure: Obtain sparse codes for all features.

(a) For those features in the training set, obtain the corresponding sparse code s_i^{tr} of the feature x_i using Eq. (3.4)

$$\min_S \|PX - U'S\|_F^2 + \lambda \sum_{i=1}^n \|s_i\|_1 + \frac{\gamma}{2} \sum_{i=1}^n \sum_{j=1}^n \|s_i - s_j\|_2^2 w(x_i, x_j)$$

(b) For a new coming feature x_i^{ts} , find its nearest neighbor in the training set x_i^{tr} and assign the sparse code s_i^{tr} to s_i^{ts}

3.3 Activity-based Human Identification

The basic idea for activity-based identification is illustrated in Figure 3.1. For each video sequence, we first extract the binary human body masks by using background subtraction, thresholding, cropping, alignment and resizing. In the training phase, we learn a projection matrix and a low-dimensional codebook to quantize these binary masks into a set of histogram features by using graph-based sparse coding

Figure 3.1: Flowchart of the proposed activity-based human identification method.

and embedding. Subsequence, a maximum pooling function is adopted to represent each video clip, and a nearest neighbor classifier is used for classification.

3.3.1 Feature Extraction

Each video clip contains one person performing an activity. To obtain human body shape of movement as a set of image frames, we first extract coarse silhouettes using background subtraction, followed by a thresholding technique to get binary body masks. To mitigate the distance effect between camera and the subject and obtain the region of interest (ROI), we align these binary masks and resize them into the size of 64 x 48 which is similar to that in [151]. The sample image of each step is shown in Figure 3.2. By doing so, we successfully obtain a series of binary movement shapes. Some are the same person performing different activities, and some are the same activity performed by different individuals. Figure 3.3 shows two series of shape movement binary masks performed drinking behavior by two different persons. We aim to learn from these binary masks a set of basic patterns and use these basic patterns to represent each mask so that it can have much discriminative power for recognition. The GSCE method is proposed for this purpose and the following will introduce the method in detail.

Figure 3.2: The steps of feature extraction. From left to right, given a sample image, background subtraction and thresholding is used to obtain a binary mask. Then, we align the binary mask to reduce the effect caused by the distance between camera and the subject. By doing so, we can obtain the region of interest (ROI). Finally, we resize the ROI into 64x48 to mitigate the computational complexity.

Figure 3.3: Two series of sample ROI frames from two video sequences performed drinking behavior by two different persons. From left to right are the binary ROI masks of movement performed by the same person. The rows show different persons performing the same activity.

3.3.2 Feature Coding

We encode each binary mask x_i into a vector of sparse coefficients s_i using the proposed graph-based sparse and embedding algorithm discussed in Section 3.2.

3.3.3 Video Representation

For each video clip containing one person performing an activity, we extract the binary body masks of movement as a set of image frames, and, for each image frame, we concatenate pixels together to form a feature vector. For semantic image representation, we quantize these feature vectors into a small set of visual words using the GSCE algorithm listed in Algorithm 4 and Algorithm 5. After obtaining the sparse code of each ROI feature, we represent each video clip using the a maximum pooling function. There are three reasons of using the maximum pooling scheme. First, the coefficients of a sparse code represent clustering weights of visual words in the dictionary. Therefore, those sparse codes characterize the dynamic information of a video clip, and the largest weighting value in terms of absolute value means this visual word become a major cluster in certain image frame. Second, noise usually gains a small weighting response. Third, many algorithms have successfully applied the maximum pooling function into image representation [89, 91, 151, 175]. Let $S_i = [s_1, s_2, \dots, s_{n_i}] \in R^{p \times n_i}$ be a set of sparse codes with n_i image frames from video clip i . The video representation of the clip i denoted by $Q_i \in R^p$ is formulated as follows:

$$Q_i(j) = \max(|s_1(j)|, |s_2(j)|, \dots, |s_{n_i}(j)|)$$

where $j = 1, \dots, p$, and p is the number of visual words in the dictionary. For each visual word, the maximum pooling function characterizes the largest weighting parameter in terms of absolute value among those sparse codes in the image set of a video clip to describe the dynamic information of activity.

Figure 3.4: Sample frames from 9 different activity video clips performed by the same person in the Weizmann database. From left to right and top to bottom are the activities of jumping jack, jump, jump in place, run, gallop-sideways, skip, walk, one-hand wave, and two-hands wave, respectively.

3.3.4 Human Identification

For each test video clip t , we recognize the identity of the person from his/her activity. We first extract binary body masks of movement from the test video, and then quantize these body masks into a set of visual words with corresponding weightings of sparse codes by finding the nearest neighbors in the training set. With maximum pooling, we can obtain the feature vector Q_t for video representation. Subsequently, the identify can be recognized by using a nearest neighbor classifier with Euclidean distance between the test sample Q_t and each training sample Q_i formulated as follows:

$$\begin{aligned} k &= \arg \min_i d(Q_t, Q_i) \\ &= \arg \min_i \|Q_t - Q_i\|_2^2 \end{aligned} \quad (3.22)$$

where $d(\cdot)$ is a distance matrix, and the Euclidean distance is adopted here. Therefore, according to Eq. (3.22), Q_t and Q_k are expected to be the same person.

3.3.5 Data Sets

We perform experiments on two publicly available databases, Weizmann [176] and MOBISERV-AIIA [150]. In the Weizmann database, there are 9 persons performing 10 different actions including walk, run, jump, gallop-sideways, skip, bend, jump in place, jumping jack, one-hand wave, and two-hands wave as shown in Figure 3.4. In our experiments, we segment the videos into 215 clips so that each clip contains one person with at least one action cycle. Note that we remove the videos of the bend action due to the lack of enough training and test samples. Then, for each person, we randomly select one clip from each action for training, and the rest are as test samples.

The MOBISERV-AIIA database contains 12 persons performing eating and drinking activities in four different days. In each day, there are 2 clothing scenarios, one wearing short sleeves and one wearing long sleeves, respectively as shown in Figure 3.5. We adopt two activities, namely, eating with a fork and drinking with a cup so that we segment the videos into 776 clips containing one person with one action cycle of an activity. For each person, we randomly choose one-day clips for testing, and the rest clips for training.

3.3.6 Parameter Settings

We compare our method with existing state-of-the-art activity-based human identification methods proposed in [149, 151] and other sparse coding techniques. We implement the fuzzy c-means method proposed in [149] with a fuzzy parameter $m = 1.1$. The number of dynemes are empirically set as 60, 200, 450, for the Weizmann and MOBISERV-AIIA, respectively. For the sparse coding methods, the

Figure 3.5: Sample frames from two activity video sequences in the MOBISERV-AIIA database. We use two activities, drinking with a cup and eating with a fork, and two clothing scenarios, one wearing long sleeves and one wearing short sleeves, respectively. From left to right are the person drinking with a cup with long sleeves, the person drinking with a cup with short sleeves, the person eating with a fork with long sleeves and the person eating with a fork with short sleeves, respectively.

parameters are set as 200 bases, $\lambda = 0.3$ and $\beta = 0.3$ for the Weizmann databases, and $\lambda = 0.1$ and $\beta = 1$ for MOBISERV-AIIA database. The regularization parameters λ , γ and β we used are determined via cross validation. Experiments are repeated ten times with different randomly selected training and testing samples, and the final results are shown as the mean of the correct recognition rate (CRR).

3.3.7 Comparisons with Existing Activity-based Human Identification Methods

We first compare the proposed GSCE method with other existing approaches for activity-based human identification on the two databases. Table 3.1 shows the CRR of different methods on each database, respectively. As shown, the proposed GSCE method outperforms the methods in [149, 151] with gains in correct recognition rate of 13.43% and 2.24% on the Weizmann database, and 24.35% and 2.07% on the MOBISERV-AIIA database, respectively. This is because the method in [149] adopts the fuzzy c-mean as the quantization method which may lost some discriminative information. Instead, [151] uses the sparse coding technique for the quantization step which has been shown to have a better performance than the fuzzy c-means.

However, local structure between features are not considered. In our work, we further improve the performance by integrating the conventional sparse coding method with projection matrix and the consideration of local structure.

Table 3.1: CRR comparison (%) with the existing activity-based human identification methods.

Method	Weizmann	MOBISERV-AIIA
FCM [149]	78.36	52.85
SC [151]	89.55	75.13
GSCE	91.79	77.20

3.3.8 Comparisons with Existing Sparse Coding Methods

To further investigate the effectiveness of the proposed GSCE algorithm, we compare our method with existing sparse coding methods as shown in Table 3.2 in terms of the correct recognition rate. Here, the conventional sparse coding method is denoted by SC, the Laplacian sparse coding method proposed in [91] is denoted by LSC, and the graph-based sparse coding method proposed in [59] is denoted by GraphSC. As shown, the proposed GSCE method outperforms SC, LSC, GraphSC with gains in correct recognition rate of 2.24%, 1.49%, and 1.49% on the Weizmann database, and 2.07%, 4.66%, and 4.66% on the MOBISERV-AIIA database, respectively.

3.3.9 Parameter Analysis

We first evaluate the effect of the feature dimension of our GSCE on the recognition performance. Figure 3.6 shows the correct recognition rate of our GSCE versus different feature dimensions from 50-dim to 3000-dim on the Weizmann dataset. We

Table 3.2: CRR comparison (%) with the existing sparse representation methods.

Method	Weizmann	MOBISERV-AIIA
SC	89.55	75.13
LSC [91]	90.30	72.54
GraphSC [59]	90.30	72.54
GSCE	91.79	77.20

Figure 3.6: Average recognition rate (%) of our GSCE versus different feature dimension on the Weizmann dataset

can see that our proposed GSCE can achieve stable performance when the feature dimension reaches 50 and onwards. We show in Figure 3.7 the effect of the feature dimension of our GSCE in terms of computational cost (seconds). We can see that although we can obtain stable performance from 50-dim to 3000-dim, computational cost increases when the feature dimension increases.

Figure 3.7: Average computational cost (second) of our GSCE versus different feature dimension on the Weizmann dataset

3.3.10 Confusion Matrix

Table 3.3 and Table 3.4 show the confusion matrices of the recognition results on the Weizmann and MOBISERV-AIHA datasets, accordingly. In Table 3.3, we can find that two persons are well-identified, Den and Iyo, and the person Mos has the lowest correct recognition rate which has 14% misidentified to Sha and 7% to Ido. In Table 3.4, we can find that three persons are well-identified, P6, P7 and P12, and several persons are seriously mis-identified and obtain under 50% correct recognition rate such as P3, P5 and P8. The person P3 has lowest correct recognition rate 25% which has 50% misidentified to P8 and 25% to P1. Figure 3.8 and Figure 3.9 show the sample visual words (codebooks).

Figure 3.8: Sample codebook pictures of the Weizmann dataset.

Figure 3.9: Sample codebook pictures of the MOBISERV-AIHA dataset.

Table 3.3: Confusion matrix of our GSCE method on the Weizmann dataset.

	Dar	Den	Eli	Ido	Ira	Len	Iyo	Mos	Sha
Dar	92	0	8	0	0	0	0	0	0
Den	0	100	0	0	0	0	0	0	0
Eli	0	0	94	0	6	0	0	0	0
Ido	0	0	0	89	11	0	0	0	0
Ira	0	0	0	0	94	0	0	0	6
Len	0	0	0	0	0	85	0	0	15
Iyo	0	0	0	0	0	0	100	0	0
Mos	0	0	0	7	0	0	0	79	14
Sha	0	0	0	0	0	0	0	8	92

Table 3.4: Confusion matrix of our GSCE method on the MOBISERV-AIIA dataset.

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12
P1	88	0	0	0	0	0	0	0	0	0	13	0
P2	0	81	0	0	0	0	0	0	0	0	19	0
P3	25	0	25	0	0	0	0	50	0	0	0	0
P4	0	0	0	81	0	19	0	0	0	0	0	0
P5	6	0	0	0	50	38	0	6	0	0	0	0
P6	0	0	0	0	0	100	0	0	0	0	0	0
P7	0	0	0	0	0	0	100	0	0	0	0	0
P8	25	0	0	0	31	0	0	44	0	0	0	0
P9	0	0	19	0	0	0	0	0	81	0	0	0
P10	0	0	0	0	0	19	0	0	0	81	0	0
P11	0	6	0	0	0	0	0	0	0	0	94	0
P12	0	0	0	0	0	0	0	0	0	0	0	100

3.3.11 Discussion

According to the above results, we have made the following observations:

- (1) The proposed graph-based sparse coding and embedding method for feature quantization can achieve higher recognition accuracy than the other state-of-the-art methods using Fuzzy c-means [149] or general sparse coding [151]. The improvement in recognition is because the proposed graph-based sparse coding and embedding approach finds a compact and low-dimensional feature subspace and considers the geometrical structure between features in such a new space, while Fuzzy c-means is a region-based method which may result in a large quantization error and sparse coding does not utilize feature relationship which may cause similar features with totally different sparse codes.
- (2) The proposed graph-based sparse coding and embedding method for feature quantization can achieve better recognition performance than the other sparse coding related techniques such as SC, LSC and GraphSC. One of the possible reason is that they ignore the fact that original feature data may lie in the high-dimensional feature space which may give rise to the difficulty of learning a large over-complete dictionary.
- (3) The proposed graph-based sparse coding and embedding method for feature quantization can achieve similar correct recognition rates under different feature dimension levels but reduce the computational cost when mapping features into a smaller feature subspace. This is because we simultaneously seek a compact low-dimensional subspace, minimize reduction cost and maintain geometrical structure in each space.

3.4 Summary

We have proposed in this chapter a graph-based sparse coding and embedding (GSCE) method for activity-based human identification which aims to minimize the low-dimensional reconstruction error, as well as the loss caused by dimension transformation and the penalty of local sparse reconstructed coefficients. We first extract binary human body masks from video clips by background subtraction, thresholding, alignment and resizing. Then, a projection matrix is learned to map these binary body masks into a low-dimensional feature space, and a codebook is learned with consideration of local structure to quantize the masks into a set of histogram features. Videos are represented by a maximum pooling function and a NN classifier is used for classification. Our experimental results have clearly demonstrated the efficacy of our proposed method in recognizing human at a distance. Further, it is worth mentioning that the proposed method appears to be the first algorithm that formally addresses the compact and low-dimensional geometrical structure into the sparse coding framework.

Chapter 4

Sparse Manifold Coding

4.1 Introduction

The study on activity-based human identification is relatively new, and there are very limited efforts to tackle this problem in the literature. Gkalelis *et al.* [149] and Iosifidis *et al.* [150] are arguably the pioneers working along this direction. In their work, features are encoded using fuzzy c-means (FCM) vector quantization which is a region-based soft assignment. Each video clip containing an activity performed by a person was represented using a mean pooling method followed by a linear discriminant analysis (LDA) based algorithm. They adopted two-step classification. The first classifier is an action classifier for movement recognition and then a movement specific person classifier was used for human identification. While some encouraging results have been obtained, their method requires correctly recognizing the activity type of the person and may fail to work well when the activity recognition is incorrect. In another attempt, Lu *et al.* [151] quantized features using sparse coding followed by a discriminative metric learning method. While they show that using sparse coding for feature quantization can obtain a smaller quantization error than using FCM, their method ignores the geometrical

structure between features in the data space which may cause that similar features are quantized into different visual words. Thus, our previous work [177] proposed a feature coding method using the graph-based sparse coding and embedding (GSCE) approach which considers the local structure in the low-dimensional subspace so that similar features may obtain similar visual words. While some encouraging results have been obtained, there are still some limitations which can be further improved for feature coding, sparsity estimation and dictionary learning for activity-based human identification:

- (1) While a number of existing methods have been proposed for feature quantization, they usually assume that there only exists a global geometrical structure in the feature set where the similarity between features is hold. The assumption, however, may not be always sufficient because the relationship of movement patterns performed by a person which usually lies in a movement manifold is ignored.
- (2) While a number of sparsity estimation methods have been proposed in recent years, most of them estimate sparse coefficients using the lasso solutions, which may cause a large computational cost, especially when curring high feature dimensionality or large amount of visual words in the dictionary. Therefore, it is necessary to have an efficient sparsity solution.

Motivated by the above observations, we propose a novel sparse manifold coding (SMC) method for activity-based human identification, integrating multi-manifold structures and graph-based sparse coding into the general sparse coding framework for feature quantization, and further providing an efficient solution using marginal regression coefficients for sparsity estimation.

Our contributions in this chapter includes the following:

- (1) We utilize the temporal information to form a video-based manifold. By doing so, an unsupervised model can become a weakly supervised model, and each frame is encoded with the consideration of its local and global neighborhoods, accordingly so that similar features can be obtained similar coefficients especially for those coming from the same manifolds. Besides, we provide a supervised model by building hierarchical video-based and person-based manifolds where features hold their similarity relationship in the corresponding manifolds, accordingly.
- (2) We provide sparsity estimation using marginal regression and dictionary learning algorithm to solve the proposed sparse manifold coding problem which can achieve more than ten times faster than the conventional feature-sign solution.

The rest of the chapter is organized as follows: Section 4.2 details the proposed method in the unsupervised manner for activity-based human identification. Section 4.3 describes the proposed supervised sparse manifold coding model. Section 4.4 presents the experimental results. Finally, we conclude the chapter in Section 4.5.

4.2 Unsupervised Sparse Manifold Coding

We proposed a sparse manifold coding (SMC) method for activity-based human identification, aiming to recognize the identity of a person from his/her activities such as walking, running, eating and drinking. Figure 4.2 illustrates main idea of the

Figure 4.1: Flowchart of the proposed sparse manifold coding framework for activity-based human identification.

proposed sparse manifold coding approach. We encode an input by simultaneously considering its neighborhood in the same manifold and its neighborhood in the global data space so that similar features, especially for those extracted from the same movement sequences, may have higher probability of obtaining similar sparse codes. To address this idea, we propose in this chapter a discriminative feature coding approach to encode dynamic movement patterns by capturing their corresponding dynamic characteristics within his/her movement manifold and simultaneously maintaining the global geometric structure between manifolds. We estimate sparsity using marginal regression coefficients, which significantly improves the coding speed.

Figure 4.1 shows the flowchart of the proposed activity-based human identification framework. We first extract binary body masks from each activity clip to construct a feature basis set. In the training phase, we learn a dictionary and estimate sparsity for feature quantization via the proposed unsupervised sparse manifold coding (USMC) model which explores geometrical structure within and between manifolds. In the test phase, each new binary body mask can be quantized into visual words via the proposed NN-based sparsity approximation. Finally, we

Figure 4.2: The basic idea of our sparse manifold coding approach, which emphasizes the dependency of features within each movement manifold and simultaneously maintains the global structure between features in the data space, so that similar features, especially for those extracted from the same movement sequences may obtain similar sparse codes.

represent each video clip using a maximum pooling function and adopt a nearest-neighbor classifier for recognition.

4.2.1 Motivation

This chapter presents a new feature quantization approach for activity-based human identification. The activity-based human identification task aims to characterize dynamic movement patterns performed by a person so that his/her identity can be recognized via his/her behavior. Conventional feature quantization methods for activity-based human identification usually assume that there only exists a global

Figure 4.3: Unsupervised sparse manifold coding (USMC) model. For this example, each video form a video-based manifold denoted by $[\mathcal{V}_1, \dots, \mathcal{V}_8]$ with its local features $[x_1, \dots, x_n]$ denoted by circles. Note that n is number of features which varies according to each clip. We aim to minimize the quantization error and simultaneously maintain the geometrical structure within and between manifolds. To achieve this goal, we construct two similarity matrices denoted by S^v and S^u with different thresholding parameters w^v and w^u , respectively. Therefore, the red dot feature is quantized respect to its selected neighborhood within and between manifolds.

manifold structure available in the feature set. This assumption, however, may not hold because it ignores the intrinsic structure of a video, where successive movement frames performed by a person form a video manifold. While a number of existing methods have been proposed for feature quantization, there is little work on considering the relationship of feature dependency in/between manifolds. To address this, we propose in this chapter a novel sparse manifold coding (SMC) approach for feature quantization which explicitly utilizes the dynamic information of a person in the video-specific manifold and simultaneously considers the global geometrical structure of the data space between multiple manifolds.

4.2.2 Unsupervised Sparse Manifold Coding

To encode dynamic patterns of human activities performed by different persons, Gkalelis *et al.* [149] and Iosifidis [150] *et al.* used FCM to cluster human body masks into a number of clusters. Since some discriminative information may

be lost due to the quantization error, Lu *et al.* [151] used sparse coding to quantize these masks into a set of visual words. Since their method ignores the geometrical structure between features in the data space which may cause that similar features are quantized into different visual words, our previous work [177] proposed a GSCE method to quantize them by exploiting the global geometrical structure. As mentioned before, however, they ignore the intrinsic structure of a video. Inspired by the fact that features extracted from successive movement frames share the same person information, we propose a sparse manifold coding framework to utilize the temporal information from each video clip and simultaneously exploit the geometrical structure in the data space. The main idea is illustrated in Figure 4.3. We assume that each video contains a person performing an activity, and based on this assumption, a series of movement features can form a video manifold denoted by $\{\mathcal{V}_i\}_{i=1}^N$ where N is the number of videos in the training set. Let $X = [x_1, \dots, x_n] \in \mathbb{R}^{d \times n}$ be a set of normalized features in a d -dimensional feature space of $\{\mathcal{V}_i\}$. We aim to minimize the quantization error and simultaneously maintain the geometrical structure within and between manifolds. To achieve this goal, we construct two similarity matrices, W^u and W^v with different thresholding parameters w^u and w^v , respectively. Then a feature x_i is quantized respect to its selected neighborhood within and between manifolds. The mathematical expression of USMC can be written as follows:

$$\min_{U, S, \forall i} \sum_i \left(\sum_j W^u(x_i, x_j) \|x_j - U s_i\|_2^2 + \sum_k W^v(x_i, x_k) \|x_k - U s_i\|_2^2 \right) \quad (4.1)$$

$$\text{subject to } \|s_i\|_1 \leq E$$

$$\|u_m\|_1 \leq 1 \quad m = 1, \dots, p$$

where $U = [u_1, \dots, u_p] \in \mathbb{R}^{d \times p}$ is a dictionary with p normalized bases, and s_i is the sparse coefficients of x_i . The first term in Eq. (4.1) is the inter-manifold

quantization error, and the second term is the intra-manifold quantization error with an l_1 -norm penalty constraint E where l_1 -norm refers to the sum of absolute value of each element $\|s_i\|_1 = \sum_{m=1}^p |s_i^{(m)}|$. There are many ways to determine the similarity function, and we define the intra-manifold similarity between two features with their correlations as follows:

$$W^v(x_i, x_k) = \begin{cases} x_k^T x_i & x_k^T x_i \geq w^v \text{ and } x_i, x_k \in \mathcal{V}_\alpha \\ 0 & \text{otherwise} \end{cases} \quad (4.2)$$

$W^v(x_i, x_k)$ contains value when x_i and x_k reside in the same manifold and their correlation is above the threshold w^v . On the other hand, the inter-manifold similarity between two features can be defined as follows:

$$W^u(x_i, x_j) = \begin{cases} x_j^T x_i & x_j^T x_i \geq w^u \text{ and } x_i \in \mathcal{V}_\alpha, x_j \notin \mathcal{V}_\alpha \\ 0 & \text{otherwise} \end{cases} \quad (4.3)$$

$W^u(x_i, x_j)$ contains value when x_i and x_j reside in different manifolds and their correlation is above the threshold w^u . The objective function in Eq. (4.1) is not convex on U and s_i simultaneously, but it is convex when one of them is fixed. Therefore, in the training phase, we can iteratively update the dictionary and the sparse codes. The process of sparsity estimation and dictionary learning is listed in Algorithm 6.

4.2.3 Sparsity Estimation

When U is fixed, we can estimate each s_i by optimizing the following formulation:

$$\begin{aligned} \min_{s_i} \quad & \sum_j W^u(x_i, x_j) \|x_j - U s_i\|_2^2 + \sum_k W^v(x_i, x_k) \|x_k - U s_i\|_2^2 \\ \text{subject to} \quad & \|s_i\|_1 \leq E \end{aligned} \quad (4.5)$$

Algorithm 6 Sparsity Estimation and Dictionary Learning

Require: Data $X = [x_1, \dots, x_n] \in \mathbb{R}^{n \times n}$, a random normalized dictionary $U = [u_1, \dots, u_p] \in \mathbb{R}^{d \times p}$, an energy constraint E and similarity matrices W^u and W^v with corresponding parameters w^u and w^v , respectively.

Ensure: Learned dictionary $U \in \mathbb{R}^{d \times p}$, sparse coefficients $S = [s_1, \dots, s_n] \in \mathbb{R}^{p \times n}$

1: **repeat**

2: Obtain sparse codes using the latest dictionary:

$$\min_{s_i} \sum_j W^u(x_i, x_j) \|x_j - U s_i\|_2^2 + \sum_k W^v(x_i, x_k) \|x_k - U s_i\|_2^2 \quad (4.4)$$

subject to $\|s_i\|_1 \leq E$

3: Update the dictionary based on the sparse codes obtained from the previous step.

$$\min_U \sum_i \|x_i - U s_i\|_2^2 + \gamma \|U^T U - I\|_F^2$$

subject to $\|u_m\|_1 \leq 1, \forall m$

4: **until** convergence

Eq. (4.5) can be solved via marginal regression which is revisited in recent literature [87, 88, 145] showing that the computational complexity of marginal regression based methods are two orders faster than that of the lasso-based methods [79, 83], even though there are some efficient algorithms using gradient descent [81, 82] for lasso [77, 78]. Therefore, we propose a marginal regression solution for sparsity estimation of USMC due to its efficiency. Specifically, we consider the following optimization problem:

$$\min_{s_i} \|x_i - U s_i\|_2^2 \quad (4.6)$$

The component-wise marginal regression coefficients can be written as follows:

$$\hat{a}_i \equiv U^T x_i \quad (4.7)$$

Therefore, based on the objective function of Eq. (4.5), the regression coefficients can be defined as follows:

$$\hat{a}_i^{(m)} = \begin{cases} \frac{\sum_j W^u(x_i, x_j) u_m^T x_j + \sum_k W^v(x_i, x_k) u_m^T x_k}{\|W_i^u + W_i^v\|_2^2} & , \|W_i^u + W_i^v\|_2^2 > 0 \\ 0 & , \text{otherwise} \end{cases} \quad (4.8)$$

where $\hat{a}_i^{(m)}$ refers to the m th element of $\hat{a}_i$, W_i^u and W_i^v are similarity vectors of W^u and W^v subject to x_i , respectively. Then we select the top k large coefficients in terms of their absolute values whose l_1 norm is bounded by a constraint E [87, 145] so that each sparse code can be represented under a similar sparsity level. The equation can be reformulated as follows:

$$s_i^{*(m)} = \begin{cases} \hat{a}_i^{(m)} & , m \in Q \\ 0 & , \text{otherwise} \end{cases} \quad (4.9)$$

where $Q = \{J_1, \dots, J_r : r \leq p : \sum_{z=1}^r |\hat{a}_i^{(J_z)}| \leq E\}$. We sort the coefficients in terms of their absolute values in descending order which denotes by indexes $J_1, \dots, J_p$ where $|\hat{a}^{(J_1)}| > |\hat{a}^{(J_2)}| > \dots > |\hat{a}^{(J_p)}|$. Finally, we can encode a set of features $X = [x_1, \dots, x_n] \in \mathbb{R}^{d \times n}$ into a set of sparse coefficients $S = [s_1^*, \dots, s_n^*] \in \mathbb{R}^{p \times n}$ via Eq. (4.8) and Eq. (4.9). Notice that s_i^* may be an zero-vector when feature x_i is an isolated point so that it is no highly correlated neighbors within and between manifolds with $\|W_i^u + W_i^v\|_2^2 = 0$. From the recognition viewpoint, this kind of features are outliers in the dataset.

4.2.3.1 NN-based Sparse Code Approximation

In order to speed up the feature quantization process, we propose a NN-based sparse code approximation method to encode new data which can easily trace the geometrical structure in the training space. As shown in Figure 4.4, when a new test

Figure 4.4: NN-based sparsity approximation. In this example, the new feature denotes by the black circle x_t . $W^t(x_t, x_i) = x_i^T x_t$ calculates the correlation between new sample and every sample x_i in the training dataset. By finding the most correlated sample x_o which denotes by the red circle, we assign sparse codes b_o^* to x_t to maintain the geometrical structure.

sample x_t comes, the new feature coefficients s_t can be obtained via Eq. (4.8) and Eq. (4.9) based on the SMC model in Eq. (4.1). To take the advantage from the original geometrical structure in the training set and simultaneously speed up the feature quantization procedure, we approximate the new sparse code s_t by finding in the training dataset the most correlated feature x_o as follows:

$$o = \arg \max_i x_i^T x_t \quad (4.10)$$

and then we can obtain the new sparse code in the following way:

$$s_t^* = s_o^* \quad (4.11)$$

4.2.4 Dictionary Learning

In this subsection, we introduce the dictionary learning method. When S is fixed, we can estimate the dictionary by optimizing the following formulation:

$$\min_U \sum_i \|x_i - U s_i\|_2^2 \quad (4.12)$$

$$\text{subject to } \|u_m\| \leq 1, \forall m \quad (4.13)$$

where the dictionary U is a set of normalized visual words denoting by each column u_m . Since marginal regression works well when the columns in the dictionary have low correlation [87, 88, 145, 146], we follow the previous work to formulate the optimization problem of the dictionary learning by adding a term $\|U^T U - I\|_F^2$ as follows:

$$\min_U \sum_i \|x_i - U s_i\|_2^2 + \gamma \|U^T U - I\|_F^2 \quad (4.14)$$

$$\text{subject to } \|u_m\| \leq 1, \forall m \quad (4.15)$$

where γ is a regularization parameter. This optimization problem can be solved via the first-order gradient descent method [145, 146]. It is worth mentioning that we initialize the dictionary as the results of online dictionary learning [146] to speed up the convergence. Let $F(U)$ be the objective function as follows:

$$F(U) = \sum_i \|x_i - U s_i\|_2^2 + \gamma \|U^T U - I\|_F^2 \quad (4.16)$$

Then, we compute its gradient respect to U at the q iteration which denotes by $\nabla F(U_q)$:

$$\nabla F(U_q) = 2(U_q S S^T - X S^T) + 4\gamma(U_q U_q^T U_q - U_q) \quad (4.17)$$

where $X = [x_1, \dots, x_n]$. The first-order gradient descent method can be given as follows:

$$U_{q+1} = \Pi_U \{U_q - \beta \nabla F(U_q)\} \quad (4.18)$$

Algorithm 7 Dictionary Learning via Sparse Manifold Coding

Require:
Require: Given a set of training samples $X = [x_1, \dots, x_n] \in \mathbb{R}^{d \times n}$, regularization parameters β and γ , and the sparse level constraint E or e .

Ensure:
Ensure: A dictionary $U \in \mathbb{R}^{d \times p}$, sparse coefficients $S = [s_1, \dots, s_n] \in \mathbb{R}^{p \times n}$

- 1: Initialization: For $x_i \forall i$, zero coefficients of s_i and an initial dictionary $U \in \mathbb{R}^{d \times p}$.
- 2: **repeat**
- 3: Obtain sparse codes S via Eq. (4.5) with the latest dictionary U :

$$\hat{a}_i^{(m)} = \begin{cases} \frac{\sum_j W^u(x_i, x_j) u_m^T x_j + \sum_k W^v(x_i, x_k) u_m^T x_k}{\|W_i^u + W_i^v\|_2^2} & , \|W_i^u + W_i^v\|_2^2 > 0 \\ 0 & , \text{otherwise} \end{cases}$$

$$s_i^{*(m)} = \begin{cases} \hat{a}_i^{(m)} & , m \in Q \\ 0 & , \text{otherwise} \end{cases}$$

 where $Q = \{J_1, \dots, J_r : r \leq p : \sum_{z=1}^r |\hat{a}_i^{(J_z)}| \leq E\}$.

- 4: **for** $q = 1 : Q$ **do**
- 5: Compute U_q based on the sparse codes obtained from the Step 3 so that

$$U_{q+1} = \Pi_U\{U_q - \beta \nabla F(U_q)\}$$

- 6: **end for**
 - 7: $U = U_{Q+1}$
 - 8: **until** convergence
-

where parameter β is the step size, and Π_U is the projection function which maps each column u_m to the l_2 -norm unit ball. More specifically, the dictionary is learned iteratively by finding the local minimum along the gradient direction based on the small step size and normalizing each column vector with length smaller or equal to one until convergence. The sparsity estimation and the dictionary learning process is shown in Algorithm 7.

Figure 4.5: Supervised sparse manifold coding (SSMC) with video-based and person-based manifolds structure. For this example, we form two layers of manifold. Each video form a first layer video-based manifold denoted by $[\mathcal{V}_1, \dots, \mathcal{V}_8]$ with its local features $[x_1, \dots, x_n]$ denoted by circles. Note that n is number of features which varies according to each video clip. Further, we construct a person-based manifold for each class. As shown, $\mathcal{V}_1, \mathcal{V}_2, \mathcal{V}_5, \mathcal{V}_6$ build a person-based manifold and $\mathcal{V}_3, \mathcal{V}_4, \mathcal{V}_7, \mathcal{V}_8$ build another so that each feature with the corresponding person-based manifold shares the same person information.

4.3 Supervised Sparse Manifold Coding

For recognition tasks, minimizing feature quantization error for sparse representation plays an important role, but how to increase discriminative power between classes is also important. We provide in this section a supervised version of sparse manifold coding denoted by SSMC. SSMC constructs a two-layer manifold structure as shown in Figure 4.5. The first layer indicates person-specific manifolds referring to the person classes. Within each person manifold is the second layer manifold structure indicating activity clips performed by the person. By doing so, SSMC utilizes the temporal information of a video which shares the movements performed by a person and simultaneously exploits the extra information from other video manifolds which contain other activities performed by the same person. Let $\{\mathcal{C}_i\}_{i=1}^T$ form each person-based manifold where T is the number of classes. Then, we reformulate the objective

function of USMC in Eq. (4.1) as the following optimization problem:

$$\begin{aligned} \min_{U, S, \forall i} \quad & \sum_i \left(\sum_j W^s(x_i, x_j) \|x_j - U s_i\|_2^2 + \sum_k W^v(x_i, x_j) \|x_k - U s_i\|_2^2 \right) \quad (4.19) \\ \text{subject to} \quad & \|s_i\|_1 \leq E \\ & \|u_m\|_1 \leq 1 \quad m = 1, \dots, p \end{aligned}$$

where the supervised inter-manifold similarity between two features which denotes by W^s can be defined as follows:

$$W^s(x_i, x_j) = \begin{cases} x_j^T x_i & x_j^T x_i \geq w^s \text{ and} \\ & x_i \in \mathcal{V}_\alpha, x_j \notin \mathcal{V}_\alpha, x_i, x_j \in \mathcal{C}_\beta \\ 0 & \text{otherwise} \end{cases} \quad (4.20)$$

$W^s(x_i, x_j)$ contains value when x_i and x_j reside in different video-based manifolds but in the same person-based manifold with their correlation above the threshold w^s . Similar to the unsupervised model, the optimization of the sparsity estimation can be re-formulated as follows:

$$\begin{aligned} \min_{s_i} \quad & \sum_j W^s(x_i, x_j) \|x_j - U s_i\|_2^2 + \sum_k W^v(x_i, x_j) \|x_k - U s_i\|_2^2 \quad (4.21) \\ \text{subject to} \quad & \|s_i\|_1 \leq E \end{aligned}$$

Then, the marginal regression coefficients can be computed:

$$\hat{a}_i^{(m)} = \begin{cases} \frac{\sum_j W^s(x_i, x_j) u_m^T x_j + \sum_k W^v(x_i, x_k) u_m^T x_k}{\|W_i^s + W_i^v\|_2^2} & , \|W_i^s + W_i^v\|_2^2 > 0 \\ 0 & , \text{otherwise} \end{cases} \quad (4.22)$$

where W_i^s and W_i^v are similarity vectors of W^s and W^v subject to x_i , respectively. Based on the bound of the l_1 -norm penalty, E , the equation can be reformulated as follows:

$$s_i^{*(m)} = \begin{cases} \hat{a}_i^{(m)} & , m \in Q \\ 0 & , \text{otherwise} \end{cases} \quad (4.23)$$

where $Q = \{J_1, \dots, J_r : r \leq p : \sum_{z=1}^r |\hat{a}_i^{(J_z)}| \leq E\}$.

4.4 Activity-based Human Identification

We perform activity-based human identification experiments on two publicly available databases, Weizmann [176] and MOBISERV-AIIA [150]. In the dataset, each video clip contains one person performing one activity at least one cycle. We denote the proposed unsupervised sparse manifold coding by USMC and denote the supervised sparse manifold coding by SSMC. The regularization parameter γ and the penalty constraint E is chosen via cross-validation. In addition, the weighting parameters in our formulation, w^u and w^v of USMC, and w^s and w^v of SSMC, are set empirically and have more discussed in the following sub-section. To have a fair comparison, the number of clusters is set as 200. We compare our method with existing state-of-the-art activity-based human identification methods [149, 151, 177] and other sparse coding techniques such as sparse coding (SC), Laplacian Sparse Coding (LSC) [90], Graph-based Sparse Coding (GraphSC) [59] and our previous work Graph-based Sparse Coding and Embedding (GSCE) [177]. Note that LSC and GraphSC are two models with the same concept but different initialization. We evaluate the performance according to the correct recognition rate (CRR).

For feature extraction, we first extract binary body masks from each clip to construct a feature set. Each clip contains a person performing an activity and multi-types of activities are considered. To obtain human shape of movement, we extract silhouettes using background subtraction and then apply a thresholding technique to obtain binary body masks. To mitigate the distance effect between camera and the subject and obtain the region of interest (ROI), we align these binary masks and resize them into the size of 64 x 48 which is similar to the way in [151, 177]. We aim to learn from these binary masks to a set of basic patterns and use them to represent each mask so that it can have much discriminative power for

recognition. The SCM method is proposed for this purpose and the following will introduce the USCM method in detail.

Having obtained a set of sparse codes $S = [s_1^*, \dots, s_n^*] \in \mathbb{R}^{p \times n}$ from binary body masks of an activity clip $\mathcal{V}_i$, we represent each clip by a maximum pooling function [151, 175, 177] as follows:

$$h_i^{*(m)} = \max\{|s_1^{*(m)}|, |s_2^{*(m)}|, \dots, |s_n^{*(m)}|\} \quad (4.24)$$

Then, for a test video clip $\mathcal{V}_t$, we recognize the identity of a person using a simple nearest neighbor classifier based on the Euclidean distance between the test sample h_t and each training sample h_i as follows:

$$c = \arg \min_i d(h_t, h_i) \quad (4.25)$$

where $d(\cdot)$ is the Euclidean distance function. Therefore, the identity of h_c , is likely the identity of the test sample h_t .

4.4.1 Data Sets

The Weizmann dataset, as shown in Figure 4.6, contains 9 persons, and each person performed 10 different activities, including bend, run, skip, walk, gallop-sideways, jumping-jack, jumping-forward-on-two-legs, jump in place-on-two-legs, wave-one-hand, and wave-two-hands, respectively. Similar to [151, 177], we segment the videos into clips to generate an activity dataset so that each clip contains a single-period activity performed by a person. Note that the clips of bend activity are removed because of the lack of enough training/test samples for this activity. At last, we construct a dataset of 215 clips in total. We randomly select one clip per action

Figure 4.6: Three activity examples of five persons from the Weizmann dataset. Every two columns refer to successive frames of the jump, side, and skip activities from left to right, respectively. Each row represents a person performing three activities.

Figure 4.7: There are sample frames for the MOBISERV-AIIA dataset. The first row is the activity of drinking with a cup, and the second row is the activity of eating with a fork. Different columns mean different clothing scenarios for a person.

per class as training data, and use the remaining as test data. The MOBISERV-AIIA dataset, as shown in Figure 4.7, is the first publicly available dataset for the activity-based person identification task. It contains 12 persons performing eating and drinking activities with 2 clothing scenarios in four different days. The same as [149–151, 177], we adopt two kinds of the activities, drinking with a cup and eating with a fork, with 776 clips in total, and randomly choose one-day sequences as test samples, and use the rest as the training samples.

4.4.2 Comparisons with Existing Activity-based Human Identification Methods

We compare the proposed SMC methods including USMC and SSMC with the existing state-of-the-art approaches [150, 151, 177] for activity-based human identification. The results are listed in Table 4.1 and Table 4.2 for the Weizmann dataset and the MOBISERV-AIIA dataset, respectively. As shown, USMC is

comparable to GSCE in both of the datasets. SSMC outperforms USMC by slightly 0.75% by utilizing the class-based manifold structure in the Weizmann dataset, and SSMC further outperforms USMC by about 1.03% in the MOBISERV-AIIA dataset.

Table 4.1: Comparison with existing methods for the Weizmann dataset.

Method	Accuracy (%)
FCM [149]	78.36
SC [151]	89.55
GSCE	91.79
USMC	91.04
SSMC	91.79

Table 4.2: Comparison with existing methods on the MOBISERV-AIIA dataset.

Method	Accuracy (%)
FCM	52.85
SC	75.13
GSCE	77.20
USMC	78.76
SSMC	79.79

4.4.3 Comparisons with Existing Sparse Coding Methods

We compare the proposed SMC model including USMC and SSMC with existing sparse coding approaches [59, 90, 177] for activity-based human identification. Table 4.3 and Table 4.4 list the results in terms of accuracy for the Weizmann dataset and the MOBISERV-AIIA dataset, respectively. As shown, the proposed method outperforms other existing methods in both datasets. Further, SSMC outperforms USMC with additional class information.

Table 4.3: Comparison with existing sparse coding methods on the Weizmann dataset.

Method	Accuracy (%)
SC	89.55
LSC [90]	90.30
GraphSC [59]	90.30
GSCE	91.79
USMC	91.04
SSMC	91.79

Table 4.4: Comparison with existing sparse coding methods on the MOBISERV-AIIA dataset.

Method	Accuracy (%)
SC	75.13
LSC [90]	72.54
GraphSC [59]	72.54
GSCE	77.20
USMC	78.76
SSMC	79.79

4.4.4 Speed Comparison

We examine the efficiency of the conventional sparse coding solution solved by an efficient lasso-based solution, the feature-sign search [83], and the marginal regression approach [87]. We randomly generate 10000 data points drawn from the standard uniform distribution on the open interval $[0, 1]$; each of them lies in a 200-dimensional feature space and normalizes to have unit norm. The dictionary is learned by the online learning algorithm [146]. Table 4.5 shows the average results

Table 4.5: Speed comparison between lasso solution and marginal regression.

Method	Speed (second)
Lasso based Feature-sign Search	12.89 ± 0.02
Marginal Regression based Sure Independent Screen	0.25 ± 0.01

Table 4.6: Speed comparison between GSCE and SMC on the Weizmann dataset.

Method	Accuracy (%)	Speed (second)
GSCE	91.79	12.31
USMC	91.04	2.00

with 10 independent runs. As shown, the marginal regression solution can achieve 50 times faster than the lasso-based solution which is easily scale-up to deal with the large-scale problem. The improvement in speed is because the marginal regression based methods divide the optimization problems into several subproblems and then solve each regression problem by calculating its marginal coefficients followed by a thresholding step to shrinkage coefficients toward zero for the promotion of sparsity. On the other hand, the feature-sign search method continuously selects and updates potential candidates in an active set until reach the constraint.

We further investigate the efficiency of the proposed GSCE and USMC approaches on the Weizmann dataset. As shown in Table 4.6, we can see that GSCE slightly outperforms USMC by 0.75% in terms of correct recognition rate but the coding speed of USMC is about 6 times faster than that of GSCE. It is worth mentioning that the feature dimensions of USMC and GSCE are 3072-dim and 64-dim, respectively.

Table 4.7: Confusion matrix of our USMC method on the Weizmann dataset.

	Dar	Den	Eli	Ido	Ira	Len	Iyo	Mos	Sha
Dar	92	0	8	0	0	0	0	0	0
Den	0	94	6	0	0	0	0	0	0
Eli	0	0	88	0	0	0	6	6	0
Ido	0	0	0	89	11	0	0	0	0
Ira	0	0	0	0	94	0	0	0	6
Len	0	0	0	0	0	92	0	0	8
Iyo	0	0	8	0	0	0	92	0	0
Mos	0	0	0	7	0	0	0	86	7
Sha	0	0	0	8	0	0	0	0	92

4.4.5 Confusion Matrix

Table 4.7 and Table 4.8 show the confusion matrices of the USMC recognition results on the Weizmann and MOBISERV-AIIA datasets, accordingly. In Table 4.7, we can find that two persons, Den and Iyo, which are well-identified in the previous chapter, obtain 94% and 92% correct recognition rate here. Instead, Mos obtains 86% correct recognition rate which increases 7% compared to the previous GSCE method as shown in Table 3.3. In Table 4.8, we can find that three persons are well-identified, P6, P7 and P12, the same as those in Table 3.4. Three persons, P2, P3 and P5, are seriously mis-identified and obtain under 50% correct recognition rate, which are different from the results, P3, P5 and P8, of GSCE in Table 3.4. P3 still obtains the lowest correct recognition rate 25% which has 50% misidentified to P8 and 25% to P1. Figure 4.8 and Figure 4.9 show the sample visual words (codebooks).

Table 4.9 and Table 4.10 show the confusion matrices of the SSMC recognition results on the Weizmann and MOBISERV-AIIA datasets, accordingly. In Table 4.9,

Figure 4.8: Sample codebook pictures of the Weizmann dataset.

Figure 4.9: Sample codebook pictures of the MOBISERV-AIIA dataset.

Table 4.8: Confusion matrix of our USMC method on the MOBISERV-AIIA dataset.

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12
P1	94	0	0	0	0	0	6	0	0	0	0	0
P2	6	56	0	0	0	0	0	0	0	0	38	0
P3	25	0	25	0	0	0	0	50	0	0	0	0
P4	0	0	0	81	0	19	0	0	0	0	0	0
P5	13	0	6	0	50	25	0	6	0	0	0	0
P6	0	0	0	0	0	100	0	0	0	0	0	0
P7	0	0	0	0	0	0	100	0	0	0	0	0
P8	0	0	6	0	19	0	0	75	0	0	0	0
P9	0	0	6	0	0	6	0	0	88	0	0	0
P10	0	0	0	0	0	13	0	0	0	88	0	0
P11	0	13	0	0	0	0	0	0	0	0	88	0
P12	0	0	0	0	0	0	0	0	0	0	0	100

we can find that two persons, Ido and Iyo, are well-identified which improve the results of USMC with 89% and 92% for the correct recognition rate. However, Mos obtains only 79% in correct recognition rate, which has 7% misidentified to Ido, 7% misidentified to Iyo and 7% to Sha. As shown in Table 4.10, P6, P7 and P12 remain well-identified, and only two persons, P3 and P8, obtain below 50% correct recognition rate, both of which are 31%.

4.4.6 Analysis of Weighting Parameters

We investigate in this section the effect of the weighting parameters w^u , w^s and w^v of our proposed USMC and SSMC methods. Figure 4.10 shows the accuracy map of USMC with different pairs of w^u and w^v . Figure 4.11 shows the accuracy map of SSMC with different pairs of w^u and w^v . The brighter block obtains higher recognition accuracy using a specific pair of w^u and w^v in USMC or w^s and w^v

Table 4.9: Confusion matrix of our SSMC method on the Weizmann dataset.

	Dar	Den	Eli	Ido	Ira	Len	Iyo	Mos	Sha
Dar	88	0	8	0	0	4	0	0	0
Den	0	94	6	0	0	0	0	0	0
Eli	0	6	94	0	0	0	0	0	0
Ido	0	0	0	100	0	0	0	0	0
Ira	0	0	0	0	94	0	0	0	6
Len	0	0	0	0	0	92	0	0	8
Iyo	0	0	0	0	0	0	100	0	0
Mos	0	0	0	7	0	0	7	79	7
Sha	0	0	0	0	0	0	0	8	92

Table 4.10: Confusion matrix of our SSMC method on the MOBISERV-AIIA dataset.

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12
P1	94	0	0	0	0	0	6	0	0	0	0	0
P2	0	69	0	0	0	0	0	6	0	0	25	0
P3	13	0	31	0	0	0	0	44	0	0	6	6
P4	0	0	0	88	0	6	0	0	0	6	0	6
P5	0	0	0	0	81	19	0	0	0	0	0	0
P6	0	0	0	0	0	100	0	0	0	0	0	0
P7	0	0	0	0	0	0	100	0	0	0	0	0
P8	38	0	0	0	31	0	0	31	0	0	0	0
P9	0	0	0	0	6	0	0	0	94	0	0	0
P10	0	0	0	0	0	13	0	0	0	81	6	0
P11	6	6	0	0	0	0	0	0	0	0	88	0
P12	0	0	0	0	0	0	0	0	0	0	0	100

in SSMC, respectively. As shown in Figure 4.10, the brighter blocks of USMC lie in the left-upper corner, and the brightest block shows in the upper middle area. We can conclude that the neighborhood within each video-based manifold is more important than outside the manifold. This is because we utilize the temporal information of each video which shares the same person information so that similar features extracted from the same person can obtain similar visual words and further improve the performance. On the other hand, the brighter blocks of SSMC in Figure 4.11 lies in the left and middle areas. This change comes from the additional label information with person-based manifold structure. With adding person information, we construct the neighborhood not only within each video-based manifold but also the neighborhoods between manifolds under the same person-based manifold structure as shown in Figure 4.5. While SSMC slightly outperforms SSMC as shown in Table 4.1 to Table 4.4, the results of SSMC are more robust than those of USMC under different parameter selections as shown in Figure 4.10 and Figure 4.11.

4.4.7 Discussion

According to the above results, we have made the following observations:

- (1) The proposed unsupervised and supervised sparse manifold coding methods using marginal regression-based solutions for feature quantization can achieve higher or comparable recognition accuracy than the other state-of-the-art methods using k-means, Fuzzy c-means [149] or general sparse coding [151]. The proposed sparse manifold coding methods even outperform the graph-based sparse coding and embedding approach on the MOBISERV-AIIA dataset, and are comparable to that on the Weizmann dataset with

six times faster in terms of speed. The improvement in recognition is because the proposed unsupervised sparse manifold coding utilizes the intrinsic structure of a video clip which contains a series of movement frames and such kind of temporal information can help for recognition. On the other hand, the proposed supervised sparse manifold coding approach considers the hierarchical structure of manifolds including temporal information and class label which shows its robustness under different parameter selections. The improvement in efficiency in terms of computational cost is because we estimate sparsity using marginal regression algorithm.

- (2) The proposed sparse manifold coding methods for feature quantization can achieve better recognition performance than the other sparse coding related techniques such as SC, LSC, GraphSC and GSCE. One of the possible reason is that they ignore the spatial or the temporal information within an image or a video clip.

4.5 Summary

We have proposed in this chapter a novel sparse manifold coding framework for activity-based human identification to exploit the temporal information of a video and utilize the global geometrical structure in the unsupervised manner. We have also provided sparsity estimation using marginal regression and dictionary learning approaches to solve the proposed optimization problem. In addition, a supervised version has been proposed by forming a hierarchical manifold structure with an additional class-based manifold layer. Specifically, we have provided a NN-based sparsity approximation method to encode new data which can expedite the feature

quantization process. Our experimental results have demonstrated the efficacy of the proposed methods. Further, it is worth mentioning that the proposed method appears to be the first algorithm that formally addresses the local and global geometrical structures into the sparse coding framework.

Figure 4.10: Color map of correct recognition rate of our USMC versus different parameter settings on the Weizmann dataset. The brighter block obtains higher recognition accuracy.

Figure 4.11: Color map of correct recognition rate of our SSMC versus different parameter settings on the Weizmann dataset. The brighter block obtains higher recognition accuracy.

Chapter 5

Marginal-Lasso Coding

5.1 Introduction

Sparse coding has been successfully applied to many machine learning and computer vision tasks, including image classification [59, 89, 90], face recognition [58, 60], abnormal event detection [61], activity-based person identification [151, 177], etc. Sparse coding refers to a feature quantization mechanism which obtains a codebook to encode each input signal as a sparse histogram feature based on the linear combination of a few visual words. Due to its soft assignment principal, it can make the quantization error much smaller [91, 151]. Moreover, sparse coding is more robust to the noise and can be easily incorporated into the bag-of-feature (BoF) framework [91].

The least absolute shrinkage and selection operator which is also known as lasso [77] in short is a popular sparse coding model in statistics signal processing which has wildly used in sparsity estimation. It simultaneously minimizes the quantization error, and impose an l_1 penalty with a regularization parameter λ , which controls the sparsity of the coefficients. The regularization parameter should be chosen

adaptively to minimize the quantization error, but, in general, it assigns a sole regularization parameter to all inputs. While there exist many efficient algorithms [81–83], however, finding the lasso solutions remain a computational task with the complexity $O(p^3 + dp^2)$ where d refers to the input feature dimension, and p refers to the size of visual words in the dictionary. Thus, it can hardly support large scale data analysis efficiently and effectively [87, 88, 145, 146].

Recently, marginal regression has been revisited and shown its efficient performance for the visual word selection and sparse code estimation [87, 88, 145]. For each feature, it calculates the correlation with each visual word and imposes a tuning parameter to achieve the sparsity. While marginal regression is simple and fast with the complexity $O(dp)$, it usually considers and utilizes a fixed tuning parameter which is not always a good threshold to fit each feature well and could result in a large quantization error since some coefficients may shrink too much, but some may include too much noise. Figure 5.1 (a) shows an example with two feature distributions of marginal coefficients denoted by s_A and s_B . If we select $t = 0.75$ to be the threshold so that only those coefficients whose values are larger than 0.75 are kept and others are set to zero to achieve sparse representation, the marginal coefficients of s_B may shrink too much and obtain a small l_1 energy norm. Instead, if we select $t = 0.55$ to be the threshold, the marginal coefficients of s_A may include too much noise and obtains a large l_1 norm. Hence, how to determine a suitable tuning value becomes a typical problem in featuring coding. To address this, the sure independent screening (SIS) [87] method imposes an l_1 penalty and an energy constraint E to each marginal regression coefficients vector. It sorts coefficients in terms of their absolute values and selects the top k coefficients whose l_1 norm is bounded by E so that each sparse code may be represented by similar sparsity level as shown in 5.1 (b). However, this method may also cause a large quantization error

Figure 5.1: Distributions of two sampled marginal regression coefficients, s_A and s_B . (a) With a fixed tuning parameter $t = 0.75$, some of the coefficients may shrink too much such as s_B ; instead, with a fixed regularization parameter $t = 0.55$, some of the coefficients may include too much noise such as s_A . (b) With an l_1 energy constraint $\|s\|_1 \leq E$, each feature can be represented by similar sparsity level.

because marginal regression works well when visual words have low correlation in the dictionary [145, 146]; however, in general, they are usually highly correlated and thus may fail to clarify the relationship between visual words and input signals.

While there are many works focusing on sparse coding and dictionary learning, none of them has exploited the advantages of the marginal regression and the lasso simultaneously to provide more efficient and effective solutions. To this end, we propose in this chapter an efficient sparsity estimation approach using Marginal-Lasso Coding (MLC) to quantize each feature into a small set of visual words using marginal regression combined with the lasso framework. Our model simultaneously represents each sparse code with similar sparsity level and shrinks individual coefficients to alleviate the bias of sparse codes caused by the highly correlated visual words. Moreover, our approach adaptively and automatically determines the shrinking regularization parameter λ for each individual feature and further designs a self-ratio energy constraint of sparsity level different from the traditional constraint with a case-dependent chosen value.

Our contributions in this chapter includes the following:

- (1) We propose an efficient Marginal-Lasso Coding (MLC) framework for feature quantization with sparsity estimation, regularization parameter selection and dictionary learning which is a more stable soft-thresholding technique compared to the traditional marginal regression methods dealing with hard-thresholding solutions and a more efficient way compared to the lasso solution in terms of computational cost.
- (2) The proposed method determines the regularization parameter automatically and adaptively for each feature, and provides a self-ratio energy bound selection. In existing work, regularization parameter is usually assigned empirically and all features share the same regularized value to solve the objective function. Instead, in our proposed MLC method, each feature vector

obtains an optimized regularized value for coding. Further, the energy bound is usually a case-dependent chosen value, but, in our method, we propose to use a sparsity ratio ranging from 0 to 1, which is a more general way for bound selection.

- (3) We successfully apply the proposed method to various classification and recognition tasks such as image classification, scene classification, handwritten digits clustering, action recognition and activity-based human identification for feature quantization.

The rest of the chapter is organized as follows: Section 5.2 describes the background of marginal regression for feature quantization. Section 5.3 details the proposed marginal-lasso coding method. Section 5.4 describes the experiments on the task of activity-based human identification. Section 5.5 presents the experiments on handwritten digit recognition. Section 5.6 shows the experiments on image classification. Section 5.7 provides the experiments on action recognition. Finally, we conclude the chapter in Section 5.8.

5.2 Marginal Regression for Feature Quantization

Given an input feature x , we aim to quantize it into a small subset of visual words with corresponding non-zero weights to minimize the quantization error. To address this, we consider a problem of estimating p -dimensional vector s with a column-wisely normalized dictionary $U = [u_1, \dots, u_p] \in \mathbb{R}^{d \times p}$ which consists of a set of visual words denoting by each column u_m so that the linear combination of U and s

can minimize the quantization error to the original signal. Note that the dictionary U is over-complete with $p \gg d$. It follows that:

$$x = Us + z \quad (5.1)$$

where $x \in \mathbb{R}^d$ is an input feature, $s \in \mathbb{R}^p$ is a coefficient vector, $U \in \mathbb{R}^{d \times p}$ is a column-wisely normalized dictionary and $z \in \mathbb{R}^d$ is a noise vector. Then, the optimization problem of minimizing quantization error can be rewritten as follows:

$$\min_s \|x - Us\|_2^2 \quad (5.2)$$

To estimate the sparse code s via marginal regression, we first compute the component-wise marginal regression coefficients $\hat{a}$:

$$\hat{a} \equiv U^T x \quad (5.3)$$

where the m th element of $\hat{a}$ denotes by $\hat{a}^{(m)} = u_m^T x$. Then, we threshold the coefficients in terms of their absolute values using a tuning parameter $t > 0$ [88, 147] so that the coefficients whose absolute values are larger than t are kept and others are set to zero. The mathematical expression can be written as follows:

$$\hat{s}^{(m)} = \begin{cases} \hat{a}^{(m)} & \text{if } |\hat{a}^{(m)}| > t \\ 0 & \text{otherwise} \end{cases} \quad (5.4)$$

where $\hat{s} = \hat{a}$ as $t = 0$. However, we show in Figure 5.1 (a) that utilizing a fixed tuning parameter may cause a large quantization error since it is not always a good threshold to fit each feature well. For example, one may include too much noise or one may shrink too much. Instead, we can select the top k large coefficients in terms of their absolute values whose l_1 norm is bounded by a constraint E [87, 145] so that each sparse code can be represented under a similar sparsity level as shown in 5.1 (b). The equation can be reformulated as follows:

$$\hat{s}^{(m)} = \begin{cases} \hat{a}^{(m)} & \text{if } m \in Q \\ 0 & \text{otherwise} \end{cases} \quad (5.5)$$

where

$$Q = \{J_1, \dots, J_r : r \leq p : \sum_{z=1}^r |\hat{a}_i^{(J_z)}| \leq E\} \quad (5.6)$$

We sort the coefficients in terms of their absolute values in descending order which denotes by indexes $J_1, \dots, J_p$ where $|\hat{s}^{(J_1)}| > |\hat{s}^{(J_2)}| > \dots > |\hat{s}^{(J_p)}|$. It is worth mentioning that the approaches in both Eq. (5.4) and Eq. (5.5) are hard-thresholding-type methods with a specific tuning parameter.

5.3 Marginal-Lasso Coding

We propose a novel marginal-lasso coding framework for sparsity estimation, automatic and adaptive regularization parameter selection and dictionary learning for the problem of feature quantization.

5.3.1 Motivation

While the lasso model and its extensions have been wildly used in many visual applications for feature quantization and sparse representation in recent years, it still requires further improvement on sparsity estimation, dictionary learning and regularization parameter selection. For example, a number of sparsity estimation methods have been proposed, but most of them solve the general sparse coding method with the lasso solution. While there exist efficient sparsity estimation methods such as least angle regression [79], gradient descent [81, 82], and feature-sign search [83], it still requires a large computational cost. For example, the feature-sign search method [59, 83, 90] is a popular and efficient solution for sparsity estimation. It continuously selects and updates potential candidates in an active set to maintain these nonzero coefficients and their corresponding sign until the constraint is met.

However, with a large dictionary size, the active set would become large and the feature-sign searching process may be hard to terminate [145, 146]. While these methods aim to obtain a small quantization error, they have a large computational cost [87, 88].

On the other hand, recently, marginal regression has been shown its efficiency for sparsity estimation. For example, Fan and Lv [87] proposed a sure independent screening (SIS) method using marginal regression to obtain sparse codes efficiently and can easily deal with large-scale problems. After that, Genovese *et al.* [88] provided a statistical comparison of the lasso and marginal regression and have shown that the computational complexity of marginal regression based methods are two orders faster than that of the lasso-based methods, even though there are some efficient algorithms using gradient descent for lasso. While marginal regression has shown its effectiveness, there still exist some limitations:

- (1) Marginal regression works well when the columns of the dictionary have low correlation [145, 146]. However, in general, the dictionary is over-complete and, therefore, columns of the dictionaries are usually highly correlated. Thus, in a linear regression model, their coefficients may become poorly determined and exhibit high variance.
- (2) The regularization parameter t in Eq. (5.4) should be selected adaptively according to each input feature for sparse representation. For an universal parameter t , one may include too much noise but the other may have no information as shown in Figure 5.1 (a).

While there are many works working on sparse coding and dictionary learning, none of them has exploited the advantages of the marginal regression and the lasso

simultaneously to provide more efficient and effective solutions. In addition, most of the models use a specific regularization parameter which may also cause a large quantization error. Motivated by the above arguments, we introduce a more efficient framework to incorporate marginal regression coefficients into the sparse coding model and propose a new sparsity estimation algorithm, regularization parameter selection and dictionary learning approach.

5.3.2 Marginal-Lasso Coding

We exploit the advantages of the marginal regression and the lasso simultaneously to provide more efficient and effective solutions. Specifically, we propose a marginal-lasso framework to integrate marginal regression coefficients and an l_1 -norm penalty to minimize the quantization error using conventional marginal regression methods computed from an over-complete dictionary. We consider the following arguments:

- (1) Due to the correlated columns of the over-complete dictionary, the coefficients should be shrunk towards to zero to alleviate the bias of sparse codes caused by the highly correlated visual words using marginal regression.
- (2) The regularization parameter should be adaptively for each feature to minimize individual estimate of expected quantization error.

To this end, we introduce a Marginal-Lasso Coding (MLC) method with adaptively regularization parameter λ to better characterize input features and provide a theoretically soft-thresholding-type solution. Similar to the existing work, we first compute the marginal regression coefficients of each feature via Eq.

(5.3), $\hat{a} \equiv U^T x$. Then, the objective function of the marginal-lasso coding can be formulated as follows:

$$\min_{U, s, \lambda} \frac{1}{2} \|U^T x - s\|_2^2 + \lambda \|s\|_1 \quad (5.7)$$

$$\text{subject to } \|s\|_1 \leq E$$

$$\|u_m\|_2^2 \leq 1, \quad \forall m$$

The first term is the quantization error between coefficients and the corresponding sparse code, and the second term is an l_1 penalty with an adaptive regularization parameter $\lambda \geq 0$ which controls the threshold selection and the shrinkage of the coefficients for each input feature x . For example, a large λ provide a large tuning value and shrink coefficients seriously towards zero. In other words, if $\lambda = 0$, there exist no cut-off points and $s = \hat{a} = U^T x$. The sparsity constraint E bounds each sparse code to the similar sparsity level. In this optimization problem, we aim to estimate a sparse code s and a regularization parameter λ which minimize the quantization error and simultaneously shrink the coefficients under a certain sparsity level. When considering multiple features, the optimization problems can be refined as follows:

$$\min_{\substack{U \\ s_1, \dots, s_n \\ \lambda_1, \dots, \lambda_n}} \frac{1}{2} \|U^T X - S\|_F^2 + \sum_{i=1}^n \lambda_i \|s_i\|_1 \quad (5.8)$$

$$\text{subject to } \|s_i\|_1 \leq E, \quad \forall i$$

$$\|u_m\|_2^2 \leq 1, \quad \forall m$$

where $X = [x_1, \dots, x_n] \in \mathbb{R}^{d \times n}$ be a set of features in a d -dimensional feature space, and $S = [s_1, \dots, s_n] \in \mathbb{R}^{p \times N}$ be a set of corresponding sparse codes in a

p -dimensional feature space. In Eq. (5.8), we can clearly see that the regularization parameter λ_i is adaptively selected according to the input feature x_i . Different from λ , the energy level E is a global constraint which bound each sparse energy under this l_1 -norm level. Since the sparsity level E is a specific value and needs to be determined, we provide another self-ratio level e such that each feature can obtain its sparse code by proportionally and automatically shrinking their marginal regression coefficients to the desired level. The optimization problem can be reformulated as follows:

$$\min_{\substack{U \\ s_1, \dots, s_n \\ \lambda_1, \dots, \lambda_n}} \frac{1}{2} \|U^T X - S\|_F^2 + \sum_{i=1}^n \lambda_i \|s_i\|_1 \quad (5.9)$$

$$\text{subject to } \frac{\|f_{\lambda_i}(U^T x_i)\|_1}{\|U^T x_i\|_1} \leq e, \quad \forall i$$

$$\|u_m\|_2^2 \leq 1, \quad \forall m$$

where $f_{\lambda_i}(\hat{a}_i)$ refers to a hard-thresholding function, given as:

$$f_{\lambda_i}(\hat{a}_i^{(k)}) = \begin{cases} \hat{a}_i^{(k)} & \text{if } |\hat{a}_i^{(k)}| > \lambda_i \\ 0 & \text{otherwise} \end{cases} \quad (5.10)$$

Both constraints in Eq. (5.8) and Eq. (5.9) are global constraints which bound each l_1 -norm or l_1 -norm ratio of a sparse code to a specific level, respectively. The formulations in Eq. (5.8) and Eq. (5.9) are not convex on U , S and $\lambda = [\lambda_1, \dots, \lambda_n]$ simultaneously, but it is convex on U when S and $\lambda = [\lambda_1, \dots, \lambda_n]$ are fixed. Further, it is convex on S and $\lambda = [\lambda_1, \dots, \lambda_n]$ as well when U is fixed. Therefore, we introduce the sub-optimized problems in the following sections. We discuss in Section 5.3.3 the case when U is fixed and in Section 5.3.4 the case when S and $\lambda = [\lambda_1, \dots, \lambda_n]$ are fixed.

5.3.3 Sparsity Estimation and Parameter Selection

Given a dictionary U , the objective of Eq. (5.7) is convex on S and $\lambda_1, \dots, \lambda_n$ simultaneously. Hence, we can optimize S and $\lambda_1, \dots, \lambda_n$ by the following optimization problem:

$$\min_{\substack{s_1, \dots, s_n \\ \lambda_1, \dots, \lambda_n}} \frac{1}{2} \|U^T X - S\|_F^2 + \sum_{i=1}^n \lambda_i \|s_i\|_1 \quad (5.11)$$

$$\text{subject to } \|s_i\|_1 \leq E, \forall i$$

which is equivalent to the follow of vector form:

$$\min_{\substack{s_1, \dots, s_n \\ \lambda_1, \dots, \lambda_n}} \sum_{i=1}^n \frac{1}{2} \|U^T x_i - s_i\|_2^2 + \lambda_i \|s_i\|_1 \quad (5.12)$$

$$\text{subject to } \|s_i\|_1 \leq E, \forall i$$

Since Eq. (5.12) can be separated into n sub-problems, we can rewrite the objective of Eq. (5.12) with respect to s_i and λ_i as follows:

$$\min_{s_i, \lambda_i} \frac{1}{2} \|U^T x_i - s_i\|_2^2 + \lambda_i \|s_i\|_1 \quad (5.13)$$

$$\text{subject to } \|s_i\|_1 \leq E$$

With the replacement of $\hat{a}_i = U^T x_i$, we rewrite Eq. (5.13) as:

$$\min_{s_i, \lambda_i} \frac{1}{2} \|\hat{a}_i - s_i\|_2^2 + \lambda_i \|s_i\|_1 \quad (5.14)$$

$$\text{subject to } \|s_i\|_1 \leq E$$

Minimizing the Eq. (5.14) is equivalent to minimizing individual element errors. Thus, we can rewrite the objective as the sum of the elements:

$$\min_{s_i, \lambda_i} \sum_{m=1}^p \frac{1}{2} ((\hat{a}_i^{(m)} - s_i^{(m)})^2 + \lambda_i |s_i^{(m)}|) \quad (5.15)$$

$$\text{subject to } \sum_{m=1}^p |s_i^{(m)}| \leq E$$

Since the marginal regression coefficients are computed independently for each element with a fixed regularization parameter λ_i , the problem decomposes into p separate optimization tasks. For each element m , the optimization problem can be defined as follows:

$$\min_{s^{(m)}} \frac{1}{2} (\hat{a}_i^{(m)} - s^{(m)})^2 + \lambda_i |s^{(m)}| \quad (5.16)$$

Then, by solving the above optimization problem, we can obtain a soft-thresholding-type optimal solution of each sparse code element $s_i^{*(m)}$:

$$s_i^{*(m)} = \begin{cases} \hat{a}_i^{(m)} - \lambda_i & \text{if } \hat{a}_i^{(m)} > \lambda_i \\ \hat{a}_i^{(m)} + \lambda_i & \text{if } \hat{a}_i^{(m)} < -\lambda_i \\ 0 & \text{otherwise} \end{cases} \quad (5.17)$$

$$= \text{sign}(\hat{a}_i^{(m)}) (|\hat{a}_i^{(m)}| - \lambda_i)_+$$

where $\text{sign}(j)$ refers to the sign of j , and $(l)_+$ means to keep the value when l is positive and set it to zero when l is negative. The l_1 -norm penalty of $\|s_i^*\|_1$ can be defined as the sum of absolute elements $\|s_i^*\|_1 = \sum_{m=1}^p |s_i^{*(m)}| = \sum_{m=1}^p |\text{sign}(\hat{a}_i^{(m)}) (|\hat{a}_i^{(m)}| - \lambda_i)_+|$. In Eq. (5.17), the regularization parameter λ_i controls the element selection and the shrinkage of coefficients simultaneously. If λ_i is zero, then s_i^* equals to the original marginal regression coefficients $\hat{a}_i$, and thus $\|s_i^*\|_1 = \|\hat{a}_i\|_1$. Instead, if $\lambda_i > 0$, then some coefficients will be directly set to

zero and others will be shrunk towards zero automatically so that $\|s_i^*\|_1 < \|\hat{a}_i\|_1$. To maximize $\|s_i^*\|_1$ bounded by the sparsity constraint E , the individual λ_i^* can be calculated as follows:

$$\begin{aligned}\lambda_i^* &= \arg \max_{\lambda_i} \|s_i^*\|_1 \\ &= \arg \max_{\lambda_i} \sum_{m=1}^p |s_i^{(m)}| \\ &= \arg \max_{\lambda_i} \sum_{m=1}^p |\text{sign}(\hat{a}_i^{(m)}) (|\hat{a}_i^{(m)}| - \lambda_i)_+|\end{aligned}\tag{5.18}$$

$$\text{subject to } \|s_i^*\|_1 \leq E$$

For the self-ratio constraint in Eq. (5.9), similar to Eq. (5.18), the optimized regularization parameter λ_i^* can be computed with the self-ratio constraint:

$$\lambda_i^* = \arg \max_{\lambda_i} \sum_{m=1}^p |\text{sign}(\hat{a}_i^{(m)}) (|\hat{a}_i^{(m)}| - \lambda_i)_+|\tag{5.19}$$

$$\text{subject to } \frac{\|f_{\lambda_i}(\hat{a}_i)\|_1}{\|\hat{a}_i\|_1} \leq e$$

According to Eq. (5.18) or Eq. (5.19), we can estimate each element of a sparse code s_i^* via a corresponding regularization parameter λ_i^* whose l_1 -norm is bounded by the energy constraint E or the self-ratio constraint e as follows:

$$s_i^{*(m)} = \text{sign}(\hat{a}_i^{(m)}) (|\hat{a}_i^{(m)}| - \lambda_i^*)_+\tag{5.20}$$

Algorithm 8 lists the details of the proposed sparsity estimation and regularization parameter selection of the marginal-lasso framework for feature quantization.

5.3.4 Dictionary Learning

The dictionary U is a set of normalized visual words denoting by each column u_m . To learn the dictionary in the training phase, we iteratively optimize U and

Algorithm 8 Efficient Sparsity Estimation and Regularization Parameter Selection via Marginal-Lasso Coding

Require: Data $X = [x_1, \dots, x_n] \in \mathbb{R}^{d \times n}$, a learned dictionary $U = [u_1, \dots, u_p] \in \mathbb{R}^{d \times p}$, and a specific value of the sparsity constraint E or a self-ratio constraint e

Ensure: Sparse coefficients $S = [s_1, \dots, s_n] \in \mathbb{R}^{p \times n}$

- 1: For $x_i \forall i$, compute the marginal regression coefficients

$$\hat{a}_i^{(m)} = \frac{u_m^T x_i}{\|u_m\|_2^2} \quad \forall m$$

- 2: Calculate the regularization parameter λ_i for each $x_i \forall i$ based on the sparsity constraint E of Eq. (5.7) and Eq. (5.18)

$$\begin{aligned} \lambda_i^* &= \arg \max_{\lambda_i} \|s_i^*\|_1 = \arg \max_{\lambda_i} \sum_{k=1}^p |\text{sign}(\hat{a}_i^{(k)}) (|\hat{a}_i^{(k)}| - \lambda_i)_+| \\ &\text{subject to } \|s_i\|_1 \leq E \end{aligned}$$

Or calculate the regularization parameter λ_i for each $x_i \forall i$ based on the self-ratio constraint e in Eq. (5.9)

$$\begin{aligned} \lambda_i^* &= \arg \max_{\lambda_i} \|s_i^*\|_1 = \arg \max_{\lambda_i} \sum_{k=1}^p |\text{sign}(\hat{a}_i^{(k)}) (|\hat{a}_i^{(k)}| - \lambda_i)_+| \\ &\text{subject to } \frac{\|f_{\lambda_i}(\hat{a}_i)\|_1}{\|\hat{a}_i\|_1} \leq e \end{aligned}$$

- 3: Obtain sparse codes

$$\begin{aligned} s_i^{*(m)} &= \begin{cases} \hat{a}_i^{(m)} - \lambda_i^* & \text{if } \hat{a}_i^{(m)} > \lambda_i^* \\ \hat{a}_i^{(m)} + \lambda_i^* & \text{if } \hat{a}_i^{(m)} < -\lambda_i^* \\ 0 & \text{otherwise} \end{cases} \\ &= \text{sign}(\hat{a}_i^{(m)}) (|\hat{a}_i^{(m)}| - \lambda_i^*)_+ \end{aligned}$$

S by obtaining S with a fixed U and updating U based on a given S . The sparsity estimation S via the marginal-lasso model has been introduced in the previous sections. In this section, we introduce the dictionary learning method. The optimization problem of the dictionary learning can be formulated as follows:

$$\begin{aligned} \min_U \quad & \|U^T X - S\|_F^2 \\ \text{subject to} \quad & \|u_m\|_2^2 \leq 1, \forall m \end{aligned} \quad (5.21)$$

Eq. (5.21) aims to minimize the error between marginal regression coefficients $U^T X$ and a set of sparse codes S . Instead of solving Eq. (5.21), we transform the optimization problem as follows:

$$\begin{aligned} \min_U \quad & \|X - US\|_F^2 \\ \text{subject to} \quad & \|u_m\|_2^2 \leq 1, \forall m \end{aligned} \quad (5.22)$$

With a set of sparse codes S , we aim to minimize the quantization error between input features X and the linear combination of the dictionary U and the corresponding weights S of the visual words. Since marginal regression works well when the columns in the dictionary have low correlation [87, 88, 145, 146]. The correlation between two visual words can be expressed as follows:

$$\min_{i \neq j} |u_i^T u_j| \quad (5.23)$$

The matrix form can be formulated as follows:

$$\min \|U^T U - I\|_F^2 \quad (5.24)$$

where $I \in \mathbb{R}^{p \times p}$ is a $p \times p$ identity matrix. Therefore, we simultaneously minimize the quantization errors and the correlation between pairs of visual words as follows:

$$\begin{aligned} \min_U \quad & \|X - US\|_F^2 + \gamma \|U^T U - I\|_F^2 \\ \text{subject to} \quad & \|u_m\|_2^2 \leq 1, \forall m \end{aligned} \quad (5.25)$$

where γ is a regularization parameter. This optimization problem can be solved via the first-order gradient descent method [145, 146]. We first define the objective function $F(U)$ and its gradient respect to U at the q iteration denoted by $\nabla F(U_q)$ as follows:

$$F(U) = \|X - US\|_F^2 + \gamma\|U^T U - I\|_F^2 \quad (5.26)$$

and

$$\nabla F(U_q) = 2(U_q S S^T - X S^T) + 4\gamma(U_q U_q^T U_q - U_q) \quad (5.27)$$

Then, the first-order gradient descent method can be given as follows:

$$U_{q+1} = \Pi_U\{U_q - \beta \nabla F(U_q)\} \quad (5.28)$$

where parameter β is the step size, and Π_U is the projection function which maps each column u_k to the l_2 -norm unit ball. More specifically, the dictionary is learned iteratively by finding the local minimum along the gradient direction based on the small step size and normalizing each column vector with length smaller or equal to one until convergence. The sparsity estimation and the dictionary learning process is shown in Algorithm 9.

5.4 Activity-based Human Identification

The activity-base human identification task aims to recognize the identity of a person based on his/her behavior. We perform the experiments on two publicly available databases, Weizmann [176] and MOBISERV-AIIA [150]. The Weizmann dataset contains 215 clips with 9 persons performing 9 different activities including run, skip, walk, gallop-sideways, jumping-jack, jumping-forward-on-two-legs, jump in place-on-two-legs, wave-one-hand, and wave-two-hands, respectively. Figure 5.2 shows

Algorithm 9 Dictionary Learning via Marginal-Lasso Coding

Require:

Require: Given a set of training samples $X = [x_1, \dots, x_n] \in \mathbb{R}^{d \times n}$, regularization parameters β and γ , and the sparse level constraint E or e .

Ensure:

Ensure: A dictionary $U \in \mathbb{R}^{d \times p}$, sparse coefficients $S = [s_1, \dots, s_n] \in \mathbb{R}^{p \times n}$

- 1: Initialization: For $x_i \forall i$, zero coefficients of s_i and an initial dictionary $U \in \mathbb{R}^{d \times p}$.
- 2: **repeat**
- 3: Obtain sparse codes S using Algorithm 8 with the latest dictionary U .

$$\begin{aligned} \min_{\substack{s_1, \dots, s_n \\ \lambda_1, \dots, \lambda_n}} \quad & \|U^T X - S\|_F^2 + \sum_{i=1}^n \lambda_i \|s_i\|_1 \\ \text{subject to} \quad & \|s_i\|_1 \leq E, \forall i \end{aligned}$$

or

$$\begin{aligned} \min_{\substack{s_1, \dots, s_n \\ \lambda_1, \dots, \lambda_n}} \quad & \|U^T X - S\|_F^2 + \sum_{i=1}^n \lambda_i \|s_i\|_1 \\ \text{subject to} \quad & \frac{\|f_{\lambda_i}(U^T x_i)\|_1}{\|U^T x_i\|_1} \leq e, \forall i \end{aligned}$$

- 4: **for** $q = 1 : Q$ **do**
- 5: Compute U_q based on the sparse codes obtained from the Step 3 so that

$$\begin{aligned} \min_U \quad & \|X - US\|_F^2 + \gamma \|U^T U - I\|_F^2 \\ \text{subject to} \quad & \|u_m\| \leq 1, \forall m \end{aligned}$$

- 6: **end for**
 - 7: $U = U_Q$
 - 8: **until** convergence
-

Figure 5.2: Sample frames of the Weizmann dataset. From left to right are sample image frames of handwaving, jumping-forward-on-two-legs, jump-in-place and gallop-sideways.

some sample frames. The MOBISERV-AIIA dataset contains 12 persons performing eating and drinking activities under two clothing scenarios in four different days with 776 clips in total. Some sample frames are shown in Figure 5.3. In the datasets, each clip contains a person performing one activity at least one cycle. For each image frame, we extract a binary human body mask by background subtraction and thresholding. To obtain the region of interest (ROI), we align and resize each mask into the size of 64×48 , similar to that in [140, 149–151, 177]. It is worth mentioning that we initialize the dictionary as the results of online dictionary learning [146] to expedite the convergence. For the Weizmann dataset, we randomly select one clip per action per class as training data, and use the remaining as test data. For the MOBISERV-AIIA dataset, we randomly choose one-day sequences as test samples, and use the rest as the training samples. The parameters are determined via cross validation. After feature quantization using the proposed MLC approach, we represent each video clip using a maximum pooling function and adopt a nearest-neighbor classifier for recognition.

Figure 5.3: Sample frames of the AIIA dataset. From left to right, every two samples are the same clothing scenario with two activities, eating with a folk and drinking with a cup.

5.4.1 Comparisons with Existing Marginal-based Sparsity Estimation Methods

We first compare the proposed MLC method with other sparsity estimation methods using marginal regression for activity-based human identification as follows:

- MR-Hard: the sparsity estimation method that uses marginal regression with simple hard thresholding.
- MR-Soft: the sparsity estimation method that uses marginal regression with simple soft thresholding.
- MR-SIS: the sparsity estimation method that uses marginal regression with hard thresholding under a specific l_1 penalty level.
- MLC: the proposed sparsity estimation method.

Table 5.1 shows the results in terms of recognition accuracy under the 10% self-ratio sparsity constraint. As shown, the proposed MLC outperforms other methods for both data sets. In addition, the soft-thresholding methods such as MLC and MR-Soft work better than the hard-thresholding methods such as MR-Hard and MR-SIS.

5.4.2 Comparisons with Existing Activity-based Human Identification Methods

We compare the proposed feature quantization method with other existing methods for activity-based human identification. Table 5.2 shows the results in terms of recognition accuracy. It is worth mentioning that we denote the proposed unsupervised sparse manifold coding approach by MR-USMC which uses marginal regression for sparsity estimation discussed in Chapter 4. In this chapter, we proposed a new marginal-lasso regression solution algorithm for sparsity estimation to improve the traditional marginal regression method in terms of robustness, and we apply MLC to USMC for sparsity estimation denoted by MLC-USMC.

The proposed MLC method outperforms FCM on both datasets and is comparable to SC on the Weizmann dataset and outperforms SC on the other dataset. Compared to GSCE and MR-USMC methods where both of them consider the consistency of geometrical structure between pairs of sparse codes and their corresponding original features, MLC is lightly worse than them on the Weizmann dataset, but comparable on the MOBISERV-AIIA dataset. It is worth mentioning that the proposed MLC method does not consider the geometrical structure between features. Instead, MLC estimates each sparsity individually and independently. When we consider the geometrical structure using MLC denoted by MLC-USMC, it outperforms others on both Weizmann and MOBISERV-AIIA datasets. One of the possible reasons is that the proposed MLC algorithm integrates the advantages between conventional marginal regression methods and the lasso-based approaches.

Table 5.1: Recognition accuracy (%) of different marginal regression methods on the Weizmann and the MOBISERV-AIIA datasets.

Method \ Dataset	Weizmann	MOBISERV-AIIA
MR-Hard	69.40	62.69
MR-Soft	79.85	70.47
MR-SIS	77.61	67.88
MLC	89.55	77.72

Table 5.2: Recognition accuracy (%) of different methods on the Weizmann and MOBISERV-AIIA datasets.

Method \ Dataset	Weizmann	MOBISERV-AIIA
FCM	78.36	52.85
SC	89.55	75.13
GSCE	91.79	77.20
MR-USMC	91.04	78.76
MLC	89.55	77.72
MLC-USMC	92.54	79.79

Table 5.3: Comparison with existing sparse coding methods.

Method \ Dataset	Weizmann	MOBISERV-AIIA
SC	89.55	75.13
LSC [90]	90.30	72.54
GraphSC [59]	90.30	72.54
GSCE	91.79	77.20
MR-USMC	91.04	78.76
MLC	89.55	77.72
MLC-USMC	92.54	79.79

5.4.3 Comparisons with Existing Sparse Coding Methods

We compare the proposed MLC model with existing sparse coding approaches for activity-based human identification. Table 5.3 lists the results in terms of accuracy on the Weizmann dataset and the MOBISERV-AIIA dataset. As shown, the proposed MLC method is comparable with SC, LSC and GraphSC on the Weizmann dataset and outperforms them on the MOBISERV-AIIA dataset. It is worth mentioning that LSC and GraphSC consider the geometrical structure between features but the proposed MLC does not. When we compare the proposed MLC-USMC algorithm with the consideration of the geometrical structure between features, it outperforms existing methods as shown in Table 5.3.

5.4.4 Confusion Matrix

Table 5.4 and Table 5.5 show the confusion matrices of the recognition results on the Weizmann and MOBISERV-AIIA datasets, accordingly. In Table 5.4, we can find that two persons are well-identified, Len and Iyo, and the person Ido has the

Figure 5.4: Sample codebook pictures of the Weizmann dataset.

Figure 5.5: Sample codebook pictures of the MOBISERV-AIIA dataset.

Table 5.4: Confusion matrix of our MLC method on the Weizmann dataset.

	Dar	Den	Eli	Ido	Ira	Len	Iyo	Mos	Sha
Dar	96	0	0	0	4	0	0	0	0
Den	0	94	6	0	0	0	0	0	0
Eli	6	6	88	0	0	0	0	0	0
Ido	0	0	0	82	0	0	0	18	0
Ira	6	0	6	0	89	0	0	0	0
Len	0	0	0	0	0	100	0	0	0
Iyo	0	0	0	0	0	0	100	0	0
Mos	0	0	0	0	0	0	0	92	8
Sha	0	0	0	0	0	8	0	0	92

Table 5.5: Confusion matrix of our MLC method on the MOBISERV-AIIA dataset.

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12
P1	80	0	20	0	0	0	0	0	0	0	0	0
P2	0	68	0	0	0	0	0	11	0	0	21	0
P3	0	0	100	0	0	0	0	0	0	0	0	0
P4	0	0	0	100	0	0	0	0	0	0	0	0
P5	0	0	0	0	59	0	0	35	6	0	0	0
P6	0	0	0	8	23	65	0	0	4	0	0	0
P7	0	0	0	0	0	0	100	0	0	0	0	0
P8	0	6	44	0	0	0	0	44	0	6	0	0
P9	0	0	0	0	0	0	0	0	100	0	0	0
P10	0	0	0	7	0	0	0	0	0	93	0	0
P11	0	14	0	0	0	0	0	0	0	0	86	0
P12	0	0	0	0	0	0	0	0	0	0	0	100

Table 5.6: Recognition accuracy (%) of different methods on the USPS dataset.

Method \ Size of training set	100	500	1000	2000	5000	7291
OR	81.7	88.4	90.1	91.4	92.3	93.8
SC	79.6	88.0	90.4	91.8	92.9	94.5
GraphSC	83.2	90.2	92.0	92.6	93.6	95.0
MLC	74.5	91.1	92.9	94.2	95.3	94.0

lowest correct recognition rate which has 14% misidentified to Sha and 7% to Ido. In Table 5.5, we can find that three persons are well-identified, P6, P7 and P12, and several persons are seriously mis-identified and obtain under 50% correct recognition rate such as P3, P5 and P8. The person P3 has lowest correct recognition rate 25% which has 50% misidentified to P8 and 25% to P1. Figure 5.4 and Figure 5.5 show the sample visual words (codebooks).

5.5 Handwritten Digit Recognition

USPS is a handwritten digits dataset [59, 159] which consists of 7291 training images and 2007 test images of size 16×16 with digits 0 to 9. Figure 5.6 shows some samples from the dataset. We represent each sample by a 256-dimensional vector. The dictionary sizes under different sizes of training samples are selected from 32, 64, 128, 256, 512, 1024. For the penalty constraint, we adopt self-ratio in Eq. (5.9), and the parameters are set empirically via cross-validation. To evaluate the performance, we train a linear SVM classifier and compare to original representation (OR), sparse coding (SC) and graph regularized sparse coding (GraohSC) methods. Table 5.6 shows the results under different sizes of training samples from 100 to

Figure 5.6: Sample digits from 0 to 9 of the USPS dataset.

Table 5.7: Recognition accuracy (%) of different methods on the Scene 15 dataset.

Method \ Size of training set	100
KSPM [4]	81.40
KC [33]	76.67
ScSPM [89]	80.28
HardSPM	75.72
SoftSPM	79.48
SISSPM	81.37
MLcSPM	82.54

7291. As shown, while the proposed MLC method is the worst when the size of training set is very small, MLC outperforms the other three methods in most of the cases and obtains the best accuracy result 95.3% under 5000 training samples.

5.6 Image Classification

In this section, we perform experiments on image classification and evaluate our method on several datasets: Caltech-101, Caltech-256 and Scene 15. We conduct experiments towards the following objectives: (1) Evaluating the performance of the proposed MLC and comparing to the state-of-the-art methods; (2) Demonstrating the advantage of the proposed approach in terms of speed-up, quantization error and accuracy; (3) Illustrating the parameter settings in dictionary learning.

5.6.1 Experimental Settings

In this experiment, we first convert RGB images to grayscale images and fix the maximum image side length to 300 pixels. We then adopt the widely used SIFT local feature [53], as in [89–91], which shows good performance on image classification [35, 178, 179]. Each SIFT feature is represented by a 128-dimensional vector. Fei-Fei *et al.* [30] have demonstrated that SIFT feature extracted from a dense regular grid outperform interest point so here the SIFT features are extracted from 16×16 pixel patches densely with the step size of 8 pixels. Noted that using a smaller step size can obtain better performance but increase computational costs due to the larger amount of patches. We randomly select 200000 SIFT samples as the training set to learn the dictionary for each dataset. The size of visual words is set as 1024. It is worth mentioning that the dictionary is initialized as the results of online dictionary learning [146] to expedite the convergence. The sparsity parameter of online dictionary learning is set as 0.1 and γ is set empirically via cross-validation. After obtaining SIFT features, we then quantize each SIFT feature into sparse representation via dictionary. To maintain the spatial order of local descriptors, we represent each image by 3-layer spatial pyramid matching (SPM) [4] which improves the descriptive power of the image representation. The 3-layer SPM partitions each image into 3 scales of segments. In our case, we adopt $2^0 \times 2^0$, $2^1 \times 2^1$ and $2^2 \times 2^2$ segments with 21 segments in total. We first obtain sparse coefficients of each SIFT feature, then use the maximum pooling function within each segment to generate a feature vector and finally concatenate all these feature vectors to form an image representation. To fair comparison with existing work, we use the linear SVM classifier with the one-vs-all strategy to train the classifier [89]. The parameters are chosen empirically via cross-validation. All the results are mean recognition rates

with standard deviation based on 5 independent experiments with different random selected training and test samples from each data set. Due to the experimental settings, we quote some results from the literature [89] and have implemented some related methods using SPM by ourselves including hard-thresholding, soft-thresholding, sure independent screening denoted by HardSPM, SoftSPM, SISSPM, respectively. We denote the proposed MLC using SPM by MLcSPM.

5.6.2 Scene 15 Dataset

Scene 15 [4, 30, 180] contains 4485 images with 15 categories from the COREL collection, personal photographs and Google image search. Each category consists of ranging from 200 to 400 images with average size of 300×250 pixels. This dataset is very diverse from indoor such as living room, kitchen, office, store, etc., to outdoor including street, inside city, building, mountain, forest, etc. Following the same setting as the previous work [4, 89], we randomly select 100 images per category as training samples and the rest as test samples. Table 5.7 shows the results. As shown, our MLcSPM method outperforms other methods, especially the lasso-based ScSPM scheme. Further, the proposed MLcSPM outperforms other marginal regression based methods.

5.6.3 Caltech 101 Dataset

Caltech 101 [181] contains 101 object categories and a background category downloading examples from Google Images and consisting of 9146 images in total ranging from 31 to 800 images per category with roughly image size of 300×200 pixels. Following the general experimental setup for caltech 101, we randomly

Table 5.8: Recognition accuracy (%) of different methods on the Caltech 101 dataset.

Method	Size of training set	
	15	30
Zhang <i>et al.</i> [182]	59.10	66.20
KSPM [4]	56.4	64.40
NBNN [35]	65.00	70.40
ML+CORR [183]	61.00	69.60
KC [33]	-	64.14
ScSPM [89]	67.00	73.20
ScSPM-LARS	65.07	71.99
ScSPM-FS	64.68	72.19
HardSPM	56.87	63.17
SoftSPM	62.31	68.38
SISSPM	64.39	71.09
MLcSPM	65.48	72.98

select 15 and 30 images per category as training samples and the rest as test samples, respectively. Table 5.8 lists the results. It is worth mentioning that we have implemented two sparsity estimation methods using least angle regression and feature sign search on the framework of sparse coding with SPM denoted by ScSPM-LARS and ScSPM-FS, respectively. As shown, our MLcSPM method outperforms other methods and is comparable to ScSPM, ScSPM-LAR and ScSPM-FS. Further, the proposed MLcSPM outperforms other marginal regression based methods such as HardSPM, SoftSPM and SISSPM. Note that the existing work removes the background category and our implemented methods using the whole dataset including clutter background category.

Table 5.9: Recognition accuracy (%) of different methods on the Caltech 256 dataset.

Method \ Size of training set	15	30	45	60
KSPM [4]	-	34.10	-	-
KC [33]	-	27.17	-	-
ScSPM [89]	27.73	34.02	37.46	40.14
HardSPM	22.75	27.29	30.27	31.97
SoftSPM	25.68	30.76	33.80	35.98
SISSPM	27.43	32.53	35.48	37.66
MLcSPM	28.86	34.45	36.84	38.50

5.6.4 Caltech 256 Dataset

Caltech 256 [184] contains 256 object categories and a background category downloading examples from Google Images and consisting of 30607 images in total ranging from 80 to 827 images per category. Following the general experimental setup for Caltech 256, we randomly select 15, 30, 45 and 60 images per category as training samples and the rest as test samples, respectively. It is worth mentioning that the existing work removes the background category and our implemented methods using the whole dataset including clutter background category. Table 5.9 lists the results. As shown, our MLcSPM method outperforms KSPM and KC and is comparable to ScSPM. Further, the proposed MLcSPM outperforms other marginal regression based methods such as HardSPM, SoftSPM and SISSPM.

5.6.5 Speed Comparison of Sparsity Estimation Methods

In this experiment, we compare the speed of different sparsity estimation methods on the Caltech 101 dataset including the lasso-based sparse coding using least angle regression, feature sign search and approximated feature sign search denoted by SC-LAR, SC-FS and SC-AFS, accordingly, and the regression based methods using hard-thresholding, soft-thresholding, sure independent screening denoted by MR-Hard, MR-Soft and MR-SIS, respectively. We denote our method by ML-R and ML-V. ML-R refers to MLC with a self-ratio energy constraint, and ML-V refers to MLC using a specific value as the constraint. Figure 5.7 shows the speed under different dictionary sizes. As shown, the lasso-based methods, SC-LAR, SC-FS and SC-AFS, have much higher computational cost in terms of processing time (seconds), especially when we use a large dictionary containing more visual words. On the other hand, MR-Hard and MR-Soft obtain the lowest computational cost under different dictionary sizes. MR-SIS, ML-R and ML-V have low computational cost under small dictionary sizes which are comparable to MR-Hard and MR-Soft and the cost rise slightly when MR-SIS, ML-R and ML-V use a larger dictionary.

5.6.6 Quantization Error of Sparsity Estimation Methods

In this experiment, we compare the quantization error of different sparsity estimation methods on the Caltech 101 dataset. Figure 5.8 shows the quantization error under different dictionary sizes. As shown, the hard-thresholding method, MR-Hard, achieves the largest quantization error under all cases. Table 5.10 lists the detailed numerical results. As shown, the lasso-based methods, SC-LAR, SC-FS and SC-AFS, obtain the smallest quantization errors. While the soft-thresholding

Figure 5.7: Speed comparison between different sparsity estimation methods on Caltech 101 data set (in Seconds).

method, MR-Soft, obtains a small quantization error under size dictionary sizes, it gets a larger error when the size of dictionary increases from 256 and onwards. On the other hand, the quantization errors of MR-SIS, ML-R and ML-V are reasonable small among the marginal regression based methods.

5.6.7 Parameter Selection

To ease the parameter selection in the experiment, we first determine two parameters, the energy bound E or the energy self-ratio e and the sparsity control parameter used in the initial dictionary learning via experimentally testing the relationship between the classification accuracy and the weights on the Caltech 101 dataset. The test values for the energy self-ratio are 5%, 10%, 15%, 20%, 40%, 60%, 80%, 100%, and those for the sparsity parameter are 0.1%, 0.2%, 0.3%, 0.4%. Figure 5.9 shows the effect of the bound of energy self-ratio e and the weight of sparsity control in terms of classification accuracy. As shown, $e = 5\%$ achieves the best result among the ratios

Figure 5.8: Quantization error comparison between different sparsity estimation methods on Caltech 101 data set.

and obtains similar results under four different sparsity values. In addition, as the self-ratio goes up, the classification accuracy declines, and $DisSpar = 0.1$ obtains lightly better performance than others under different values of energy self-ratio, especially when the ratio is larger than $e = 40\%$. Therefore, we fix $e = 5\%$ and the sparsity parameter equals to 0.1. Then, we test the energy bound E with some specific values 0.5, 1, 1.5, 2, 2.5, 3, 3.5, 4. Figure 5.10 shows the results with a fixed value of sparsity parameter. As shown, this value assignment is dataset dependent.

5.7 Action Recognition

In this section, we analyze the characteristic of the proposed method using the KTH action recognition dataset with three different features and evaluate our method with other existing methods. Experimental setting, recognition accuracy, sparsity analysis, speed comparison and quantization error are also analyzed in this section.

Table 5.10: Quantization error of different sparsity estimation methods

Method \ Size of dictionary	64	128	256	512	1024	2048	4096
SC-FS	0.08	0.07	0.06	0.05	0.04	0.04	0.03
SC-AFS	-	-	-	0.05	0.04	0.04	0.03
SC-LAR	0.08	0.06	0.05	0.05	0.04	0.04	0.03
MR-Hard	2.26	7.70	30.02	105.91	382.47	1581.63	6386.63
MR-Soft	0.17	0.14	0.33	1.30	5.30	24.80	105.91
MR-SIS	0.39	0.39	0.39	0.39	0.39	0.39	0.39
ML-R	0.38	0.35	0.30	0.21	0.12	0.18	1.18
ML-V	0.22	0.22	0.23	0.23	0.24	0.24	0.25

5.7.1 Experimental Settings

The KTH dataset [185] contains 25 persons. Each person performed 6 different activities, including boxing, handclapping, handwaving, jogging, running, and walking, respectively. Figure 5.11 shows some sample frames. There are 4 scenarios under each activity, including outdoors, outdoors with scale variation, outdoors with different clothes and indoors, respectively, resulting 599 video sequences in total. We follow the original experimental setup which divides clips into 2391 sub-clips with 863 clips for the test set (9 subjects: 2, 3, 5, 6, 7, 8, 9, 10 and 22) and 1528 clips for the training set (16 subjects: 1, 4, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 23, 24 and 25). Given a video clip, salient local regions are extracted using a 3D-Harris detector [52]. This detector responds to 3D corners and hence describes motion areas. Then, each of the detected regions is described with local descriptors. In our experiments, we compute HOG, HOF, and HOGHOF local feature descriptors [8]. HOG is a

Figure 5.9: Performance comparison between different values of sparsity and energy self-ratio on Caltech 101 data set.

histogram of gradient orientations which describes the appearance over space and time, HOF is a histogram of optical flow which describes the dynamic motion patterns and HOGHOF is a histogram concatenating HOG and HOF histograms together which describes both static and dynamic information over space and time. Next, we learn a dictionary using those local descriptors obtained from the training set via Algorithm 9. Then, we quantize all the local descriptors into sparse codes via Algorithm 1 using the learned dictionary. After that, we represent each video clip using a maximum pooling method. Let $S_i = [s_1, s_2, \dots, s_{n_i}] \in R^{p \times n_i}$ be a set of sparse codes with n_i image frames from the video clip i . The video representation of the clip i denoted by $Q_i \in R^p$ is formulated as follows:

$$Q_i(j) = \max(|s_1(j)|, |s_2(j)|, \dots, |s_{n_i}(j)|) \quad (5.29)$$

where $j = 1, \dots, p$, and p is the number of visual words in the dictionary. For each visual word, the maximum pooling function characterizes the largest weighting parameter in terms of absolute value among those sparse codes in the image set of a

Figure 5.10: Performance comparison between different values of energy bound on Caltech 101 data set.

video clip to describe the dynamic information of activity. For recognition, we use a non-linear support vector machine [186] with a X^2 kernel [8] as follows:

$$K(Q_i, Q_j) = \exp\left(-\frac{1}{2A} \sum_{m=1}^p \frac{(Q_i^{(m)} - Q_j^{(m)})^2}{Q_i^{(m)} + Q_j^{(m)}}\right) \quad (5.30)$$

where A is a mean distance value between all training samples. The parameters are selected using cross validation. In order to expedite the dictionary learning process, we initialize the dictionary using online dictionary learning method [146] with 100 runs as the warm start.

5.7.2 Recognition Accuracy

We compare our method to other existing approaches using the lasso and the marginal regression framework with dictionary size 4000, respectively. The sparsity estimation methods are as follows:

Figure 5.11: Sample frames of the KTH data set. From left to right are four scenarios: outdoors, outdoors with scale variation, outdoors with different clothes and indoors. From top to down are three example activities: boxing, handclapping and handwaving.

- *k*-means: the distant-based feature quantization method that assigns features to the nearest visual words using Euclidean distance, and present each video sequence using the resulting histograms of visual word occurrences;
- LASSO-FS: the popular and efficient lasso solution that solves lasso problem using feature-sign search;
- MR-Hard: the sparsity estimation method that uses marginal regression with simple hard thresholding;
- MR-Soft: the sparsity estimation method that uses marginal regression with simple soft thresholding;
- MR-SIS: the sparsity estimation method that uses marginal regression with hard thresholding under a specific l_1 penalty level;

- MLC: the sparsity estimation method that exploits the advantages of the marginal regression and the lasso simultaneously to provide more efficient and effective solutions.

Table 5.11 shows the results for different descriptors and quantization methods, respectively. As shown, our feature quantization scheme obtains the best results under each condition which outperforms other marginal regression methods such as MR-Hard, MR-Soft and MR-SIS by a large margin and is comparable to the lasso-based method. It is worth mentioning that the computational complexity of the proposed method can be two orders faster than that of the lasso. On the other hand, the proposed method is more robust than the conventional marginal regression based methods. k -means obtains the worst for HOG and MR-Hard obtains the worst for the other two feature descriptors, especially for the HOF feature. In addition, the l_1 -norm energy-based methods such as MLC and MR-SIS perform better than those without the energy constraint such as MR-Hard and MR-Soft. Last but not least, the soft-thresholding-type methods have better performance against the hard-thresholding-type methods under the same conditions (MLC against MR-SIS and MR-Soft against MR-Hard). We will have more discussion and analysis in the following sections.

5.7.3 Sparsity Analysis

As shown in Table 5.11, MLC and MR-Soft outperform MR-SIS and MR-hard, respectively. In other words, the soft-thresholding-type methods have better performance against the hard-thresholding-type methods. In this section, we have more discuss on sparsity and adopt MR-Hard and MR-Soft as examples.

Table 5.11: Recognition accuracy (%) of different methods using HOG, HOF, HOGHOF feature descriptors, respectively, on the KTH data set.

Feature-Method	KTH
HOG- k -means [12]	80.90
HOG-LASSO-FS [83]	83.89
HOG-MR-Hard	82.27
HOG-MR-Soft	82.39
HOG-MR-SIS [87]	84.01
HOG-MLC	85.98
HOF- k -means [12]	92.10
HOF-LASSO-FS [83]	92.70
HOF-MR-Hard	61.76
HOF-MR-Soft	85.52
HOF-MR-SIS [87]	89.11
HOF-MLC	92.47
HOGHOF- k -means [12]	91.80
HOGHOF-LASSO-FS [83]	92.00
HOGHOF-MR-Hard	87.83
HOGHOF-MR-Soft	90.73
HOGHOF-MR-SIS [87]	90.03
HOGHOF-MLC	92.35

We first describe the simple hard-thresholding scheme, MR-Hard, as follows. Given a set of marginal regression coefficients calculated from the HOF local feature descriptors $[\hat{a}_1, \dots, \hat{a}_n] \in R^{p \times n}$ where p is the number of visual words in the dictionary and n is the number of descriptors, we obtain sparse codes via the thresholding parameter t . The following is the hard-thresholding scheme:

$$\hat{s}_i^{t(m)} = \begin{cases} \hat{a}_i^{(m)} & \text{if } |\hat{a}_i^{(m)}| > t \\ 0 & \text{otherwise} \end{cases} \quad (5.31)$$

where m is the m th element of the vector. We compute the mean amount of non-zero elements of sparse codes $[\hat{s}_1^t, \dots, \hat{s}_n^t]$ under a certain threshold value t . The mathematics expression can be denoted by $\frac{\sum_{i=1}^n \|\hat{s}_i^t\|_0}{n}$ where $\|\hat{s}_i^t\|_0$ refers to l_0 norm counting non-zero elements of $\hat{s}_i^t$. The mean number of non-zero elements as t varies is shown in Figure 5.12. Note that the dictionary size is 4000 in this case. As shown, t controls the sparsity of the amount of the non-zero elements, and thus with a larger value of t , there exist a fewer non-zero elements of sparse codes.

Next, we quantize these HOF local feature descriptors using the soft-thresholding scheme, MR-soft, defined as follows:

$$\hat{s}_i^{t(m)} = \begin{cases} \hat{a}_i^{(m)} - t & \text{if } \hat{a}_i^{(m)} > t \\ \hat{a}_i^{(m)} + t & \text{if } \hat{a}_i^{(m)} < -t \\ 0 & \text{otherwise} \end{cases} \quad (5.32)$$

According to Eq. (5.31) and Eq. (5.32), we compute the average l_1 norm energy of sparse codes $[\hat{s}_1^t, \dots, \hat{s}_n^t]$ obtained from the hard-thresholding method and the soft-thresholding method under a certain threshold value t , respectively. The l_1 norm calculates the sum of the absolute values of each sparse code. The mathematics expression can be denoted by $\frac{\sum_{i=1}^n \|\hat{s}_i^t\|_1}{n}$ where $\|\hat{s}_i^t\|_1 = \sum_{m=1}^p |\hat{s}_i^t(m)|$. Figure 5.13 (a) shows the average l_1 -norm energy as t varies. As shown, using the soft-thresholding

Figure 5.12: The average amount of non-zero elements of sparse codes quantized from the HOF feature descriptors using marginal regression with simple hard thresholding approach from the KTH action recognition training set as t varies. The dictionary size is set as 4000 in this example.

scheme is a more effective way for variable selection and coefficient shrinkage. Furthermore, the soft-thresholding scheme achieves a smaller standard deviation between sparse codes as t varies as shown in Figure 5.13 (b). From the recognition viewpoint, the soft-thresholding scheme outperforms the hard-thresholding scheme as well in terms of recognition accuracy under different values of the parameter t . Figure 5.14 shows the results in terms of recognition accuracy using the soft and hard thresholding schemes under three feature descriptors, HOG, HOF, and HOGHOF as t varies, respectively. As shown, the accuracy of the soft-thresholding scheme is much better than that of the hard-thresholding scheme. Therefore, we can conclude that the soft-thresholding technique such as MR-Soft is much stable and robust than the hard-thresholding technique such as MR-Hard.

We then discuss the sparsity constraint in Eq. (5.9) which controls the sparsity according to the self-ratio on the l_1 -norm of the coefficient vector. The self-ratio

(a)

(b)

Figure 5.13: Comparison of the soft and hard-thresholding schemes. (a) The average l_1 norm energy of sparse codes under different t . As shown, using the soft-thresholding scheme is a more effective way for variable selection and coefficient shrinkage. (b) The average standard deviation of the l_1 energy between sparse codes as t changes. Sparse codes obtained via the soft-thresholding scheme are much stable than those obtained via the hard-thresholding scheme under different t .

Figure 5.14: The results in terms of recognition accuracy using the soft and hard thresholding schemes under three feature descriptors, HOG, HOF, and HOGHOF as t varies, respectively. As shown, the accuracy of the soft-thresholding scheme is much better than that of the MR-Hard method.

Figure 5.15: Recognition accuracy under different self-ratios. We obtain around 90% recognition accuracy from $e = 5\%$ to $e = 35\%$, and obtain poor recognition accuracy (below 10%) from $e = 75\%$ and upwards.

constraint in Eq. (5.9) aims to keep a certain ratio denoted by e of the marginal regression coefficients in terms of l_1 -norm energy. For example, the self-ratio $e = 5\%$ means that, for each coefficient vector, only 5% of the marginal regression coefficients in terms of the l_1 -norm energy can be kept and the rest are set to zero. Thus, we can obtain sparse codes more easily by assigning a desired ratio. Figure 5.15 shows the recognition accuracy under different self-ratios. As shown, we obtain around 90% recognition accuracy from $e = 5\%$ to $e = 35\%$, and obtain poor recognition accuracy (below 10%) from $e = 75\%$ and upwards. Therefore, we can conclude that sparse representation helps recognition because it keeps characteristics information of a signal.

5.7.4 Speed Comparison

Genovese *et al.* [88] have shown that the computational complexity of marginal regression based methods are two orders faster than that of the lasso-based methods, even though there are some efficient algorithms using gradient descent for lasso. Therefore, in this section, we examine the feature quantization speed using different sparsity estimation methods under three different feature descriptors, HOG, HOF and HOGHOF, respectively. The dictionary sizes are set from 1000 to 4000. We compare the proposed approach which is implemented by Matlab with the following:

- LASSO-FS: the Matlab implementation of the popular and efficient lasso solution developed by its authors that solves lasso problem using feature-sign search;
- LASSO-LAR: the popular and efficient lasso solution which was implemented by C++ developed by Mairal *et al.* and Jenatton *et al.* [146, 187–189] that solves lasso problem using least angle regression;

- MR-Hard: the Matlab implementation of the sparsity estimation method that uses marginal regression with simple hard thresholding;
- MR-Soft: the Matlab implementation of the sparsity estimation method that uses marginal regression with simple soft thresholding;
- MR-SIS: the Matlab implementation of the sparsity estimation method that uses marginal regression with hard thresholding under a specific l_1 -norm penalty level;

Note that all the methods except for LASSO-LAR were implemented in Matlab, and thus it would be unfair to compare them with the LASSO-LAR method together since C++ program of LASSO-LAR has a built-in speed advantage. Nevertheless, as shown in Table 5.12, the marginal regression related methods still have advantage of lower computational complexity that all of them achieve less quantization time than those of the lasso solutions. In addition, MR-SIS and MLC have comparable quantization speed but perform slower than MR-Hard and MR-soft since the former two consider the l_1 -norm energy constraint, and the latter two do not.

5.7.5 Feature Quantization Error

Besides the speed comparison, Table 5.13 also shows the quantization errors using different sparsity estimation methods under three different feature descriptors, HOG, HOF and HOGHOF, respectively. The dictionary sizes are set from 1000 to 4000. The methods we compare with are the same as those of the previous section. As shown, the LASSO-based methods achieve the lowest quantization error. For the marginal regression related methods, the proposed MLC achieves

the lowest quantization error. More specifically, the methods with an l_1 -norm penalty constraint such as MR-SIS and MLC achieve a lower quantization error than those without the penalty constraint. In addition, the soft-thresholding-like methods achieve a lower quantization error than the hard-thresholding-like methods pairwise, i.e. MR-Soft against MR-Hard, and MLC against MR-SIS.

5.7.6 Discussion

For recognition tasks, minimizing feature quantization error for sparse representation plays an important role, but how to increase discriminative power between classes efficiently is also important. While the lasso-based methods achieve the lowest quantization error, they take more time for sparsity estimation. Instead, the marginal regression methods without penalty constraint (MR-Hard and MR-Soft) can have the fastest speed for sparsity estimation; however, they also have the highest quantization errors. While MR-SIS can compete with the proposed MLC approach in terms of quantization error, MLC obtains a better recognition accuracy than that of MR-SIS since MLC is a soft-thresholding method which is more stable and robust. While there are many works focusing on sparse coding and dictionary learning, none of them has exploited the advantages of the marginal regression and the lasso simultaneously to provide more efficient and effective solutions. Therefore, in our work, we provide such an approach with a well grounded theoretical framework. The computational complexity of the proposed method can be two orders faster than that of the lasso with the expensive of acceptable quantization errors for recognition. On the other hand, the proposed method is more robust than the conventional marginal regression based methods.

Table 5.12: Computational complexity of different methods on the KTH dataset. The results are shown in terms of speed. Notice that here we randomly sampled 1,258 features from the training set.

	time (sec)			
Dictionary size Method	1000	2000	3000	4000
HOG-LASSO-FS [83]	8.08	9.71	18.83	19.77
HOG-LASSO-LAR [79]	1.69	3.61	5.81	7.67
HOG-MR-Hard	0.03	0.05	0.08	0.11
HOG-MR-Soft	0.03	0.06	0.09	0.12
HOG-MR-SIS [87]	0.54	1.01	1.48	2.03
HOG-MLC	0.47	0.92	1.44	1.93
HOF-LASSO-FS [83]	5.47	6.95	13.89	14.96
HOF-LASSO-LAR [79]	1.15	2.64	4.61	6.07
HOF-MR-Hard	0.03	0.06	0.09	0.12
HOF-MR-Soft	0.03	0.07	0.10	0.14
HOF-MR-SIS [87]	0.65	1.19	1.80	2.34
HOF-MLC	0.64	1.21	1.86	2.48
HOGHOF-LASSO-FS [83]	7.73	9.34	17.83	18.46
HOGHOF-LASSO-LAR [79]	1.60	3.52	5.79	7.78
HOGHOF-MR-Hard	0.03	0.06	0.09	0.12
HOGHOF-MR-Soft	0.03	0.06	0.10	0.14
HOGHOF-MR-SIS [87]	0.52	0.97	1.45	1.93
HOGHOF-MLC	0.44	0.88	1.34	1.79

Table 5.13: Quantization error of different methods on the KTH dataset. The results are shown in terms of quantization error. Notice that here we randomly sampled 1,258 features from the training set.

	Reconstruction error			
Dictionary size Method	1000	2000	3000	4000
HOG-LASSO-FS [83]	0.04	0.03	0.03	0.03
HOG-LASSO-LAR [79]	0.04	0.03	0.03	0.03
HOG-MR-Hard	781.94	3438.42	8079.61	14863.70
HOG-MR-Soft	68.40	319.07	764.57	1423.95
HOG-MR-SIS [87]	2.92	2.90	12.13	4.68
HOG-MLC	2.20	2.43	2.47	2.57
HOF-LASSO-FS [83]	0.03	0.03	0.03	0.02
HOF-LASSO-LAR [79]	0.03	0.03	0.03	0.02
HOF-MR-Hard	7317.76	32218.46	77688.51	142454.52
HOF-MR-Soft	1258.00	5818.47	14413.56	26806.27
HOF-MR-SIS [87]	17.02	29.80	39.24	65.60
HOF-MLC	3.35	3.27	3.12	3.09
HOGHOF-LASSO-FS [83]	0.07	0.06	0.05	0.05
HOGHOF-LASSO-LAR [79]	0.07	0.06	0.05	0.05
HOGHOF-MR-Hard	595.99	2366.80	5247.93	9215.52
HOGHOF-MR-Soft	47.14	192.52	431.67	765.61
HOGHOF-MR-SIS [87]	2.90	2.95	3.03	2.91
HOGHOF-MLC	2.18	2.33	2.43	2.47

5.8 Summary

We have proposed in this chapter a novel feature quantization approach for sparsity estimation by incorporating marginal regression coefficients with an l_1 penalty. The proposed approach has been evaluated on several visual applications including handwritten digit recognition, image classification, scene classification, action recognition and activity-based human identification. Experimental results have shown that our method can achieve better performance in terms of speed, quantization error and recognition accuracy. Furthermore, the proposed method appears to be the first algorithm that formally addresses the adaptive and automatic regularization parameter selection.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

In this thesis, we have focused our work on feature quantization including the theory and applications of sparse representation. To address the associated problems, we have presented several quantization models, sparsity estimation methods and dictionary learning algorithms on many pattern recognition and computer vision tasks such as activity-based human identification, image classification, action recognition. The main contributions of this thesis are to investigate feature quantization problems of sparse visual signal representation, analyze existing frameworks, propose new solutions and then apply them to a number of pattern recognition and computer vision tasks. Briefly speaking, we have proposed in this thesis three frameworks, graph-based sparse coding and embedding (GSCE), sparse manifold coding (SMC), and marginal-lasso coding (MLC), and, further, provided the solutions to these frameworks, experimentally evaluated their performance and compared with some of the state-of-the-art methods.

We first introduced in this thesis the background of sparse visual signal repre-

sensation in the literature, drawn an overview of the existing feature quantization models and solutions, and underlined the challenges in Chapter 1. We further reviewed relative sparse representation frameworks, sparsity estimation methods and dictionary learning algorithms in Chapter 2. Then we explored each subproblem and provided solutions to them from Chapter 3 to Chapter 5. It is worth mentioning that, from the viewpoint of applications, we investigated the problem of activity-based human identification in each chapter which aims to recognize people from their activities and investigated several visual tasks in Chapter 5. On the other hand, from the viewpoint models for sparse representation, we improved the existing sparse coding methods in a more efficient and effective way and evaluated the performance on several visual tasks including image classification, action recognition and activity-based human identification. We summarize in the following sections our work and main contributions.

6.1.1 A Lasso-based Method for Feature Coding — the High-dimensional Case

As reviewed in Chapter 2, the general sparse coding techniques fall into the category of the lasso-based models using the feature-sign search algorithm for sparsity estimation and the iteratively updating approach for dictionary learning. While the existing methods have been applied to the application of image classification and have obtained reasonable results, they ignore the fact that original feature data may lie in a high-dimensional data space which may rise the difficulty of learning a large over-complete dictionary due to the high computational complexity. The application of activity-based human identification is such a case with high-dimensional feature data. Therefore, in Chapter 3, we studied and discussed the high-dimensional case

of sparse representation for the activity-based human identification task. There are two main questions we would like to answer:

- (1) How to encode different dynamic patterns performed by the same person for recognition?
- (2) How to deal with high-dimensional feature data for feature coding?

To this end, we proposed a graph-based sparse coding and embedding (GSCE) framework to encode dynamic patterns of human movements. In this work, we learned a compact feature subspace to preserve the geometrical structure of original data and then learned a low-dimensional over-complete dictionary to encode features into sparse representation maintaining their locality relationship. By doing so, the process of feature quantization can be speeded up, and similar features can be quantized into similar visual words to improve the recognition performance. Then we used the feature-sign search method for sparsity estimation and adopted conjugate gradient decent method to solve the convex optimization problem for dictionary learning. The experimental results on the publicly data sets have clearly demonstrated the efficacy of our proposed method to recognize human at a distance.

The main contributions of this work are as follows:

- (1) We considered the high-dimensional characteristics of the task of activity-based human identification and provided a corresponding feature encoding model which obtained good results and outperformed the state-of-the-art methods.

- (2) We provided a subspace learning and graph-based embedding into the general sparse coding framework and provided an optimized solution for subspace projection, sparsity estimation and dictionary learning.

6.1.2 A Marginal Regression-based Method for Feature Coding — an Unsupervised Solution

Following the effort in Chapter 3, we found that the existing activity-based human identification methods usually assume that there only exists a global geometrical structure available in the feature set and ignore the temporal information from video clips. More specifically, some work quantizes features independently by finding their nearest clusters [149, 150] or minimizing the quantization error [151], and some work quantize them dependently with the consideration of the neighborhood structure [177], but none of them take advantage of the temporal relationship. In general, human movements always consist of a series of image frames and features extracted from these successive frames share the same person information. Hence, the temporal information can help for human identification and the unsupervised model can become a weakly supervised model.

Motivated by the fact above, in Chapter 4, we answered the questions below:

- (1) How to utilize the temporal information for feature coding to improve the recognition of a person who performed several different activities?
- (2) The general sparse coding methods are time-consuming for sparsity estimation and dictionary learning. How to increase feature coding efficiency?

To address these questions, we proposed in Chapter 4 a sparse manifold coding (SMC) framework to characterize dynamic patterns of a person. We utilized the

intrinsic structure of a video to form a video-based manifold and therefore SMC exploits the geometrical structure within and between manifolds with unequal weights. Different from Chapter 3, we provided a sparsity estimation approach using marginal regression which is two orders faster than that of the lasso-based methods and a dictionary learning algorithm using the gradient descent to solve the SMC problem. We further proposed an efficient sparsity approximation method for new coming data. Experimental results have been presented to demonstrate the efficacy of the proposed approach.

The main contributions of this work are as follows:

- (1) We considered the temporal information of a person performing an activity for the task of activity-based human identification and provided a corresponding feature encoding model which obtained good results and outperformed the state-of-the-art methods.
- (2) We provided a sparse manifold coding into a general sparse coding framework and considered a marginal regression-based solution which has shown its efficiency [88] for sparsity estimation. Further, due to the utilization of the temporal information, the unsupervised model becomes to a weakly supervised model which improves the recognition accuracy with the implied label information.

6.1.3 A Marginal Regression-based Method for Feature Coding — a Supervised Solution

When we considered an unsupervised sparse coding manifold structure as shown in the previous section, one question came into to our mind: How to construct a

supervised manifold structure if we have not only temporal information from video clips but the class information? Specifically, we answered the questions below:

- (1) The same activity performed by different persons may have a smaller intra-class variance than different activities performed by the same person. Therefore, how to build up a hierarchical manifold structure and how to incorporate temporal and class information to improve recognition performance are the main problems for the supervised case.
- (2) What are the differences between unsupervised and supervised models and solutions? Is the supervised solution better than the unsupervised one?

To address these questions, we investigated the supervised version of sparse manifold coding, constructed hierarchical manifolds including video-based and person-based manifolds structure and provided an optimized solution. While the performance of the supervised method is slightly better than the unsupervised one, the supervised version is more robust under different parameter selection.

The main contributions of this work are as follows:

- (1) This is the first work considering the supervised feature encoding for the task of activity-based human identification. We utilized the temporal information from motion patterns and simultaneously explored the class-label information for feature coding and provided a corresponding coding model and solution which have obtained good results and outperformed the state-of-the-art methods.
- (2) We provided a supervised sparse manifold coding into the general sparse coding framework with marginal regression-based solution.

6.1.4 A Marginal-lasso-based Method for Feature Coding — an Efficient and Robust Solution

While we have shown the efficiency of marginal regression for sparsity estimation, it obtains large quantization errors. On the other hand, the lasso using the feature-sign algorithm is a popular but time-consuming way for sparsity estimation and dictionary learning. Therefore, in Chapter 5, we answered the questions below:

- (1) Can we have an efficient and robust feature coding approach for sparse estimation and dictionary learning? Can we utilize the advantage between the lasso and the marginal regression methods with small quantization error but more efficiency? Further, the regularization parameter should be automatically and adaptively selected for each input signal so that the quantization error can be minimized. Can we do that?
- (2) How is the performance when we apply the proposed method into several visual tasks?

To this end, in Chapter 5, we proposed a marginal-lasso coding (MLC) framework for feature quantization including sparsity estimation, automatic and adaptive regularization parameter selection and dictionary learning which has exploited the advantages of the marginal regression and the lasso simultaneously to provide a more efficient and effective solution. Our model simultaneously represents each sparse code with similar sparsity level and shrinks individual coefficients to alleviate the bias of sparse codes caused by the highly correlated visual words. Experimental results and detailed model analysis have been presented to demonstrate the efficacy of our proposed methods.

The main contributions of this work are as follows:

- (1) We provided a generic feature coding framework for sparse representation with a more efficient and effective way. The computational complexity of the proposed method can be two orders faster than that of the lasso with the expense of acceptable quantization errors for recognition. On the other hand, the proposed method is more robust than the conventional marginal regression based methods.
- (2) We have successfully applied the proposed method to various classification and recognition tasks such as image classification, scene classification, handwritten digits clustering, action recognition and activity-based human identification and achieved good results.

6.2 Future Work

In this section, we present some research directions which can be pursued in the future.

6.2.1 Activity-based Human Identification

From the viewpoint of applications, activity-based human identification is a new research direction and we have formulated the problems and provided some solutions. While encouraging results have been achieved, there are several interesting issues which can be addressed in the future:

- (1) Datasets

Currently, only the MOBISERV-AIIA dataset [150] is constructed for this purpose. It contains 12 persons performing eating and drinking activities under two dressing scenarios in four different days. These video clips are acquired in an constrained environment with a small sample size. It is desired to build a larger dataset to have a larger diversity in terms of actions, persons and with the presence of large variations in viewpoint, scale, illumination condition, camera motion, etc. Furthermore, more realistic samples such as that from Olympic games and football games or samples in surveillance environments. By doing so, it can encourage further research into action recognition and simultaneously human identification by learning and modeling new features and representation methods.

(2) Feature Extraction

For feature extraction, we follow the existing work [149–151, 177] using appearance-based binary body masks which can work well in the condition of a clear background, but may fail to work when background is cluttered. Therefore, it is promising to explore more effective and efficient ways so that we can extract meaningful features under more complexity environment which may be more practical for real-world applications.

(3) Potential Problems

We assume in this thesis that each video clip contains one person performing an activity. This assumption, however, can be relaxed to be more realistic. It would be also interesting if we can recognize the identity of a person and his/her activity at the same time. How about two persons performing the same activity or different activities within a video clip? Can we detect the bounding box of each person and tell what they are doing? It would be more

challenging when we have two more objects available in a video clip performing at least one activity. On the other hand, can we recognize the identity of a person and his/her behaviors when he/she successively performs two different activities? Furthermore, under a surveillance environment, can we indicate a unknown person who does not include in the permitted list or detect a known person performing forbidden behaviors or activities?

6.2.2 Sparse Visual Signal Representation

From the modeling viewpoint, in this thesis, we focus on sparse coding related techniques and algorithms for feature quantization. Some potential optimization problems have also been identified to be addressed in the future:

(1) Nonlinear Sparse Coding

The general sparse coding assumes that features lies in an Euclidean space and therefore it aims to minimize the quantization error based on l_2 -norm distance, which calculates the Euclidean distance between original samples and their corresponding quantization ones representing via the linear combination of a small subset of visual words. However, sometimes this may not be true, since in real-world applications, features often belong to some manifolds with an intrinsic metric structure. Hence, one potential research is nonlinear sparse coding on Riemannian manifold [190–192]. It is worth mentioning that the computational complexity should be an important issue in this problem.

(2) Sparsity Estimation

We have shown in this thesis that sparsity estimation using marginal regression obtains a larger quantization error comparing to that using the lasso-based

solutions such as least angle regression or feature-sign search as described in chapter 5. While we have shown that the proposed MLC can achieve a much smaller quantization error among those marginal regression based methods, it can be further improved for its effectiveness and efficiency.

(3) Supervised Dictionary Learning

Instead of using predefined dictionaries such as a set of training samples [58, 193], in this thesis, we have learned an over-complete dictionary with a number of basic patterns greater than the feature dimension of data in an unsupervised manner which is a process of visual word generation. Recently, Coates *et al.* [194] have indicated that when using the conventional lasso based sparse coding as an encoder which is solved by the coordinate descent algorithm [82], any unsupervised training algorithm can be adopted to create a dictionary with similar performance. However, recent work [106, 109, 195–197] has presented some different directions. They have shown that using a supervised dictionary learning algorithm in the training stage can improve the performance for classification and recognition. More work is required to ascertain the best approaches.

Publications

Journal Papers and Manuscripts

- (1) **Tzu-Yi Hung**, Jiwen Lu, and Yap-Peng Tan, “Graph-Based Sparse Coding and Embedding for High Dimensional Feature Coding,” to be submitted.
- (2) **Tzu-Yi Hung**, Jiwen Lu, and Yap-Peng Tan, “Sparse Manifold Coding,” to be submitted.
- (3) **Tzu-Yi Hung**, Jiwen Lu, and Yap-Peng Tan, “An Efficient Sparse Representation Algorithm via Marginal-Lasso Coding,” to be submitted.
- (4) **Tzu-Yi Hung**, Zhenzhong Chen, and Yap-Peng Tan, “Deadline-aware packet scheduling and playout adaptation for scalable video over wireless network,” *Journal of Visual Communication and Image Representation*, vol. 22, no. 6, pp. 491-503, Aug., 2011.

Conference Papers

- (1) **Tzu-Yi Hung**, Jiwen Lu, and Yap-Peng Tan, “Efficient Sparsity Estimation via Marginal-Lasso Coding,” *Proceedings of European Conference on Computer Vision (ECCV)*, 2014.

- (2) **Tzu-Yi Hung**, Jiwen Lu, and Yap-Peng Tan, “Graph-Based Sparse Coding and Embedding for Activity-based Human Identification,” *Proceedings of IEEE International Conference on Multimedia and Expo (ICME)*, 2013. (oral presentation, best paper candidate)
- (3) **Tzu-Yi Hung**, Jiwen Lu, and Yap-Peng Tan, “Activity-Based Human Identification,” *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2013.
- (4) **Tzu-Yi Hung**, Jiwen Lu, and Yap-Peng Tan, “Cross-Scene Abnormal Event Detection,” *Proceedings of IEEE International Symposium on Circuit and Systems (ISCAS)*, 2013.
- (5) **Tzu-Yi Hung**, Ce Zhu, Gao Yang, and Yap-Peng Tan, “Video organization: near-duplicate video clustering,” *Proceedings of IEEE International Symposium on Circuit and Systems (ISCAS)*, 2013.
- (6) **Tzu-Yi Hung**, Zhenzhong Chen, and Yap-Peng Tan, “Playout adaptation based packet scheduling for scalable video delivery over wireless links,” *Proceedings of IEEE International Conference on Multimedia and Expo (ICME)*, 2010.

References

- [1] “Youtube.” [Online]. Available: <http://www.youtube.com/yt/press/statistics.html>
- [2] “Instagram statistics.” [Online]. Available: <http://instagram.com/press/>
- [3] K. Grauman and T. Darrell, “The pyramid match kernel: discriminative classification with sets of image features,” *Proceedings of IEEE International Conference on Computer Vision*, pp. 1458–1465, 2005.
- [4] S. Lazebnik, C. Schmid, and J. Ponce, “Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 2169–2178, 2006.
- [5] J. Willamowski, D. Arregui, G. Csurka, C. R. Dance, and L. Fan, “Categorizing nine visual classes using local appearance descriptors,” *Proceedings of IEEE International Conference on Pattern Recognition Workshop on Learning for Adaptable Visual Systems*, pp. 1–8, 2004.
- [6] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid, “Local features and kernels for classification of texture and object categories: A comprehensive study,” *International Journal of Computer Vision*, vol. 73, no. 2, pp. 213–238, 2007.

REFERENCES

- [7] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, “Object retrieval with large vocabularies and fast spatial matching,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 1–8, 2007.
- [8] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, “Learning realistic human actions from movies,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 1–8, 2008.
- [9] C. Wang, S. Yan, L. Zhang, and H.-J. Zhang, “Multi-label sparse coding for automatic image annotation,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 1643–1650, 2009.
- [10] M. Marszalek, I. Laptev, and C. Schmid, “Actions in context,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 2929–2936, 2009.
- [11] K. K. Reddy and M. Shah, “Recognizing 50 human action categories of web videos,” *Machine Vision and Applications*, vol. 24, no. 5, pp. 971–981, 2013.
- [12] H. Wang, M. M. Ullah, A. Kläser, I. Laptev, and C. Schmid, “Evaluation of local spatio-temporal features for action recognition,” *British Machine Vision Conference*, pp. 1–11, 2009.
- [13] D. Larlus, J. Verbeek, and F. Jurie, “Category level object segmentation by combining bag-of-words models with dirichlet processes and random fields,” *International Journal of Computer Vision*, vol. 88, no. 2, pp. 238–253, 2010.
- [14] P. Tirilly, V. Claveau, and P. Gros, “Language modeling for bag-of-visual words image categorization,” *Proceedings of ACM International Conference on Content-based Image and Video Retrieval*, pp. 249–258, 2008.

REFERENCES

- [15] J. Yang, Y.-G. Jiang, A. G. Hauptmann, and C.-W. Ngo, “Evaluating bag-of-visual-words representations in scene classification,” *Proceedings of ACM International Workshop on Multimedia Information Retrieval*, pp. 197–206, 2007.
- [16] M. Rusinol and J. Lladós, “Logo spotting by a bag-of-words approach for document categorization,” *Proceedings of IEEE International Conference on Document Analysis and Recognition*, pp. 111–115, 2009.
- [17] Y. Cao, C. Wang, Z. Li, L. Zhang, and L. Zhang, “Spatial-bag-of-features,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 3352–3359, 2010.
- [18] H. Jégou, M. Douze, and C. Schmid, “Improving bag-of-features for large scale image search,” *International Journal of Computer Vision*, vol. 87, no. 3, pp. 316–336, 2010.
- [19] L. Wu, S. Hoi, and N. Yu, “Semantics-preserving bag-of-words models and applications,” *IEEE Transactions on Image Processing*, vol. 19, no. 7, pp. 1908–1920, 2010.
- [20] H. M. Wallach, “Topic modeling: Beyond bag-of-words,” *Proceedings of ACM International Conference on Machine Learning*, pp. 977–984, 2006.
- [21] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray, “Visual categorization with bags of keypoints,” *Proceedings of European Conference on Computer Vision Workshop on Statistical Learning in Computer Vision*, pp. 1–22, 2004.

REFERENCES

- [22] E. Nowak, F. Jurie, and B. Triggs, “Sampling strategies for bag-of-features image classification,” *Proceedings of European Conference on Computer Vision*, pp. 490–503, 2006.
- [23] M. Marszałek and C. Schmid, “Spatial weighting for bag-of-features,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 2118–2125, 2006.
- [24] Y. Mu, J. Sun, T. X. Han, L. F. Cheong, and S. Yan, “Randomized locality sensitive vocabularies for bag-of-features model,” *Proceedings of European Conference on Computer Vision*, pp. 748–761, 2010.
- [25] Y.-G. Jiang, C.-W. Ngo, and J. Yang, “Towards optimal bag-of-features for object categorization and semantic video retrieval,” *Proceedings of ACM International Conference on Image and Video Retrieval*, pp. 494–501, 2007.
- [26] D. Batra, R. Sukthankar, and T. Chen, “Learning class-specific affinities for image labelling,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 1–8, 2008.
- [27] A. Agarwal and B. Triggs, “Multilevel image coding with hyperfeatures,” *International Journal of Computer Vision*, vol. 78, no. 1, pp. 15–27, 2008.
- [28] P. Quelhas, F. Monay, J.-M. Odobez, D. Gatica-Perez, T. Tuytelaars, and L. Van Gool, “Modeling scenes with local descriptors and latent aspects,” *Proceedings of IEEE International Conference on Computer Vision*, vol. 1, pp. 883–890, 2005.
- [29] A. Bosch, A. Zisserman, and X. Muñoz, “Scene classification using a hybrid generative/discriminative approach,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 4, pp. 712–727, 2008.

REFERENCES

- [30] F.-F. Li and P. Perona, “A bayesian hierarchical model for learning natural scene categories,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 524–531, 2005.
- [31] J. Winn, A. Criminisi, and T. Minka, “Object categorization by learned universal visual dictionary,” *Proceedings of IEEE International Conference on Computer Vision*, vol. 2, pp. 1800–1807, 2005.
- [32] F. Perronnin, C. Dance, G. Csurka, and M. Bressan, “Adapted vocabularies for generic visual categorization,” *Proceedings of European Conference on Computer Vision*, pp. 464–475, 2006.
- [33] J. C. Gemert, J.-M. Geusebroek, C. J. Veenman, and A. W. Smeulders, “Kernel codebooks for scene categorization,” *Proceedings of European Conference on Computer Vision*, pp. 696–709, 2008.
- [34] J. C. van Gemert, C. J. Veenman, A. W. M. Smeulders, and J. M. Geusebroek, “Visual word ambiguity,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 7, pp. 1271–1283, 2010.
- [35] O. Boiman, E. Shechtman, and M. Irani, “In defense of nearest-neighbor based image classification,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 1–8, 2008.
- [36] A. Bosch, A. Zisserman, and X. Muoz, “Image classification using random forests and ferns,” *Proceedings of IEEE International Conference on Computer Vision*, pp. 1–8, 2007.
- [37] O. Chum and A. Zisserman, “An exemplar model for learning object classes,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 1–8, 2007.

REFERENCES

- [38] F. Jurie and B. Triggs, “Creating efficient codebooks for visual recognition,” *Proceedings of IEEE International Conference on Computer Vision*, pp. 604–610, 2005.
- [39] J. Liu and M. Shah, “Scene modeling using co-clustering,” *Proceedings of IEEE International Conference on Computer Vision*, pp. 1–7, 2007.
- [40] F. Moosmann, E. Nowak, and F. Jurie, “Randomized clustering forests for image classification,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 9, pp. 1632–1646, 2008.
- [41] F. Perronnin, “Universal and adapted vocabularies for generic visual categorization,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 7, pp. 1243–1256, 2008.
- [42] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, “Lost in quantization: Improving particular object retrieval in large scale image databases,” in *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.
- [43] P. Quelhas, F. Monay, J. Odobez, D. Gatica-Perez, and T. Tuytelaars, “A thousand words in a scene,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 9, pp. 1575–1589, 2007.
- [44] K. E. A. Van de Sande, T. Gevers, and C. G. M. Snoek, “Evaluation of color descriptors for object and scene recognition,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 1–8, 2008.

REFERENCES

- [45] J. Sivic and A. Zisserman, “Video google: a text retrieval approach to object matching in videos,” *Proceedings of IEEE International Conference on Computer Vision*, pp. 1470–1477, 2003.
- [46] T. Tuytelaars and C. Schmid, “Vector quantizing feature space with a regular lattice,” *Proceedings of IEEE International Conference on Computer Vision*, pp. 1–8, 2007.
- [47] A. Gandhi, K. Alahari, and C. Jawahar, “Decomposing bag of words histograms,” *Proceedings of IEEE International Conference on Computer Vision*, pp. 305–312, 2013.
- [48] J. Stöttinger, J. Uijlings, A. Pandey, N. Sebe, and F. Giunchiglia, “(unseen) event recognition via semantic compositionality,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 3061–3068, 2012.
- [49] J. Uijlings, K. van de Sande, T. Gevers, and A. Smeulders, “Selective search for object recognition,” *International Journal of Computer Vision*, no. 2, pp. 151–171, 2013.
- [50] H. Harzallah, F. Jurie, and C. Schmid, “Combining efficient object localization and image classification,” in *Proceedings of IEEE International Conference on Computer Vision*, 2009, pp. 237–244.
- [51] K. Mikolajczyk and C. Schmid, “Scale and affine invariant interest point detectors,” *International Journal of Computer Vision*, vol. 60, no. 1, pp. 63–86, 2004.
- [52] I. Laptev, “On space-time interest points,” *International Journal of Computer Vision*, vol. 64, no. 2-3, pp. 107–123, 2005.

REFERENCES

- [53] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.
- [54] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 886–893, 2005.
- [55] J. A. Hartigan and M. A. Wong, “A k-means clustering algorithm,” *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, vol. 28, no. 1, pp. 100–108, 1979.
- [56] J. C. Chen, “A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 3, no. 3, pp. 32–57, 1973.
- [57] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*. Norwell, MA, USA: Kluwer Academic Publishers, 1981.
- [58] J. Wright, A. Yang, A. Ganesh, S. Sastry, and Y. Ma, “Robust face recognition via sparse representation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 210–227, 2009.
- [59] M. Zheng, J. Bu, C. Chen, C. Wang, L. Zhang, G. Qiu, and D. Cai, “Graph regularized sparse coding for image representation,” *IEEE Transactions on Image Processing*, vol. 20, no. 5, pp. 1327–1336, 2011.
- [60] E. Elhamifar and R. Vidal, “Robust classification using structured sparse representation,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 1873–1879, 2011.

REFERENCES

- [61] Y. Cong, J. Yuan, and J. Liu, “Sparse reconstruction cost for abnormal event detection,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 3449–3456, 2011.
- [62] M. Long, G. Ding, J. W. 0001, J. Sun, Y. Guo, and P. S. Yu, “Transfer sparse coding for robust image representation.” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 407–414, 2013.
- [63] H. Wang, F. Nie, H. Huang, and C. H. Q. Ding, “Heterogeneous visual features fusion via sparse multimodal machine,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 3097–3102, 2013.
- [64] L. Zhuang, A. Y. Yang, Z. Zhou, S. S. Sastry, and Y. Ma, “Single-sample face recognition with image corruption and misalignment via sparse illumination transfer.” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 3546–3553, 2013.
- [65] Y.-T. Chi, M. Ali, A. Rajwade, and J. Ho, “Block and group regularized sparse modeling for dictionary learning.” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 377–382, 2013.
- [66] L. Zhuang, H. Gao, Z. Lin, Y. Ma, X. Zhang, and N. Yu, “Non-negative low rank and sparse graph for semi-supervised learning.” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 2328–2335, 2012.
- [67] L. Cao, R. Ji, Y. Gao, Y. Yang, and Q. Tian, “Weakly supervised sparse coding with geometric consistency pooling.” *Proceedings of IEEE International*

REFERENCES

- Conference on Computer Vision and Pattern Recognition*, pp. 3578–3585, 2012.
- [68] X. Jia, H. Lu, and M.-H. Yang, “Visual tracking via adaptive structural local sparse appearance model.” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 1822 – 1829, 2012.
- [69] D. Cai, H. Bao, and X. He, “Sparse concept coding for visual analysis.” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 2905–2910, 2011.
- [70] K. Yu, Y. Lin, and J. D. Lafferty, “Learning image representations from the pixel level via hierarchical sparse coding.” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 1713–1720, 2011.
- [71] B. Zhao, F.-F. Li, and E. P. Xing, “Online detection of unusual events in videos via dynamic sparse coding.” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 3313–3320, 2011.
- [72] Z. Hong, X. Mei, D. Prokhorov, and D. Tao, “Tracking via robust multi-task multi-view joint sparse representation,” *Proceedings of IEEE International Conference on Computer Vision*, pp. 649–656, 2013.
- [73] Y.-T. Chi, M. Ali, M. Rushdi, and J. Ho, “Affine-constrained group sparse coding and its application to image-based classifications,” *Proceedings of IEEE International Conference on Computer Vision*, pp. 681–688, 2013.
- [74] T. Zhang, B. Ghanem, S. Liu, C. Xu, and N. Ahuja, “Low-rank sparse coding for image classification,” *Proceedings of IEEE International Conference on Computer Vision*, pp. 281–288, 2013.

REFERENCES

- [75] J. Thiagarajan, K. Ramamurthy, P. Turaga, and A. Spanias, *Image Understanding Using Sparse Representations*, ser. Synthesis Lectures on Image, Video, and Multimedia Processing. Morgan & Claypool, 2014.
- [76] B. Natarajan, “Sparse approximate solutions to linear systems,” *SIAM Journal on Scientific Computing*, vol. 24, no. 2, pp. 227–234, 1995.
- [77] R. Tibshirani, “Regression shrinkage and selection via the lasso,” *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996.
- [78] S. Chen, D. Donoho, and M. Saunders, “Atomic decomposition by basis pursuit,” *SIAM Journal on Scientific Computing*, vol. 20, no. 1, pp. 33–61, 1998.
- [79] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, “Least angle regression,” *Annals of Statistics*, pp. 407–499, 2004.
- [80] M. Osborne, B. Presnell, and B. Turlach, “A new approach to variable selection in least squares problems,” *IMA journal of numerical analysis*, vol. 20, no. 3, pp. 389–404, 2000.
- [81] J. Friedman, T. Hastie, H. Hofling, and R. Tibshirani, “Pathwise coordinate optimization,” *Annals of Applied Statistics*, pp. 302–332, 2007.
- [82] T. Wu and K. Lange, “Coordinate descent algorithms for lasso penalized regression,” *Annals of Applied Statistics*, pp. 224–244, 2008.
- [83] H. Lee, A. Battle, R. Raina, and A. Y. Ng, “Efficient sparse coding algorithms,” *Proceedings of Advances in Neural Information Processing Systems 19*, pp. 801–808, 2007.

REFERENCES

- [84] J. Fan and R. Li, “Variable selection via nonconcave penalized likelihood and its oracle properties,” *Journal of the American Statistical Association*, pp. 1348–1360, 2001.
- [85] H. Zou, “The adaptive lasso and its oracle properties,” *Journal of the American Statistical Association*, vol. 101, pp. 1418–1429, 2006.
- [86] N. Meinshausen, “Relaxed lasso,” *Computational Statistics and Data Analysis*, vol. 52, no. 1, pp. 374–393, 2007.
- [87] J. Fan and J. Lv, “Sure independence screening for ultrahigh dimensional feature space,” *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 70, no. 5, pp. 849–911, 2008.
- [88] C. R. Genovese, J. Jin, L. Wasserman, and Z. Yao, “A comparison of the lasso and marginal regression,” *Journal of Machine Learning Research*, vol. 13, no. 1, pp. 2107–2143, 2012.
- [89] J. Yang, K. Yu, Y. Gong, and T. Huang, “Linear spatial pyramid matching using sparse coding for image classification,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 1794–1801, 2009.
- [90] S. Gao, I. Tsang, and L. Chia, “Laplacian sparse coding, hypergraph laplacian sparse coding, and applications,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 92–104, 2012.
- [91] S. Gao, I. W.-H. Tsang, L.-T. Chia, and P. Zhao, “Local features are not lonely - laplacian sparse coding for image classification,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 3555–3561, 2010.

REFERENCES

- [92] J. Yang, J. Wright, T. S. Huang, and Y. Ma, “Image super-resolution via sparse representation,” *IEEE Transaction on Image Processing*, vol. 19, no. 11, pp. 2861–2873, 2010.
- [93] J. Han, S. He, X. Qian, D. Wang, L. Guo, and T. Liu, “An object-oriented visual saliency detection framework based on sparse coding representations,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 12, pp. 2009–2021, 2013.
- [94] X. Gao, N. Wang, D. Tao, and X. Li, “Face sketch photo synthesis and retrieval using sparse representation,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 8, pp. 1213–1226, 2012.
- [95] H. Xiong, Z. Pan, X. Ye, and C. W. Chen, “Sparse spatio-temporal representation with adaptive regularized dictionary learning for low bit-rate video coding,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 4, pp. 710–728, 2013.
- [96] J. Yang, J. Wright, T. Huang, and Y. Ma, “Image super-resolution via sparse representation,” *IEEE Transactions on Image Processing*, vol. 19, no. 11, pp. 2861–2873, 2010.
- [97] R. Ptucha and A. Savakis, “Lge-ksvd: Robust sparse representation classification,” *IEEE Transactions on Image Processing*, vol. 23, no. 4, pp. 1737–1750, 2014.
- [98] S. Gao, I. Tsang, and L.-T. Chia, “Sparse representation with kernels,” *IEEE Transactions on Image Processing*, vol. 22, no. 2, pp. 423–434, 2013.

REFERENCES

- [99] C.-K. Chiang, C.-H. Liu, C.-H. Duan, and S.-H. Lai, “Learning component-level sparse representation for image and video categorization,” *IEEE Transactions on Image Processing*, vol. 22, no. 12, pp. 4775–4787, 2013.
- [100] Y. Han, F. Wu, Y. Zhuang, and X. He, “Multi-label transfer learning with sparse representation,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 20, no. 8, pp. 1110–1121, 2010.
- [101] T. Ogawa and M. Haseyama, “Missing image data reconstruction based on adaptive inverse projection via sparse representation,” *IEEE Transactions on Multimedia*, vol. 13, no. 5, pp. 974–992, 2011.
- [102] L. Shen, C. Yeo, and B.-S. Hua, “Intrinsic image decomposition using a sparse representation of reflectance,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 12, pp. 2904–2915, 2013.
- [103] W. Dong, L. Zhang, R. Lukac, and G. Shi, “Sparse representation based image interpolation with nonlocal autoregressive modeling,” *IEEE Transactions on Image Processing*, vol. 22, no. 4, pp. 1382–1394, 2013.
- [104] S. Shekhar, V. Patel, N. Nasrabadi, and R. Chellappa, “Joint sparse representation for robust multimodal biometrics recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 1, pp. 113–126, 2014.
- [105] Z. Jiang, Z. Lin, and L. Davis, “Label consistent k-svd: Learning a discriminative dictionary for recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 11, pp. 2651–2664, 2013.
- [106] H. Wang, F. Nie, W. Cai, and H. Huang, “Semi-supervised robust dictionary learning via efficient l-norms minimization,” *Proceedings of IEEE International Conference on Computer Vision*, pp. 1145–1152, 2013.

REFERENCES

- [107] C. Bao, J.-F. Cai, and H. Ji, “Fast sparsity-based orthogonal dictionary learning for image restoration,” *Proceedings of IEEE International Conference on Computer Vision*, pp. 3384–3391, 2013.
- [108] Y.-T. Chi, M. Ali, A. Rajwade, and J. Ho, “Block and group regularized sparse modeling for dictionary learning,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 377–382, 2013.
- [109] J. Mairal, F. Bach, and J. Ponce, “Task-driven dictionary learning,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 4, pp. 791–804, 2012.
- [110] C. Lu, J. Shi, and J. Jia, “Online robust dictionary learning,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 415–422, 2013.
- [111] Q. Liu, S. Wang, L. Ying, X. Peng, Y. Zhu, and D. Liang, “Adaptive dictionary learning in sparse gradient domain for image recovery,” *IEEE Transactions on Image Processing*, vol. 22, no. 12, pp. 4652–4663, 2013.
- [112] C. Yang, J. Peng, and J. Fan, “Image collection summarization via dictionary learning for sparse representation,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 1122–1129, 2012.
- [113] Y.-C. Chen, V. Patel, J. Pillai, R. Chellappa, and P. Phillips, “Dictionary learning from ambiguously labeled data,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 353–360, 2013.

- [114] M. Harandi, C. Sanderson, C. Shen, and B. Lovell, “Dictionary learning and sparse coding on grassmann manifolds: An extrinsic solution,” *Proceedings of IEEE International Conference on Computer Vision*, pp. 3120–3127, 2013.
- [115] C.-K. Chiang, T.-F. Su, C. Yen, and S.-H. Lai, “Multi-attributed dictionary learning for sparse coding,” *Proceedings of IEEE International Conference on Computer Vision*, pp. 1137–1144, 2013.
- [116] N. Zhou, Y. Shen, J. Peng, and J. Fan, “Learning inter-related visual dictionary for object recognition,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 3490–3497, 2012.
- [117] E. Elhamifar and R. Vidal, “Sparse subspace clustering,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 2790–2797, 2009.
- [118] Y.-X. Wang and H. Xu, “Noisy sparse subspace clustering,” *Proceedings of ACM International Conference on Machine Learning*, pp. 89–97, 2013.
- [119] X. Peng, L. Zhang, and Z. Yi, “Scalable sparse subspace clustering,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 430–437, 2013.
- [120] H. Ji and F. Su, “Robust motion segmentation via refined sparse subspace clustering,” *Proceedings of IEEE International Conference on Pattern Recognition*, pp. 1546–1549, 2012.
- [121] V. Patel, H. V. Nguyen, and R. Vidal, “Latent space sparse subspace clustering,” *Proceedings of IEEE International Conference on Computer Vision*, pp. 225–232, 2013.

REFERENCES

- [122] E. Elhamifar and R. Vidal, “Sparse subspace clustering: Algorithm, theory, and applications,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 11, pp. 2765–2781, 2013.
- [123] B. Wang and Z. Tu, “Sparse subspace denoising for image manifolds,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 468–475, 2013.
- [124] Y. Wang and G. Mori, “Human action recognition by semilattent topic models,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 10, pp. 1762–1774, 2009.
- [125] A. Kovashka and K. Grauman, “Learning a hierarchy of discriminative space-time neighborhood features for human action recognition,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 2046–2053, 2010.
- [126] J. Sun, X. Wu, S. Yan, L.-F. Cheong, T.-S. Chua, and J. Li, “Hierarchical spatio-temporal context modeling for action recognition,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 2004–2011, 2009.
- [127] I. Junejo, E. Dexter, I. Laptev, and P. Pérez, “View-independent action recognition from temporal self-similarities,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 1, pp. 172–185, 2011.
- [128] A. Fathi and G. Mori, “Action recognition by learning mid-level motion features,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 1–8, 2008.

REFERENCES

- [129] N. Boulgouris, D. Hatzinakos, and K. Plataniotis, “Gait recognition: a challenging signal processing technology for biometric identification,” *IEEE Signal Processing Magazine*, vol. 22, no. 6, pp. 78–90, 2005.
- [130] M. Nixon and J. Carter, “Automatic recognition by gait,” *Proceedings of the IEEE*, vol. 94, no. 11, pp. 2013–2024, 2006.
- [131] D. Tao, X. Li, X. Wu, and S. Maybank, “General tensor discriminant analysis and gabor features for gait recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 10, pp. 1700–1715, 2007.
- [132] N. Boulgouris and Z. Chi, “Gait recognition using radon transform and linear discriminant analysis,” *IEEE Transactions on Image Processing*, vol. 16, no. 3, pp. 731–740, 2007.
- [133] J. Lu and Y.-P. Tan, “View recognition of human gait sequences in videos,” *Proceedings of IEEE International Conference on Image Processing*, pp. 2457–2460, 2010.
- [134] J. Lu and E. Zhang, “Gait recognition for human identification based on ica and fuzzy svm through multiple views fusion,” *Pattern Recognition Letters*, vol. 28, no. 16, pp. 2401–2411, 2007.
- [135] C. Wang, J. Zhang, J. Pu, X. Yuan, and L. Wang, “Chrono-gait image: a novel temporal template for gait recognition,” *Proceedings of European Conference on Computer Vision*, pp. 257–170, 2010.
- [136] J. Lu and Y.-P. Tan, “Uncorrelated discriminant simplex analysis for view-invariant gait signal computing,” *Pattern Recognition Letters*, vol. 31, no. 5, pp. 382–393, 2010.

REFERENCES

- [137] J. Lu and T. Y.-P., “Gait-based human age estimation,” *IEEE Transactions on Information Forensics and Security*, vol. 5, no. 4, pp. 761–770, 2010.
- [138] N. Liu, J. Lu, and Y.-P. Tan, “Joint subspace learning for view-invariant gait recognition,” *IEEE Signal Processing Letters*, vol. 18, no. 7, pp. 431–434, 2011.
- [139] L. Wang, T. Tan, H. Ning, and W. Hu, “Silhouette analysis-based gait recognition for human identification,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 12, pp. 1505–1518, 2003.
- [140] S. Sarkar, P. J. Phillips, Z. Liu, I. R. Vega, P. Grother, and K. W. Bowyer, “The humanID gait challenge problem: data sets, performance, and analysis,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 2, pp. 162–177, 2005.
- [141] J. Lu and Y.-P. Tan, “Regularized locality preserving projections and its extensions for face recognition,” *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 40, no. 3, pp. 958–963, 2010.
- [142] M. Belkin and P. Niyogi, “Laplacian eigenmaps and spectral techniques for embedding and clustering,” *Proceedings of Advances in Neural Information Processing Systems*, pp. 585–591, 2001.
- [143] U. Luxburg, “A tutorial on spectral clustering,” *Statistics and Computing*, vol. 17, no. 4, pp. 395–416, 2007.
- [144] M. Belkin and P. Niyogi, “Laplacian eigenmaps for dimensionality reduction and data representation,” *Neural Computation*, pp. 1373–1396, 2003.
- [145] K. Balasubramanian, K. Yu, and G. Lebanon, “Smooth sparse coding via marginal regression for learning sparse representations,” *Proceedings of ACM International Conference on Machine Learning*, pp. 289–297, 2013.

REFERENCES

- [146] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, “Online learning for matrix factorization and sparse coding,” *Journal of Machine Learning Research*, vol. 11, pp. 19–60, 2010.
- [147] D. L. Donoho, “For most large underdetermined systems of linear equations the minimal ℓ_1 -norm solution is also the sparsest solution,” *Communications on Pure and Applied Mathematics*, vol. 59, pp. 797–829, 2004.
- [148] A. Coates and A. Ng, “The importance of encoding versus training with sparse coding and vector quantization,” *Proceedings of International Conference on Machine Learning*, pp. 921–928, 2011.
- [149] N. Gkalelis, A. Tefas, and I. Pitas, “Human identification from human movements,” *Proceedings of IEEE International Conference on Image Processing*, pp. 2585–2588, 2009.
- [150] A. Iosifidis, A. Tefas, and I. Pitas, “Activity based person identification using fuzzy representation and discriminant learning,” *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 2, pp. 530–542, 2012.
- [151] J. Lu, J. Hu, X. Zhou, and Y. Shang, “Activity-based human identification using sparse coding and discriminative metric learning,” *Proceedings of ACM International Conference on Multimedia*, pp. 1061–1064, 2012.
- [152] A. Iosifidis, A. Tefas, and I. Pitas, “Person identification from actions based on artificial neural networks,” *IEEE workshop on Computational Intelligence in Biometrics and Identity Management*, pp. 7–13, 2013.
- [153] H. Yan, J. Lu, and X. Zhou, “Activity-based person identification using discriminative sparse projections and orthogonal ensemble metric learning,”

REFERENCES

- Proceedings of European Conference on Computer Vision Workshop*, vol. 8926, 2014.
- [154] A. K. Jain and D. Zongker, “Representation and recognition of handwritten digits using deformable templates,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 12, pp. 1386–1391, 1997.
- [155] A. H.-R. Ko, R. Sabourin, and A. De Souza Britto, Jr., “A new hmm-based ensemble generation method for numeral recognition,” *Proceedings of the 7th International Conference on Multiple Classifier Systems*, pp. 52–61, 2007.
- [156] A. H.-R. Ko, R. Sabourin, and A. d. S. Britto, Jr., “Ensemble of hmm classifiers based on the clustering validity index for a handwritten numeral recognizer,” *International Journal of Pattern Analysis and Applications*, vol. 12, no. 1, pp. 21–35, 2009.
- [157] A. Hung-Ren Ko, P. Cavalin, R. Sabourin, and A. de Souza Britto, “Leave-one-out-training and leave-one-out-testing hidden markov models for a handwritten numeral recognizer: The implications of a single classifier and multiple classifications,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 12, pp. 2168–2178, 2009.
- [158] K. Labusch, E. Barth, and T. Martinetz, “Simple method for high-performance digit recognition based on sparse coding,” *IEEE Transactions on Neural Networks*, vol. 19, no. 11, pp. 1985–1989, 2008.
- [159] J. Hull, “A database for handwritten text recognition research,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 5, pp. 550–554, 1994.

REFERENCES

- [160] N. Cuntoor, B. Yegnanarayana, and R. Chellappa, “Activity modeling using event probability sequences,” *IEEE Transactions on Image Processing*, vol. 17, no. 4, pp. 594–607, 2008.
- [161] B. Yao, X. Jiang, A. Khosla, A. Lin, L. Guibas, and L. Fei-Fei, “Human action recognition by learning bases of action attributes and parts,” *Proceedings of IEEE International Conference on Computer Vision*, pp. 1331–1338, 2011.
- [162] S. Sadanand and J. Corso, “Action bank: A high-level representation of activity in video,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 1234–1241, 2012.
- [163] M. Jain, H. Jegou, and P. Bouthemy, “Better exploiting motion for better action recognition,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 2555–2562, 2013.
- [164] H. Wang, A. Klaser, C. Schmid, and C.-L. Liu, “Action recognition by dense trajectories,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 3169–3176, 2011.
- [165] H. Wang and C. Schmid, “Action recognition with improved trajectories,” *Proceedings of IEEE International Conference on Computer Vision*, 2013.
- [166] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu, “Dense trajectories and motion boundary descriptors for action recognition,” *International Journal of Computer Vision*, vol. 103, no. 1, pp. 60–79, 2013.
- [167] S. Ji, W. Xu, M. Yang, and K. Yu, “3d convolutional neural networks for human action recognition,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 35, no. 1, pp. 221–231, 2013.

REFERENCES

- [168] X. Wu, D. Xu, L. Duan, J. Luo, and Y. Jia, “Action recognition using multilevel features and latent structural svm,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 8, pp. 1422–1431, 2013.
- [169] D. Gong, G. Medioni, and X. Zhao, “Structured time series analysis for human action segmentation and recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 7, pp. 1414–1427, 2014.
- [170] G. Yu, J. Yuan, and Z. Liu, “Propagative hough voting for human activity detection and recognition,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, no. 1, pp. 87–98, 2015.
- [171] H. Jhuang, J. Gall, S. Zuffi, C. Schmid, and M. Black, “Towards understanding action recognition,” *Proceedings of IEEE International Conference on Computer Vision*, pp. 3192–3199, 2013.
- [172] D. Oneata, J. Verbeek, and C. Schmid, “Action and event recognition with fisher vectors on a compact feature set,” *Proceedings of IEEE International Conference on Computer Vision*, pp. 1817–1824, 2013.
- [173] G. Sharma, F. Jurie, and C. Schmid, “Expanded parts model for human attribute and action recognition in still images,” in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, 2013, pp. 652–659.
- [174] S. Wang, Z. Ma, Y. Yang, X. Li, C. Pang, and A. Hauptmann, “Semi-supervised multiple feature analysis for action recognition,” *IEEE Transactions on Multimedia*, vol. 16, no. 2, pp. 289–298, 2014.
- [175] Y.-L. Boureau, J. Ponce, and Y. LeCun, “A theoretical analysis of feature pooling in visual recognition,” *Proceedings of ACM International Conference on Machine Learning*, pp. 111–118, 2010.

REFERENCES

- [176] L. Gorelick, M. Blank, E. Shechtman, M. Irani, and R. Basri, “Action as space-time shapes,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 12, pp. 224–2253, 2007.
- [177] T.-Y. Hung, J. Lu, and Y.-P. Tan, “Graph-based sparse coding and embedding for activity-based human identification,” *Proceedings of IEEE International Conference on Multimedia and Expo*, pp. 1–6, 2013.
- [178] K. Mikolajczyk and C. Schmid, “A performance evaluation of local descriptors,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1615–1630, 2005.
- [179] C. Wang, D. Blei, and F.-F. Li, “Simultaneous image classification and annotation,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 1903–1910, 2009.
- [180] A. Oliva and A. Torralba, “Modeling the shape of the scene: A holistic representation of the spatial envelope,” *International Journal of Computer Vision*, vol. 42, no. 3, pp. 145–175, 2001.
- [181] L. Fei-Fei, R. Fergus, and P. Perona, “Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition Workshop of Generative Model Based Vision*, pp. 59–70, 2004.
- [182] H. Zhang, A. Berg, M. Maire, and J. Malik, “Svm-knn: Discriminative nearest neighbor classification for visual category recognition,” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 2126–2136, 2006.

REFERENCES

- [183] P. Jain, B. Kulis, and K. Grauman, “Fast image search for learned metrics,” in *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.
- [184] G. Griffin, A. Holub, and P. Perona, “Caltech-256 object category dataset,” *Technical Report*, pp. 1–20, 2007. [Online]. Available: <http://authors.library.caltech.edu/7694>
- [185] C. Schuldt, I. Laptev, and B. Caputo, “Recognizing human actions: a local svm approach,” *Proceedings of IEEE International Conference on Pattern Recognition*, pp. 32–36, 2004.
- [186] C.-C. Chang and C.-J. Lin, “Libsvm: A library for support vector machines,” *Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, pp. 1–27, 2011.
- [187] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, “Online dictionary learning for sparse coding,” *Proceedings of ACM International Conference on Machine Learning*, pp. 689–696, 2009.
- [188] J. Mairal, R. Jenatton, F. R. Bach, and G. R. Obozinski, “Network flow algorithms for structured sparsity,” *Proceedings of Advances in Neural Information Processing Systems*, pp. 1558–1566, 2010.
- [189] R. Jenatton, J. Mairal, G. Obozinski, and F. Bach, “Proximal methods for hierarchical sparse coding,” *Journal of Machine Learning Research*, vol. 12, pp. 2297–2334, 2011.
- [190] J. Ho, Y. Xie, and B. Vemuri, “On a nonlinear generalization of sparse coding and dictionary learning,” *Proceedings of ACM International Conference on Machine Learning*, vol. 28, no. 3, pp. 1480–1488, 2013.

REFERENCES

- [191] S. Jayasumana, R. Hartley, M. Salzmann, H. Li, and M. T. Harandi, “Kernel methods on the riemannian manifold of symmetric positive definite matrices.” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 73–80, 2013.
- [192] J. Xu, V. K. Ithapu, L. Mukherjee, J. M. Rehg, and V. Singh, “Gosus: Grassmannian online subspace updates with structured-sparsity,” *Proceedings of IEEE International Conference on Computer Vision*, pp. 3376–3383, 2013.
- [193] Y. Cong, J. Yuan, and J. Liu, “Sparse reconstruction cost for abnormal event detection.” *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 3449–3456, 2011.
- [194] A. Coates and A. Ng, “The importance of encoding versus training with sparse coding and vector quantization,” *Proceedings of ACM International Conference on Machine Learning*, pp. 921–928, 2011.
- [195] M. Yang, D. Zhang, X. Feng, and D. Zhang, “Fisher discrimination dictionary learning for sparse representation,” *Proceedings of IEEE International Conference on Computer Vision*, pp. 543–550, 2011.
- [196] Z. Jiang, Z. Lin, and L. Davis, “Label consistent k-svd: Learning a discriminative dictionary for recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 11, pp. 2651–2664, 2013.
- [197] M. Gangeh, A. Ghodsi, and M. Kamel, “Kernelized supervised dictionary learning,” *IEEE Transactions on Signal Processing*, vol. 61, no. 19, pp. 4753–4767, 2013.