Contribution to dimension reduction techniques : application to object tracking

Weizhi Lu

Thèse

THESE INSA Rennes
sous le sceau de l'Université européenne de Bretagne
pour obtenir le titre de
DOCTEUR DE L'INSA DE RENNES
Spécialité : Traitement du signal et de l'image

présentée par
Weizhi LU

ECOLE DOCTORALE : MATISSE
LABORATOIRE : IETR

Contribution to Dimension Reduction Techniques: Application to Object Tracking

Thèse soutenue le 16.07.2014
devant le jury composé de :

Christine GUILLEMOT
Directeur de Recherche, INRIA Rennes / Président
Didier COQUIN
Professeur, Universté de Savoie / Rapporteur
Denis HAMAD
Professeur, Université du Littoral Côte d'Opale / Rapporteur
Michèle GOUIFFES
Maître de conférences, Université Paris Sud 11 / Examinateur
RONSIN Joseph
Professeur, INSA de Rennes / Co-directeur
KPALMA Kidiyo
Professeur, INSA de Rennes / Directeur de thèse

Contribution aux techniques de la réduction de dimension: application au suivi d'objet

Weizhi LU

Acknowledgements

Foremost, I would like to express my appreciation to Prof. Joseph Ronsin and Prof. Kidiyo Kpalma for offering me the opportunity of studying in France, for giving me great freedom in research, and for helping me revise the papers over and over again. I would also like to thank Mingqiang Yang for his help on my application for this lab, my Master advisor Piming Ma for her sustainable encouragement and guidance, and the CSC secretary Jingmei Zhao who is generous in giving help. I would especially like to thank Prof. Rémi Gribonval who has kindly answered me a number of questions on compressed sensing, and Prof. Jacek Cichon for his guidance on the proof related to Bernoulli vector. I also want to thank my dissertation committee for their time and insights on the thesis.

My special thanks go to the friends in Rennes. Thanks to Cong Bai and Hui Ji who took me from the airport to campus on my first day in Rennes. Thanks to Xiaohui Yi for offering me his room in the first summer vacation. Thanks to Jinglin Zhang who is always ready to help others and spread happiness. Thanks to Yi Liu, Hengyang Wei, Ming Liu, Lining Peng, Hua Fu and Tian Xia for their helpful comments on my thesis presentation. There are many things that have deeply impressed me over the last few years, while I can't mention all of them here. Sincere thanks are given to all friends who have helped me and brought me joy.

I would like to thank my family for the irreplaceable roles they have played. Thanks to my parents for their love and unreserved support in all my pursuits. Thanks to my wife Weiyu for always being there, accompanying me through these hard days.

Finally, I would like to send my gratitude and blessing to my motherland, who is funding thousands of youth like me chasing their dreams overseas.

Contents

List of Figures

List of Tables

1 Introduction

With the development of data collection technology, high-dimensional data arise in many research areas and pose severe challenge to computation and storage. For instance, in the areas of signal processing and biostatistics we often cope with the data of at least million dimensions, such as image and DNA. To overcome the curse of dimension, we usually resort to the techniques of dimension reduction, which attempt to project the high-dimensional data to a relatively low-dimensional space while still preserving the information of interest [1]. This kind of techniques has received considerable attentions in research as diverse as statistics, bioinformatics, signal processing, computer vision, machine learning and so on. In this thesis, we will focus our attention on three popular dimension reduction techniques [2]: compressed sensing, random projection and sparse representation, which share the same mathematical model $\mathbf{y} = \mathbf{A}\mathbf{x}$ but operate for different purposes, where $\mathbf{x}$ denotes a high-dimensional vector, $\mathbf{y}$ denotes a relatively low-dimensional vector, and $\mathbf{A}$ is a projection matrix. The contributions of the thesis to these techniques are briefly introduced in the sequel.

Compressed sensing is a novel technique aiming to recover the sparse signal $\mathbf{x}$ from $\mathbf{y}$ with much fewer measurements than the conventional Nyquist-rate requires [3] [4], which is of a wonderful prospect, since natural signals usually expose sparse structures in the time, space or frequency domain. One major challenge of this technique is to construct the underdetermined sensing matrix $\mathbf{A}$ with good performance. It is known that some randomized matrices with elements i.i.d drawn from some well-known distributions, such as Gaussian distribution and Bernoulli distri-

bution, can provide good sensing performance with high probability. However, their hardware implementation is expensive. In practice, it is more interesting to explore the zero-one binary matrix with deterministic structure. Recently some works have been proposed to construct such kind of matrices, while the optimal binary matrix remains unknown. In the thesis, this problem will be successfully addressed [5] [6]. Furthermore, another interesting problem of $\mathbf{A}$ with high compression ratio is also investigated for compressed sensing.

Different from compressed sensing, random projection is developed to preserve the pairwise distances of high-dimensional dataset of $\mathbf{x}$ in the low-dimensional projection space of $\mathbf{y}$, such that the task of classification can be conducted [7] [8] [9]. In fact, this technique and related applications have been extensively studied in the past decade. It is known that the random projection matrix $\mathbf{A}$ with high probability satisfying the property of distance preservation, can be constructed with elements i.i.d drawn from the symmetric distribution with zero mean and unit variance. In practice, it is computationally attractive to reduce the density of $\mathbf{A}$ while without lowering the performance of classification. Unfortunately, in theory the property of distance preservation will degrade as the matrix density decreases, which seems unfavorable for classification. However, it should be noted that the task of classification prefers to maximize the inter-class distance rather than to preserve the distance. Based on this principle, the thesis presents so far the most sparse random projection matrix, which is proved holding better feature selection performance than other more dense random matrices [10].

Similarly to the recovery process of compressed sensing, sparse representation assumes that a vector $\mathbf{y}$ of interest can be approximated by a linear combination of few elements from an overcomplete dictionary $\mathbf{A}$ [11] [12], and so it shares the same sparse solution algorithms with compressed sensing [11] [12]. But unlike compressed sensing, exact sparse solution $\mathbf{x}$ generally does not exist in sparse representation, since the sparse linear relation between $\mathbf{y}$ and $\mathbf{A}$ cannot be ensured in most applications. While the studies of two dimension reduction techniques above are mainly at the theoretical level, the problems of sparse representation generally comes from specific applications. There is no uniform criterion for the dictionary construction,

which highly depends on the specific application. For instance, in image processing the dictionary $\mathbf{A}$ is often constructed with a collection of elements of interests. The sparse elements of $\mathbf{x}$ are used to measure the similarity between $\mathbf{y}$ and $\mathbf{A}$, or to identify the elements of $\mathbf{A}$ similar to $\mathbf{y}$. In this thesis, sparse representation will be studied in the context of visual object tracking, in which random projection is also adopted as an efficient feature selection tool. Current tracking work mainly focuses on the improvement of performance, while ignoring the computation load introduced by sparse representation. In this thesis we are motivated to employ sparse representation in terms of both effectiveness and efficiency [13].

The thesis is organized with three parts corresponding to the three dimension reduction techniques mentioned above. In the first part, the fundamental knowledge of compressed sensing is first reviewed in chapter 2, and then the deterministic construction of optimal binary matrix is detailed in chapter 3. Finally, the performance of random Bernoulli matrices with high compression rates is investigated in chapter 4. In the second part, we propose so far the most sparse random projection matrix as detailed in chapter 5. In the third part, we explore the application of sparse representation to visual object tracking. A simple but effective single-object tracking scheme is proposed in chapter 6, and a simple multi-object tracking scheme is presented in chapter 7. Finally, the thesis is concluded and discussed in chapter 8.

Part I

Compressed Sensing

2 Fundamentals

Compressed sensing states that a sparse signal can be acquired/recovered with an underdetermined matrix [3] [4], which is briefly described as follows. Let $\mathbf{x} \in \mathbb{R}^n$ be a vector with $k \ll n$ nonzero elements, typically called k-sparse signal or holding sparsity k, and $\mathbf{A} \in \mathbb{R}^{m \times n}$ be a sensing matrix with $m \ll n$, compressed sensing states that the high-dimensional vector $\mathbf{x}$ can be perfectly recovered with both given $\mathbf{A}$ and few linear measurements: $\mathbf{y} = \mathbf{A}\mathbf{x}$. In terms of basic linear algebra, there should be infinitely many solutions to $\mathbf{x}$. But if the additional condition of k-sparse is considered, as will be detailed in next section, there will exist some $\mathbf{A}$ with $m \geq 2k$ ensuring that the unique solution to $\mathbf{x}$ can be derived by solving the ℓ_0-norm based minimization problem below:

$$\min_{\hat{\mathbf{x}}} ||\hat{\mathbf{x}}||_0 \quad \text{s.t.} \quad \mathbf{y} = \mathbf{A}\hat{\mathbf{x}} \tag{2.1}$$

where the ℓ_0-norm of $\hat{\mathbf{x}}$ counts the number of nonzero elements in $\hat{\mathbf{x}}$ [14]. Unfortunately, the solution to the combinatorial optimization problem (2.1) is NP-hard. In practice the unique solution can also be promised by relaxing (2.1) to ℓ_1-minimization based convex optimization problem (also referred to as basis pursuit [15])

$$\min_{\hat{\mathbf{x}}} ||\hat{\mathbf{x}}||_1 \quad \text{s.t.} \quad \mathbf{y} = \mathbf{A}\hat{\mathbf{x}} \tag{2.2}$$

by imposing a relatively restrictive isometry on $\mathbf{A}$ [16]. The exact solution to formula (2.2) can be well addressed with current convex optimization algorithms [17]. This

helps the ℓ_1-minimization problem formula (2.2) attract much more attention than ℓ_0-minimization problem formula (2.1) in compressed sensing.

It is interesting to note that, the recovery process of compressed processing can also be regarded as a process of sparse representation (also called sparse coding), which has recently been intensively studied as an independent direction. Sparse representation assumes that a vector $\mathbf{y}$ of interest can be represented with few atoms of an overcomplete dictionary $\mathbf{A}$. Thus it also shares the same sparse solution algorithms with compressed sensing. Different from compressed sensing, sparse representation has no uniform criterion on the construction of the matrix $\mathbf{A}$, which usually depends on the specific application. Generally, the matrix is either previously given or has to be learned with given $\mathbf{y}$. Note that, exact sparse solution usually cannot be derived for sparse representation, since for given matrix $\mathbf{A}$, the sparse solution $\mathbf{x}$ cannot be ensured for all possible $\mathbf{y}$. It implies that the application of sparse representation usually does not require strict theoretical support. In contrast, to provide accurate sparse solution, compressed sensing presents more challenges in theory. This explains why in this thesis we will focus our attention on the theory of compressed sensing, but on the application of sparse representation.

According to the previous defintion, compressed sensing clearly presents two major tasks. One is to explore the conditions on $\mathbf{A}$ that enable the unique solution to formula (2.1) and formula (2.2), and the other is to develop efficient solution algorithms for formulas (2.1) and (2.2). In the following parts, we will first address the two problems above, and then demonstrate the application potential of compressed sensing.

2.1 Conditions on sensing matrices

In this section we introduce several typical conditions on $\mathbf{A}$ that support the exact recovery of k-sparse $\mathbf{x}$ through solving the ℓ_0-minimization problem formula (2.1) or ℓ_1-minimization problem formula (2.2). As for the coincidence between ℓ_0 and ℓ_1, the interested readers could refer to [18–23]. In the following parts, the first two terms, called *spark* and *coherence*, are related to the ℓ_0-minimization problem,

and the remaining two terms concerned with the ℓ_1-minimization problem are named *null space property* (NSP) and *restricted isometry property* (RIP).

2.1.1 Spark and Coherence

Definition 2.1.1 (Spark)**.** *The spark of a matrix* $\mathbf{A}$*, denoted as Spark(*$\mathbf{A}$*), is the smallest number of linearly* dependent *columns of* $\mathbf{A}$*.*

Theorem 2.1.1. *A* k*-sparse signal* $\mathbf{x}$ *can be uniquely recovered from* $\mathbf{y} = \mathbf{A}\mathbf{x}$ *with the* ℓ_0*-minimization problem formula* (2.1)*, if and only if the sparsity* $k < Spark(\mathbf{A})/2$*.*

Proof. We first prove that $k < Spark(\mathbf{A})/2$ is necessary. Suppose $k \geq Spark(\mathbf{A})/2$, then there exists a nonzero vector $\mathbf{v}$ with $||\mathbf{v}||_0 = Spark(\mathbf{A}) \leq 2k$, such that $\mathbf{A}\mathbf{v} = 0$. Clearly $\mathbf{v}$ allows to be expressed as $\mathbf{v} = \mathbf{x} - \mathbf{z}$ with $||\mathbf{x}||_0 = k$ and $||\mathbf{z}||_0 = Spark(\mathbf{A}) - k \leq k$. This means $\mathbf{A}\mathbf{x} = \mathbf{A}\mathbf{z}$, and the k-sparse $\mathbf{x}$ is not the unique ℓ_0-minimization solution of $\mathbf{y} = \mathbf{A}\mathbf{x}$ due to $||z||_0 \leq k$.

Now consider the ℓ_0-minimization based recovery is unique under $k < Spark(\mathbf{A})/2$. For the sake of contradiction, we first suppose that $\mathbf{y} = \mathbf{A}\mathbf{x}$ with $||\mathbf{x}||_0 = k$, and a vector $\mathbf{z} \neq \mathbf{x}$ with $||\mathbf{z}||_0 \leq k$ exists such that $\mathbf{y} = \mathbf{A}\mathbf{z}$. Then it can be derived that

$$\mathbf{A}(\mathbf{x} - \mathbf{z}) = \mathbf{0} \quad \text{with} \quad ||\mathbf{x} - \mathbf{z}||_0 < Spark(\mathbf{A})$$

which contradicts the definition of $Spark(\mathbf{A})$. □

Note that $Spark(\mathbf{A}) \leq m + 1$ holds for arbitrary underdetermined matrix $\mathbf{A}$. It easily follows that $k \leq m/2$ is necessary for exact recovery. It is clearly difficult to characterize the spark of a given matrix, say nothing of building a matrix of relatively large spark. Thus in practice we prefer another more practical parameter, termed *coherence* defined as below [24–27].

Definition 2.1.2 (Coherence)**.** *The coherence of a matrix* $\mathbf{A}$ *is the largest absolute correlation between arbitrary two distinct columns of* $\mathbf{A}$*:*

$$\mu(\mathbf{A}) = \max_{1 \leq i < j \leq n} \frac{|\langle a_i, a_j \rangle|}{||a_i||_2 ||a_j||_2} \tag{2.3}$$

where a_i denotes the i-th column of $\mathbf{A}$.

Note that, for analysis convenience, in this thesis the sensing matrix is typically assumed to be of normalized columns except for special explanation. So we often simply say $\mu(\mathbf{A}) = \max_{1\leq i<j\leq n} |\langle a_i, a_j\rangle|$. The sparsity k allowed by exact recovery is upper bounded with $\mu(\mathbf{A})$ in the theorem below.

Theorem 2.1.2. *If*

$$k < \frac{1}{2}(\mu(\mathbf{A})^{-1} + 1), \tag{2.4}$$

a k-sparse signal $\mathbf{x}$ *is the unique recovery of* $\mathbf{y} = \mathbf{A}\mathbf{x}$ *with the ℓ_0-minimization problem formula* (2.1).

Proof. We first need to prove a critical property, that is $Spark(\mathbf{A}) \geq 1 + \mu(\mathbf{A})^{-1}$. Let subset $\psi \subseteq \{1, 2, ..., n\}$ with cardinality denoted as $|\psi|$, $\mathbf{A}_\psi$ be a submatrix of $\mathbf{A}$ with columns indexed by ψ, and $\mathbf{A}_\psi^T$ be the transpose of $\mathbf{A}_\psi$. Consider the Gram matrix $\mathbf{G} = \mathbf{A}_\psi^T\mathbf{A}_\psi \in \mathbb{R}^{|\psi|\times|\psi|}$. Clearly its diagonal elements $g_{ii} = 1$, and off-diagonal elements $g_{ij} \leq \mu(\mathbf{A}) \leq 1$, $i \neq j$. Recall that the columns of $\mathbf{A}_\psi$ are linearly independent if and if only Gram matrix $\mathbf{G}$ has positive determinant, equivalently each eigenvalue is positive. With Gershgorin circle theorem [28], the i-th eigenvalue of $\mathbf{G}$ is bounded in the interval $[g_{ii} - r_i,\ g_{ii} + r_i]$, where $r_i = \sum_{j=1;j\neq i}^{|\psi|} |g_{ij}|$. To render all eigenvalues positive, we only require $1-(|\psi|-1)\mu(\mathbf{A}) > 0$. In other words, for any $|\psi| < \mu(\mathbf{A})^{-1} + 1$, the columns of $\mathbf{A}_\psi$ are linearly independent. From the definition of spark, it can be easily deduced that $Spark(\mathbf{A}) \geq \mu(\mathbf{A})^{-1} + 1$. By merging the result with Theorem 2.1.1, we immediately derive the condition $k < \frac{1}{2}(\mu(\mathbf{A})^{-1} + 1)$ for the unique solution based on ℓ_0-minimization. □

2.1.2 Null Space Property (NSP)

For easier reading, we begin with some basic notions. $\mathcal{N}(\mathbf{A}) := \{x : \mathbf{A}x = 0\}$ is called the null space of $\mathbf{A}$. Let $\psi \subseteq \{1, 2, ..., n\}$ with cardinality $1 \leq |\psi| \leq n$ and $\psi^c := \{1, 2, ..., n\} \setminus \psi$, then we write $\mathbf{x}_\psi \in \mathbb{R}^n$ as a vector keeping the elements of $\mathbf{x}$ indexed by ψ while setting others to zero.

Definition 2.1.3 (NSP of order k)**.** *A matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ satisfies NSP of order k, if it holds*

$$||\mathbf{x}_\psi||_1 < ||\mathbf{x}_{\psi^c}||_1 \quad with \quad |\psi| = k \tag{2.5}$$

for all $\mathbf{x} \in \mathcal{N}(\mathbf{A}) \setminus \{0\}$.

Theorem 2.1.3. *A k-sparse signal $\mathbf{x}$ can be uniquely recovered from $\mathbf{y} = \mathbf{A}\mathbf{x}$ with the ℓ_1-minimization problem (1.2), if and only if $\mathbf{A}$ satisfies the NSP of order k.*

Proof. We first prove NSP is a sufficient condition. Assume $\mathbf{A}$ satisfies NSP of order k and k-sparse $\mathbf{x}$ has nonzero coordinates on the set ψ. If there exists a vector $\mathbf{z}$ such that $\mathbf{A}\mathbf{z} = \mathbf{A}\mathbf{x}$, then we have $\mathbf{A}\mathbf{v} = 0$ with $\mathbf{v} =: \mathbf{z} - \mathbf{x}$. With NSP, it follows that

$$\begin{aligned}
||\mathbf{z}||_1 &= ||\mathbf{v} + \mathbf{x}||_1 \\
&= ||\mathbf{v}_\psi + \mathbf{x}_\psi||_1 + ||\mathbf{v}_{\psi^c} + \mathbf{x}_{\psi^c}||_1 \\
&= ||\mathbf{v}_\psi + \mathbf{x}||_1 + ||\mathbf{v}_{\psi^c}||_1 \\
&\geq ||\mathbf{x}||_1 - ||\mathbf{v}_\psi||_1 + ||\mathbf{v}_{\psi^c}||_1 \\
&> ||\mathbf{x}||_1.
\end{aligned}$$

Now we are ready to prove NSP is necessary. In contrast to NSP, assume $|\mathbf{v}_\psi| > |\mathbf{v}_{\psi^c}|$ for $\mathbf{v} \in \mathcal{N}(\mathbf{A}) \setminus \{0\}$. Then we can write $\mathbf{y} := \mathbf{A}\mathbf{v}_\psi = \mathbf{A}(-\mathbf{v}_{\psi^c})$. This means a $|\psi|$ sparse signal $\mathbf{v}_\psi$ cannot be derived with ℓ_1-minimization since there exists another solution $-\mathbf{v}_{\psi^c}$ with smaller ℓ_1 norm. □

The idea of NSP appeared early in [25] [26], and got its name in [14]. For a given sensing matrix $\mathbf{A}$, it is preferable to seek the largest k satisfying (2.5). But clearly it is a NP-hard combinatorial problem. In practice, the term RIP detailed below is more popular, since to some extent it can be analyzed even in the presence of noise [29] [30]. So in the following chapter of matrix construction, RIP is exploited as a performance evaluation tool.

2.1.3 Restricted Isometry Property (RIP)

RIP also holds a critical parameter termed restricted isometry constant (RIC), as defined below.

Definition 2.1.4 (RIC)**.** *The RIC of a matrix* $\mathbf{A} \in \mathbb{R}^{m\times n}$ *is defined as the smallest* $\delta_k \in (0,1)$ *such that the inequality*

$$(1-\delta_k)||\mathbf{x}||^2 \leq ||\mathbf{A}\mathbf{x}||^2 \leq (1+\delta_k)||\mathbf{x}||^2 \tag{2.6}$$

holds for all k-sparse signals $\mathbf{x} \in \mathbb{R}^n$*.*

RIP states that if the δ_k of $\mathbf{A}$ is small enough, the k-sparse signals can be recovered with ℓ_1-minimization problem (2.2) [16] [31]. In this case, the matrix is often called satisfying RIP of order k. Note that $\delta_i \leq \delta_j$, if $i < j$ [32] [33]. It means that RIP of order k in fact supports all signals of sparsity not larger than k. In practice, a matrix of a relatively small δ_k is preferable since it affords a relatively larger sparsity k. Then two interesting problems arise. First, it is desirable if RIP holds a relatively large upper bound for δ_k. Recently a few bounds have been successively derived in [14, 34–37]. However, they usually behave pessimistically compared to the performance of actual matrices. Second, to seek a better sensing matrix, we have to determine the δ_k for a given matrix. Unfortunately, like NSP, it is proved NP-hard as well [38, 39]. But as detailed below, δ_k can be approximately evaluated with the extreme eigenvalues of Gram matrix $\mathbf{A}_\psi^T\mathbf{A}_\psi$, where $\mathbf{A}_\psi \in \mathbb{R}^{m\times|\psi|}$ is a submatrix of $\mathbf{A}$ with columns indexed by $\psi \subseteq \{1,2,...,n\}$, and $\mathbf{A}_\psi^T$ denotes the transpose of $\mathbf{A}_\psi$. Specifically, (2.6) can be reformulated as

$$1-\delta_k \leq \lambda_{min} \leq \frac{\|\mathbf{A}_\psi\mathbf{x}\|^2}{\|\mathbf{x}\|^2} \leq \lambda_{max} \leq 1+\delta_k \tag{2.7}$$

which holds for all $|\psi| = k$ and $\mathbf{x} \in \mathbb{R}^{|\psi|}$. λ_{min} and λ_{max} here denote the two extreme eigenvalues of $\mathbf{A}_\psi^T\mathbf{A}_\psi$. The pursuit to δ_k then turns to the solution to the extreme eigenvalues of Gram matrix. Note that in practice the two extreme eigenvalues are usually not symmetric about 1. But we can easily derive a symmetric form by scaling $\mathbf{A}$ with $\sqrt{\frac{\lambda_{max}+\lambda_{min}}{2}}$, thereby deriving $\delta_k = \frac{\lambda_{max}-\lambda_{min}}{\lambda_{max}+\lambda_{min}}$. So in practice we say a matrix

satisfying RIP usually under the case where its some scale form satisfies the RIP. In fact, $\delta_k \in (0, 1)$ can be derived so long as $0 < \lambda_{min} < \lambda_{max} < \infty$.

Next, we discuss the solution problems of extreme eigenvalues. Note that, for a given submatrix size $|\psi|$, it is difficult to exactly calculate the extreme eigenvalues of all possible Gram matrices $\mathbf{A}_\psi^T\mathbf{A}_\psi$. In practice, we intend to approximately bound the extreme eigenvalues by analyzing the possible distribution of the elements of $\mathbf{A}_\psi^T\mathbf{A}_\psi$ [40]. Currently there are two major algorithms to address this problem, namely Wigner semicircle law [41] [42] and Gershgorin circle theorem [28]. In practice, both of them present some limitations. To be specific, Wigner semicircle law is proposed to approximately estimate the eigenvalues of random symmetric matrix. Its solution is derived by assuming $\mathbf{A}_\psi^T\mathbf{A}_\psi$ with infinite size. In practice, this condition is hard to satisfy because the sparsity k of interest is usually small. As for Gershgorin circle theorem, it can be used to bound each eigenvalue of square matrix. The bounds cannot be really achieved until the following two conditions are simultaneously satisfied: 1) the nonzero entries of the eigenvector share the same magnitude; 2) the elementwise product between the eigenvector and the off-diagonal elements in the corresponding row vector, has all elements with the same sign. In practice, the two conditions above seem hard to satisfy. Furthermore, it is hard to judge how well both conditions above are obeyed, even though the distribution of $\mathbf{A}_\psi^T\mathbf{A}_\psi$ is given.

In this chapter, to better bound the extreme eigenvalues of small-sized $\mathbf{A}_\psi^T\mathbf{A}_\psi$, we will exploit a novel estimation method proposed in [43]. This method can accurately bound the extreme eigenvalues of arbitrary-sized random symmetric matrix, under the assumption that there exist some $\mathbf{A}_\psi^T\mathbf{A}_\psi$ that could achieve some specific distribution. The possibility that the required distributions can be well satisfied, can also be roughly estimated by observing the actual distribution of given $\mathbf{A}_\psi^T\mathbf{A}_\psi$.

2.2 Solution Algorithm

In real applications, the exact solution to ℓ_1-minimization problem (2.2) tends to be impractical due to two major reasons. First, rather than being strictly sparse,

real sparse signals are often compressible, namely with few significant elements and many negligible small ones. Second, we generally have to face the presence of noises. However, exact solution is usually not necessary for most applications related to sparse presentation, such as signal detection, classification, and so on. So in practice the ℓ_1-minimization problem is often considered with the form of errors

$$\min ||\hat{\mathbf{x}}||_1 \quad \text{s.t.} \quad ||\mathbf{y} - \mathbf{A}\hat{\mathbf{x}}||^2 \leq \epsilon \tag{2.8}$$

which is known as basis pursuit de-nosing (BPDN) [15]. In fact, similar optimization principle, termed Lasso [44] has been early demonstrated in statistics for variable selection, as below

$$\min_{\hat{\mathbf{x}}} ||\mathbf{y} - \mathbf{A}\hat{\mathbf{x}}||^2 \quad \text{s.t.} \quad ||\hat{\mathbf{x}}||_1 \leq \tau. \tag{2.9}$$

These two problems (2.8) and (2.9) are equivalent, and can obtain the same solution by tuning the constraint parameters. Additionally, in practice they are often studied in a regularized form as below

$$\min_{\hat{\mathbf{x}}} \frac{1}{2}||\mathbf{y} - \mathbf{A}\hat{\mathbf{x}}||^2 + \lambda||\hat{\mathbf{x}}||_1. \tag{2.10}$$

which coincides with (2.8) and (2.9) by adjusting the penalized parameters λ [45, Theorem 27.4]. There are also some works concerning the selection of λ [46–49].

It is known that the three problems mentioned above can be successfully addressed with general-purpose convex optimization algorithms [17]. However, these algorithms are computationally expensive. Thus a variety of improved algorithms are specially proposed for ℓ_1-minimization problem, such as gradient projection [50], Homotopy [51] [52], LARS [53], and others [54] [55] [56] [57]. It is worth mentioning that in practice the greedy least-square algorithms, known as orthogonal matching pursuit (OMP) algorithm, seems to be more popular due to its much lower complexity but competitive performance. Here we mainly review the OMP algorithm [58] [59] and its variants. For a general overview of solution algorithms, see [60] [61] [33, 62–65].

As sketched in Algorithm 2.1, OMP is a simple iteration algorithm, which de-

Algorithm 2.1 Orthogonal Matching Pursuit (OMP) algorithm

Input: $\mathbf{y}$ and $\mathbf{A}$ with the i-th column denoted as $\mathbf{a}_i$;
Output: $\hat{\mathbf{x}}$ with nonzero coordinates $\hat{\mathbf{x}}_{\psi_t} = \mathbf{z}$;
Initialize: $\mathbf{r}_0 = y$ and $\psi_0 = \emptyset$;

```
for t=1; t=t+1 do
  î = argmax_i{|⟨a_i, r_{t-1}⟩|}, 1 ≤ i ≤ n;
  ψ_t = ψ_{t-1} ⋃ î;
  z = (A_{ψ_t}^{-1} A_{ψ_t})^T A_{ψ_t}^{-1} y;
  r_t = y − Az;
  if r_t or t reaches some given threshold, then
    terminate the iteration;
  end if
end for
```

termines the column of $\mathbf{A}$ the most correlated with the residual $\mathbf{r}$ at each iteration. The coefficients of collected columns are derived by representing $\mathbf{y}$ with least-square error. The algorithm will terminate if the representation error or the number of nonzero coefficients achieves some given threshold. Note that the matrix inverse operator involved in least-square can be simply decomposed into multiple vector inverse operations [58]. OMP in fact only involves simple matrix-vector multiplication. Of course, the complexity advantage is obtained at the cost of performance. Specifically, OMP does not support uniform recovery, and cannot ensure recovering arbitrary distributions of sparse signals [66]. For instance, it works worse when the sparse signal has the same nonzero elements in magnitude [59].

To overcome this problem, several variants of OMP are successively proposed with provable performance guarantees. These algorithms are known as SP [67], ROMP [68], CoSamp [32], and StOMP [69]. Their major difference from OMP lies in that they are all implemented by selecting a few instead of one column of $\mathbf{A}$ at a time. So the four algorithms above are similar in essence. For better understanding, the flow of SP algorithm is illustrated in Algorithm 2.2. Note that except for ROMP, the other three algorithms all require prior information either on signal sparsity or on matrix-vector correlation. This clearly limits their practical application to some extent, and also stimulates the emergence of sparsity-adaptive algorithms [70].

Algorithm 2.2 Subspace Pursuit (SP) algorithm

Input: k, $\mathbf{y}$ and $\mathbf{A}$;
Output: $\hat{\mathbf{x}}$ with nonzero coordinates $\hat{\mathbf{x}}_{\psi_t} = (\mathbf{A}_{\psi_t}^{-1}\mathbf{A}_{\psi_t})^T\mathbf{A}_{\psi_t}^{-1}\mathbf{y}$;
Initialize: $\mathbf{r}_0 = \mathbf{y}$ and ψ_0={indices of the first k largest magnitude entries of $\mathbf{A^T r}_{t-1}$};
  for t=1; t=t+1 **do**
    $\tilde{\psi}_t$=$\psi_{t-1}\bigcup${indices of the first k largest magnitude entries of $\mathbf{A'r}_{t-1}$};
    $\mathbf{z} = (\mathbf{A}_{\tilde{\psi}_t}^{-1}\mathbf{A}_{\tilde{\psi}_t})^T\mathbf{A}_{\tilde{\psi}_t}^{-1}\mathbf{y}$;
    ψ_t={indices of the first k largest magnitude entries of $\mathbf{z}$};
    $\mathbf{z} = (\mathbf{A}_{\psi_t}^{-1}\mathbf{A}_{T_t})^T\mathbf{A}_{\psi_t}^{-1}\mathbf{y}$;
    $\mathbf{r}_t = \mathbf{y} - \mathbf{Az}$;
    if $||\mathbf{r}_t|| \geq ||\mathbf{r}_{t-1}||$ **then**
      let $\psi_t = \psi_{t-1}$ and terminate the iteration;
    end if
  end for

2.3 Applications

The ℓ_1-minimization technique underlying compressed sensing naturally serves two popular application fields: compression of sparse signals and sparse representation of interesting signals with overcomplete dictionary. We first discuss the application on signal compression. This application is built on the fact that most natural signals can be represented with a sparse or compressible form. The notion of 'compressible' generally means that the ordered elements of $|\mathbf{x}|$ decreases at an exponential rate. For instance, the image can be sparsely represented with the differences between adjacent rows or columns, and likewise, the sparse form can be obtained from the adjacent frames of videos. More generally, it has been widely shown that most signals $\mathbf{x}$ of interest are compressible over an orthogonal or overcomplete basis $\mathbf{\Phi}$ [26] [25], namely $\mathbf{x} = \mathbf{\Phi c}$ with $\mathbf{c}$ being compressible. In this case, the sensing matrix transforms to $\mathbf{A\Phi}$ from $\mathbf{A}$. Luckily, $\mathbf{A\Phi}$ still holds RIP if $\mathbf{A}$ is a random matrix while $\mathbf{\Phi}$ is an orthogonal matrix [33]. But if $\mathbf{A}$ is deterministic, $\mathbf{A\Phi}$ has to be considered specially. In this thesis, for analysis convenience, we simply assume that $\mathbf{x}$ is sparse and only consider the property of $\mathbf{A}$.

To better understand the procedure of compression, here we review the well-known example of single-pixel camera in Figure 2.1 [71–73]. Let us first introduce its critical part, the digital micromirror device (DMD) array. The array consists

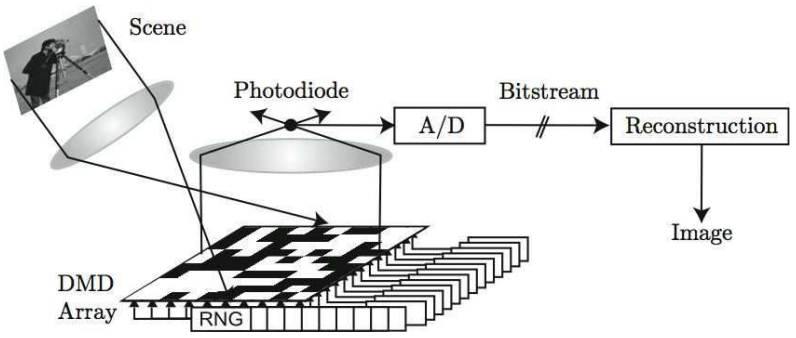


Figure 2.1: single-pixel camera [71].

of n micromirrors, each of them reflecting the light or not with the binary value from the random number generator (RNG). Then the imaging process can be easily understood as below. A n-pixel sparse image is first projected on the DMD array via a biconvex lens, and then again via a biconvex lens, some pixels randomly reflected by DMD array are focused/added on a photodiode, which outputs a voltage as a measurement of the image. In fact, DMD array here acts as a row of random binary sensing matrix. The n-pixel sparse image will be acquired after the pattern of DMD array has varied $m \ll n$ times. It is apparent that this kind of camera is low-cost since it only employs one photodiode. But it is is time-consuming since m measurements have to be collected serially. This implies that it is of practical interest to reduce the measurement number m as large as possible [74]. Compared to conventional image compression, compressed sensing now exposes more practical value in magnetic resonance imaging (MRI) [75]. For instance, it had been reported speeding up pediatric MRI with a factor of seven, while preserving diagnostic quality [76]. Another interesting problem should be noted that, for analysis convenience, most of the literature has been assuming that the sparse signal $\mathbf{x}$ is discrete-valued, though actual signals tend to be continuous in time or space. Recently this problem begins to be considered [77]. In this thesis, however, we go on conducting the research with discrete-valued $\mathbf{x}$.

3 Optimal binary sensing matrices

Compressed sensing aims to acquire sparse signals with an underdeterminded sensing matrix. In practice, the deterministic construction of sensing matrix has been a challenge. In this chapter, we propose the optimal binary matrix by searching the best Restricted Isometry Property. For a binary matrix of given size, the optimal zero-one distribution will be determined by the bipartite graph with as many edges as possible but without cycles of length 4. A greedy algorithm, termed bipartite graph based construction (BGC) algorithm, is specially developed to effectively construct such kind of graph. The practically constructed optimal binary matrix achieves the desired performance in simulation.

3.1 Introduction

Compressed sensing has recently been recognized due to its ability to recover sparse signals with an underdetermined matrix [4]. Related theories in fact can be traced back to the early study of variable selection [44] and sparse representation [25] [26]. Since then, the sparse solution algorithms applicable to compressed sensing have been extensively studied [60]. In this chapter, we will focus our attention on the deterministic construction of sensing matrices, which is still an open problem in compressed sensing.

To recover a k-sparse signal of length n, it has been proved that the optimal matrix should have the number of measurements as few as $\mathcal{O}(k \log(n/k))$ [78]. It is known that some randomized matrices and their sparse versions can provide the

optimal performance with high probability [9, 16, 31, 79]. In practice, it is more computationally attractive to deterministically construct the binary or ternary matrices even with some loss of performance [80]. Currently these matrices are mainly constructed with some known codes, such as BCH codes [81] [80], Reed-Solomon codes [82], Reed-Muller codes [83–85], LDPC codes [86] [87], and so on. These codes generally provide the codewords with relatively large mutual distances (equivalently, relatively low mutual correlations), and so they are available for the construction of sensing matrices. For compressed sensing, it is desirable to collect a set of codewords with as large mutual distances as possible. Based on this principle, DeVore [88] proposed the best known deterministic matrix with a sub-optimal performance [89]. An interesting question naturally arises: can we get the optimal binary or ternary matrix with coding theory? The answer is pessimistic. It is necessary to point out that in coding theory, to implement the parity-check decoding, the codewords are generated only on the null-space of parity-check matrix over finite field. For compressed sensing, however, it is reasonable to conjecture that a more orthogonal set of zero-one vectors will be obtained from the whole vector space. Besides performance limitation, the deterministic matrices based on coding theory are also imperfect on complexity. Specifically, it is known that the matrix with competitive performance allows to be very sparse in compressed sensing [79]. However, the desired sparsity cannot be achieved by the coding theory which generally produces the codeword with elements being 0 and 1 equiprobably. In addition, it is worth noting that the arbitrary size of matrix is hard to be obtained from the coding theory which generally cannot provide the codeword with arbitrary length.

Recently the bipartite graph has been used to seek the optimal binary matrix and achieved some interesting results [90] [91]. For instance, Gilbert and Indyk simply characterized the binary matrix of $m = \mathcal{O}(k \log(n/k))$ with an expander graph, while the explicit construction of the graph is unknown [92]. Another seemingly practical work by Dimakis and Khajehnejad, et al. [89] [93] demonstrated that the optimal performance can be achieved by the bipartite graph with girth $\Omega(\log(n))$, where the term *girth* denotes the minimum length of the shortest cycles in the bipartite graph. Their conclusion that the larger girth implies the better performance was

also supported by Liu and Xia in [94]. Unfortunately, the above result on girth is still insufficient for us to determine the optimal binary matrix, since the binary matrix cannot be accurately characterized only using the girth. In practice, most known binary matrices share the same girth, i.e. an even value equal or slightly larger than 4.

The rest of the chapter is organized as follows. In the next section, we characterize the binary matrix with bipartite graph. In section 3.3, the optimal binary matrix with best RIP is derived. In section 3.4, a novel greedy algorithm is proposed to construct the optimal binary matrix. The performance advantages of the proposed construction algorithm and the optimal binary matrix are validated in section 3.5. Finally, the chapter is concluded in section 3.6.

3.2 Binary matrix characterized with bipartite graph

We begin by introducing the basic notions of bipartite graph. As illustrated in Figure 3.1, a bipartite graph consists of two classes of nodes, which are termed as variable nodes and measurement nodes. It is associated with a binary matrix by making the two classes of nodes correspond to the columns and rows of binary matrix, respectively. The edges between both classes of nodes are determined by the nonzero positions of binary matrix. If the nodes of each class have the same number of edges, the graph and associated matrix are viewed as regular; otherwise, they are called irregular. In this chapter, we mainly study regular binary matrices. In terms of the equivalence between the binary matrix and bipartite graph, the two notions are often used interchangeably. From each variable node, a subgraph with multiple floors, as illustrated in Figure 3.1, can be generated by forward traversing all connected nodes. The subgraph often includes some closed paths, termed as cycles. The length of the cycle is measured with the number of edges, which can only take even values not less than 4. Among all the subgraphs, the length of the shortest cycles is defined as the girth of the bipartite graph. Empirically, as the edge number increases, the shorter cycles are inevitable, and the girth immediately becomes smaller. Here it is interesting to note that, in a subgraph as in Figure 3.1,

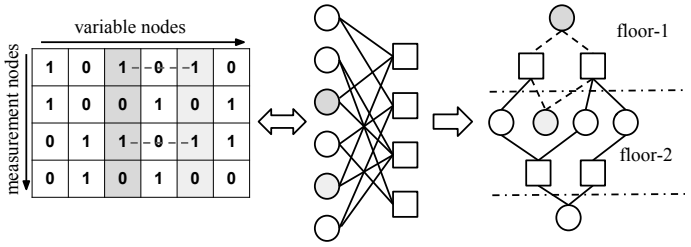


Figure 3.1: From left to right: a binary matrix, the corresponding bipartite graph and a subgraph expanded from a variable node. The variable nodes and measurement nodes are denoted with circle and square, respectively. If two variable nodes share two same nonzero positions in their corresponding columns, they will form a shortest cycle of length 4 (dashed lines), as the dashed lines shown in the subgraph.

if the root variable node is further connected to a measurement node included in the f-th floor of the graph, where $f > 1$, the generated new cycles will hold the length of $2f$. This property will be used to characterize the optimal binary matrix in the next section.

For the convenience of analysis, in this thesis the binary matrix is divided into two classes according to the distribution of girth, as detailed in the following Definitions 1 and 2:

Definition 3.2.1 ($\mathbf{A}_L$)**.** $\mathbf{A}_L \in \{0, 1/\sqrt{d}\}^{m\times n}$ *represents an ensemble of regular binary matrices with girth larger than 4. It holds* $2 \leq d \leq d_{max}$ *nonzero entries per column and* nd/m *nonzero entries per row, where* d_{max} *denotes the maximum column degree* d *allowed for* $\mathbf{A}_L$*. Any two distinct columns of* $\mathbf{A}_L$ *share at most one common nonzero position.*

Definition 3.2.2 ($\mathbf{A}_E$)**.** $\mathbf{A}_E \in \{0, 1/\sqrt{d}\}^{m\times n}$ *represents an ensemble of regular binary matrices with girth equal to 4. It holds* $3 \leq d \leq m - 2$ *nonzero entries per column and* nd/m *nonzero entries per row. The maximum number of nonzero positions overlapped between two distinct columns is* $2 \leq s \leq d - 1$*.*

To better understand the two Definitions above, it is necessary to clarify some crucial parameters at first. In Definition 1, $\mathbf{A}_L$ has an upper bound d_{max} on the column degree d, because for a matrix of given size, the shortest cycles of length 4 are inevitable as the edge number nd increases. The $\mathbf{A}_L$ with d_{max} in fact corresponds

to a regular bipartite graph with as many edges as possible, while with girth larger than 4. To avoid producing the $\mathbf{A}_E$ matrix with same columns, the ranges of d and s are restricted in Definition 2. Now it is interesting to know what kind of known matrices that the terms $\mathbf{A}_L$ and $\mathbf{A}_E$ actually correspond to. We have to say that $\mathbf{A}_E$ covers most known binary matrices, which are generally constructed without strict correlation constraint as required for $\mathbf{A}_L$. Here we take the sparse random matrix $\mathbf{R} \in \{0, 1/\sqrt{d}\}^{m\times n}$ as example, which presents comparable sensing performance with Gaussian random matrices by randomly selecting $d \ll n$ nonzero positions per column [79]. As demonstrated in Lemma 3.2.1, its each implementation is an $\mathbf{A}_E$ matrix because it will take all possible column correlations with some probability. In contrast, the matrix $\mathbf{A}_L$ is defined with column correlations being binary, namely equal to 0 or $1/d$ with the probabilities shown in Lemma 3.2.2. To the best of our knowledge, this kind of matrices is mainly explored in the study of LDPC codes, where it is used as the parity-check matrix [95]. Note that, LDPC codes are concerned only with relatively small d, i.e., usually $d = 3$ or 4. However, in compressed sensing, as will be detailed in the next section, we are more interested in the maximum column degree d, namely d_{max}.

Lemma 3.2.1 (column correlation of random binary matrices)**.** *Let r_i and r_j denote the i-th and j-th columns of random binary matrix $\mathbf{R} \in \{0, 1/\sqrt{d}\}^{m\times n}$. Then the correlation between columns satisfies the distribution*

$$r_i^T r_{j,j\neq i} = z/d \quad \textit{with probability } \eta = \frac{d!d!(m-d)!(m-d)!}{(d-z)!(d-z)!z!(m-2d+z)!m!} \tag{3.1}$$

where the integer z varies in the interval $[0,\ d]$.

Proof. The correlation between two columns is determined by the number of nonzero positions overlapped between them. The case where two columns have exactly z same nonzero positions, $0 \leq z \leq d$, occurs with the probability

$$\eta = \frac{\binom{m}{d-z}\binom{m-(d-z)}{d-z}\binom{m-2(d-z)}{z}}{\binom{m}{d}\binom{m}{d}} = \frac{d!d!(m-d)!(m-d)!}{(d-z)!(d-z)!z!(m-2d+z)!m!}$$

if d nonzero positions are selected uniformly at random in each column of $\mathbf{R}$. □

Lemma 3.2.2 (column correlation of $\mathbf{A}_L$)**.** *Let a_i and a_j denote the i-th and j-th columns of a matrix $\mathbf{A} \in \mathbf{A}_L$. Then the correlations between columns of $\mathbf{A}$ follows the distribution*

$$a_i^T a_{j,j\neq i} = \begin{cases} 1/d & \text{with probability } \eta = \frac{nd^2-md}{(n-1)m} \\ 0 & \text{with probability } 1-\eta \end{cases} \tag{3.2}$$

Proof. In the bipartite graph associated with $\mathbf{A}$, any variable node v_i, $i \in \{1, ..., n\}$, has d neighboring measurement nodes c_{b_k}, $1 \leq k \leq d$, where the subscript $b_k \in C$ which contains the indices of d measurement nodes connected to v_i. Each connected measurement node c_{b_k} also connects with other $(nd/m - 1)$ variable nodes v_j, where $j \in V_{b_k}$ which contains the indices of variable nodes connected to c_{b_k}, except for v_i. Since the variable node v_i has girth larger than 4, we have $V_{b_e} \bigcap V_{b_f} = \emptyset$, where $e, f \in \{1, ..., d\}$ and $e \neq f$, and then derive $|V_{b_1} \bigcup V_{b_2} \bigcup ... \bigcup V_{b_d}| = d(nd/m - 1)$. Therefore, in the set of n variable nodes excluding v_i, there are $(nd^2 - md)/m$ nodes each sharing exactly one common measurement node with v_i. It implies that any column of $\mathbf{A}$ has exactly $(nd^2 - md)/m$ correlated columns with correlation value $1/d$. Then the probability that any two distinct columns correlate to each other is derived as $\frac{nd^2-md}{(n-1)m}$. □

3.3 Optimal binary sensing matrix

In this section, the matrix $\mathbf{A}_L$ with $d = d_{max}$ is derived as the optimal binary matrix through searching the best RIP. Note that, for simplicity, we only calculate the values of RIC, as the smaller RIC implies the better RIP.

3.3.1 RIC of $\mathbf{A}_L$

As stated before, the RIC-δ_k will be estimated by bounding the extreme eigenvalues of all possible Gram matrices $\mathbf{A}_\psi^T \mathbf{A}_\psi$ with $\psi \subset \{1, 2, ..., n\}$ and $|\psi| = k$. According to the definition of $\mathbf{A}_L$, the elements of $\mathbf{A}_\psi^T \mathbf{A}_\psi \in \{0, 1, 1/d\}^{k\times k}$ follow a simple distribution as described below. The diagonal elements take the value 1, and

the off-diagonal elements obey the distribution shown in Lemma 3.2.2. With the distribution above, the RIC is immediately derived in Theorem 3.3.1.

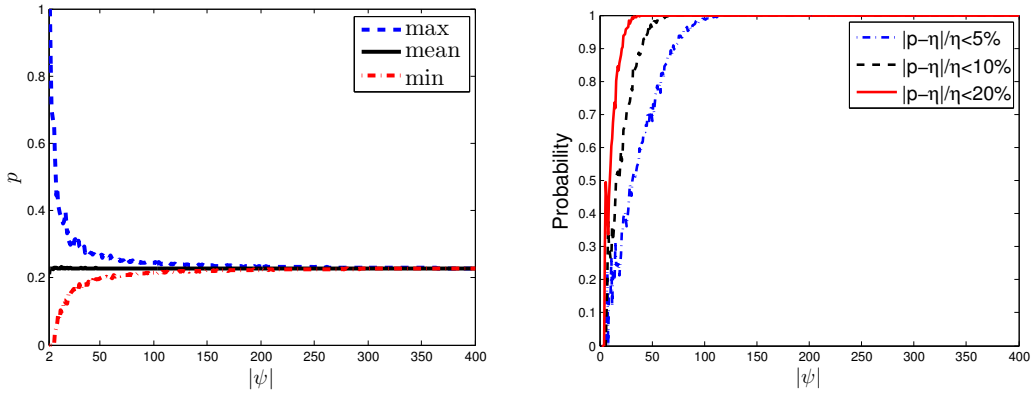
Theorem 3.3.1 (RIC-1)**.** *Consider a matrix* $\mathbf{A} \in \mathbf{A}_L$*. With the eigenvalue estimation method [43], the RIC of* $\mathbf{A}$ *can be derived as*

$$\delta_k = \frac{3k-2}{4d+k-2}. \tag{3.3}$$

Proof. See Proof 3.7.1. □

Remark: From the proof of Theorem 3.3.1, it can be inferred that the δ_k cannot reasonably reflect the average performance of $\mathbf{A}_L$, especially as $|\psi|$ increases, because the extreme eigenvalues seem hard to approximate for most submatrices $\mathbf{A}_\psi$. Specifically, to derive a sufficient condition, the eigenvalue estimation method in [43] considers only two extreme cases where the proportion of nonzero entries in the off-diagonal of $\mathbf{A}_\psi^T\mathbf{A}_\psi$, denoted as p, takes values 1 or 0.5. However, two cases above will not occur for most submatrices $\mathbf{A}_\psi$, since with increasing $|\psi|$, the proportion p defined above will center on $\eta < 1$ with higher property, as disclosed in Lemma 3.2.2. For better understanding, we test a real $\mathbf{A}_L$ matrix with size $(200, 400)$ and $d = 7$ in Figure 3.2, where the distribution of p is tested for all possible $\mathbf{A}_\psi^T\mathbf{A}_\psi$ when the submatrix size $|\psi|$ is fixed. It is clear that p will rapidly converge to the theoretical value $\eta = 0.2281 < 0.5$, as the submatrix size increases. It implies that the applied estimation method [43] is not reasonable when a relatively large sparsity $k = |\psi|$ is available in compressed sensing. In this case, the Wigner semicircle law [41] may be a better option, which works well as the guaranteed sparsity k is sufficiently large. The corresponding RIC-2 is derived in Theorem 3.3.2. Note that, in this thesis we are concerned only with the general case where the guaranteed sparsity k is relatively small. So in the following comparison between $\mathbf{A}_L$ and $\mathbf{A}_E$, we will exploit RIC-1 instead of RIC-2.

Theorem 3.3.2 (RIC-2)**.** *Consider a matrix* $\mathbf{A} \in \mathbf{A}_L$*. If the guaranteed sparsity*



(a) (b)

Figure 3.2: Given a real $\mathbf{A}_L$ matrix with size $(200, 400)$ and $d = 7$. The proportion p is tested for each set of submatrices $\mathbf{A}_\psi$ with given size $|\psi|$. In (a), the maximum, mean and minimum p that can be achieved by $\mathbf{A}_\psi^T\mathbf{A}_\psi$ are depicted at each $|\psi|$. As expected, the mean is equal to the theoretical value $\eta = 0.2281$, which is derived with Lemma 3.2.2. The probability that p centers on η with error bound $|p - \eta|/\eta$ is presented in (b).

$k \to \infty$, with the Wigner semicircle law, the RIC of $\mathbf{A}$ can be approximated as

$$\delta_k = \frac{k\eta + 2\sqrt{k\eta(1-\eta)} + 1}{k\eta - 2\sqrt{k\eta(1-\eta)} + 2d + 1}, \tag{3.4}$$

where $\eta = \frac{nd^2 - md}{(n-1)m}$ as in Lemma 3.2.2.

Proof. See Proof 3.7.2. □

3.3.2 RIC of $\mathbf{A}_E$

For a matrix $\mathbf{A} \in \mathbf{A}_E$ with given d and s, it can be observed that the Gram matrix $\mathbf{A}_\psi^T\mathbf{A}_\psi$ will take the value 1 in the diagonal, and take the values in the set $\{0, 1/d, ..., s/d\}$ in the off-diagonal. Then similarly to RIC-1, the RIC of $\mathbf{A}_E$ is also derived with the eigenvalue estimation method [43], as in Theorem 3.3.3.

Theorem 3.3.3 (RIC-3). *Consider a matrix $\mathbf{A} \in \mathbf{A}_E$ with given d and s, $2 \le s \le d - 1$ and $3 \le d \le m - 2$. Then with the eigenvalue estimation method in [43], the RIC of $\mathbf{A}$ can be derived as*

$$\delta_k = \begin{cases} \frac{(3k-2)s}{(k-2)s+4d} & if \;\; 3 \le d \le \frac{m}{2} \;\; and \;\; 2 \le s \le d-1 \\ \frac{(3k-2)s+(k-2)(m-2d)}{(k-2)s-(m-2d)k+2m} & if \;\; \frac{m}{2} < d \le m-2 \;\; and \;\; 2d-m \le s \le d-1 \end{cases} \tag{3.5}$$

Proof. See Proof 3.7.3. □

Remark: Similarly to RIC-1, RIC-2 is also derived by considering only two extreme cases. Precisely, the off-diagonal elements of $\mathbf{A}_\psi^T \mathbf{A}_\psi$ are assumed to only take the value s/d, or take binary values $\{0, s/d\}$ with equal probability. However, for random binary matrices, from Lemma 3.2.1, it can be observed that the probability that $\mathbf{A}_\psi^T \mathbf{A}_\psi$ satisfies the distribution above is very small. So it implies that, the δ_k in fact only takes care of the submatrices $\mathbf{A}_\psi$ of the worst performance, and ignores other more typical submatrices. This explains why in practice the real average performance usually behaves much better than the RIP expects.

3.3.3 Binary matrix with best RIP

This subsection demonstrates that the matrix $\mathbf{A}_L$ with $d = d_{max}$ should be the optimal binary sensing matrix with the best RIP, by comparing the RIC between $\mathbf{A}_L$ and $\mathbf{A}_E$, as detailed in Theorem 3.3.4.

Theorem 3.3.4. *Among all binary matrices, $\mathbf{A}_L$ with $d = d_{max}$ holds the smallest RIC, except for two special cases on $\mathbf{A}_E$:*

1) $\mathbf{A}_E$ exists with d satisfying $d/s > d_{max}$, where $d_{max} < d \le m/2$ and $2 \le s \le d-1$;

2) $\mathbf{A}_E$ exists with d satisfying $\frac{(k+1)(2d-m)}{6s+2(2d-m)} > d_{max}$, where $m/2 < d \le m-2$ and $2d-m \le s \le d-1$.

Proof. First, we need to prove that $\mathbf{A}_L$ achieves its smallest RIC at $d = d_{max}$. This result can be easily derived with RIC-1, in which the δ_k decreases as d increases. Next we are ready to provide the condition that $\mathbf{A}_L$ with $d = d_{max}$ holds smaller RIC than $\mathbf{A}_E$. Recall that $\mathbf{A}_E$ is defined with $3 \le d \le m-2$, and $d_{max} < m/2$

as will be derived in section 3.4.1. For clearer expression, in the following part, $\mathbf{A}_E$ is considered separately with three cases: $3 \leq d \leq d_{max}$, $d_{max} < d \leq m/2$, and $m/2 < d \leq m-2$. Let us first consider the case of $d \leq d_{max}$. By comparing RIC-1 with the first equation of RIC-3, it can be derived that $\frac{3k-2}{4d+k-2} < \frac{(3k-2)s}{(k-2)s+4d}$, if $3 \leq d \leq d_{max}$. This indicates that $\mathbf{A}_L$ holds smaller RIC than $\mathbf{A}_E$, when they share the same $d \leq d_{max}$. It follows that $\mathbf{A}_L$ with $d = d_{max}$ holds smaller RIC than all $\mathbf{A}_E$ with $d \leq d_{max}$. Subsequently, consider the case of $d_{max} < d \leq m/2$. Let RIC-1 with $d = d_{max}$ smaller than the first equation in RIC-3, it follows that $d_{max} \geq d/s$. Let RIC-1 with $d = d_{max}$ smaller than the second equation in RIC-3, we can derive $d_{max} \geq \frac{(k+1)(2d-m)}{6s+2(2d-m)}$ for the finial case of $m/2 < d \leq m-2$. Then the proof is completed. □

Remark:

1) Note that currently there is no explicit way to construct the two special $\mathbf{A}_E$ theoretically derived in Theorem 3.3.4. In fact, they probably do not really exist. For instance, the impossibility of the first case $\mathbf{A}_E$ with $d/s > d_{max}$ can be easily derived as follows. Suppose a matrix $\mathbf{A} \in \mathbf{A}_L$ with $d = d_{max}$. If d increases to $d = d_{max} + 1$, the matrix $\mathbf{A}$ will become an $\mathbf{A}_E$ matrix with $s \in \{2, ..., d-1\}$. In this case, $d_{max} > d/s$. As shown in Lemma 3.2.1, empirically, s multiplies much faster than d. This implies that as $d > d_{max}+1$, $\mathbf{A}_E$ should maintain $d_{max} > d/s$ rather than $d_{max} < d/s$. As for the second case $\mathbf{A}_E$ with $\frac{(k+1)(2d-m)}{6s+2(2d-m)} > d_{max}$, it is apparent that this kind of matrices is hard to analyze and construct.

2) Based on the fact above, it is reasonable to argue that $\mathbf{A}_L$ with $d = d_{max}$ holds the best RIP, which naturally indicates the optimal performance. It is necessary to note that, the condition that $\mathbf{A}_L$ has uniform *row* degrees is not necessary for us to define the optimal matrix with Theorems 3.3.1 and 3.3.3. In fact, this condition is only used in Lemma 3.2.2 and Theorem 3.3.2 for the convenience of analyzing the distribution of column correlations. So to obtain a relatively large d_{max}, except for special explanation, the *row* degrees are not restricted to be uniform in the following matrix construction. In addition, it is worth mentioning another general case where the column degrees are

not uniform either. In this case, we generally can construct an $\mathbf{A}_L$ matrix of average column degrees larger than the d_{max} derived on the uniform case, as will be detailed in the next two sections. Intuitively, it should also present better sensing performance than the optimal matrix $\mathbf{A}_L$ with $d = d_{max}$. In this chapter, we will not make a thorough inquiry into such kind of matrices, and still focus our attention on the binary matrix with uniform column degrees.

3.4 Deterministic construction

3.4.1 Estimation on the maximum column degree

Currently, for an $\mathbf{A}_L$ matrix of given size, it is still hard to exactly determine the maximum column degree d_{max}. In theory, this kind of bipartite graphs has been studied early as a combinatorial problem in [96] and the references therein. However, these works have no practical use because they only consider the matrix with infinite size. In practice, a rough estimation of d_{max} can be easily derived based on the fact that in a subgraph, the number of variable nodes included in the first two floors is not more than the total number of variable node [95]. Suppose the matrix is regular, it simply follows that $1 + d(dn/m - 1) \leq n$, and then $d_{max} < \sqrt{m} < m/2$.

Here, we will propose a more reasonable estimation method. From the definition of $\mathbf{A}_L$ with uniform $d = d_{max}$, it can be inferred that the corresponding bipartite graph with girth larger than 4 will achieve its maximum number of edges by additionally adding not more than $n - 1$ new edges, because if we add n edges, the column degree will turn into $d = d_{max} + 1$ and the cycles of length 4 will occur. Here, for the simplicity of analysis, we assume that the bipartite graph mentioned above has achieved its maximum number of edges, and would produce the cycles of length 4, if only one edge is introduced. In this case, each subgraph should hold all cycles of length 6 and include all the measurement nodes in its first two floors, such that any new connected measurement node will introduce the cycles of length 4. Again, suppose the bipartite graph is regular, it follows that

$$d_{max} + d_{max}(d_{max}n/m - 1)(d_{max} - 1) = m. \tag{3.6}$$

Note that the equality above is achieved only when a bipartite graph with girth larger than 4 can achieve its maximum number of edges with a regular form, but in fact this condition is hard to satisfy. An interesting evidence is that, the solution of formula (3.6) is usually not an integer. This implies that the $\mathbf{A}_L$ matrix with nonuniform column degrees probably get larger average column degrees than the d_{max} derived on the uniform case, as previously mentioned in the remark of Theorem (3.3.4). Of course, it also indicates that the estimation of formula (3.6) is not very accurate. Empirically, if the matrix is regular, it behaves slightly better than the real d_{max} that can be constructed with greedy algorithms; otherwise, it behaves pessimistically compared to the real values, as will be detailed in the final simulations.

3.4.2 Bipartite graph based construction algorithm

In practice, the maximum column degree d_{max} can be approached with greedy algorithms, although it is hard to be determined in theory. In this subsection, we will present an efficient bipartite graph based construction (BGC) algorithm to generate the bipartite graph with as many edges as possible, while with girth larger than 4.

Before detailing the BGC algorithm, we first introduce a known algorithm termed progressive edge-growth (PEG) algorithm, which is initially proposed to constrct the parity-check matrix of LDPC codes [95]. It can also be used to construct the bipartite graph mentioned above, since it attempts to build a bipartite graph with as few short cycles as possible. However, this algorithm is still imperfect in terms of both performance and complexity. More precisely, given the matrix size and column degrees, PEG can only construct the bipartite graph with large girth, rather than with girth larger than 4. In this case, it is inevitable to involve more computation to test the shortest cycles of each constructed matrix. To find the underlying maximum column degree, we generally have to construct all smaller column degrees, until the cycles of length 4 appear. Obviously, this enumerating process is computationally expensive. Moreover, in practice it is interesting to construct the optimal $\mathbf{A}_L$ matrix with nonuniform degrees, because as analyzed in the former subsection, it probably provide larger average column degrees and better sensing performance compared to the optimal $\mathbf{A}_L$ with uniform $d = d_{max}$. Obviously, the optimal distribution of

Algorithm 3.1 Bipartite graph based construction (BGC) algorithm

Initializations: Let $\mathcal{C}$ and $\mathcal{V}$ refer to the set of m measurement nodes and the set of n variable nodes, respectively. $\mathcal{I}$ is defined as the set of variable nodes still to be updated in current bipartite graph, which is initialized as $\mathcal{I} = \mathcal{V}$. $\mathcal{I}_i$ indicates the i-th element of $\mathcal{I}$. Searching for new edges will stop if $\mathcal{I}$ turns to be empty. Three subsets $\mathcal{C}_i$ with subscript $1 \leq i \leq 3$ are further defined to contain the measurement nodes appearing on the i-th floor of subgraph. If the subgraph has no measurement nodes on floor-i, the corresponding $\mathcal{C}_i = \varnothing$. The selected edges are collected in the set $\mathcal{E}$ which is initialized as $\varnothing$.

1: **for** i=1 to m **do**
2: **if** $\mathcal{I} = \varnothing$ **then**
3: break; // terminate the program and output the edge set $\mathcal{E}$;
4: **end if**
5: $\mathcal{I}' = \mathcal{I}$;
6: **for** j=1 to $|\mathcal{I}|$ **do**
7: Try to expand a subgraph from variable node $\mathcal{I}'_j$ to floor-3 with current edge set $\mathcal{E}$; and the the measurement nodes on the k-th floor are collected in the empty-initialized set $\mathcal{C}_k$, $1 \leq k \leq 3$;
8: **if** $\mathcal{C}_3 = \varnothing$ **then**
9: **if** $\mathcal{C} \backslash \{\mathcal{C}_1 \bigcup \mathcal{C}_2\} \neq \varnothing$ **then**
10: Introduce a new edge $(\mathcal{I}'_j, c)$ to the edge set $\mathcal{E}$ by $\mathcal{E} = \mathcal{E} \bigcup (\mathcal{I}'_j, c)$, where c is a measurement node randomly selected from the set $\mathcal{C} \backslash \{\mathcal{C}_1 \bigcup \mathcal{C}_2\}$;
11: **else**
12: Exclude the variable node $\mathcal{I}_j$ from $\mathcal{I}$, namely $\mathcal{I} = \mathcal{I} \backslash \mathcal{I}_j$;
13: **end if**
14: **end if**
15: **if** $\mathcal{C}_3 \neq \varnothing$ **then**
16: Introduce a new edge $(\mathcal{I}'_j, c)$ to the edge set $\mathcal{E}$ by $\mathcal{E} = \mathcal{E} \bigcup (\mathcal{I}'_j, c)$, where c is a measurement node randomly selected from the set $\mathcal{C}_3$;
17: **end if**
18: **end for**
19: **end for**

nonuniform column degrees is hard to be obtained by using PEG to enumerate all the possible cases. To address these problems, in the following part we propose a more efficient BGC algorithm, which can automatically provide the optimal uniform or nonuniform column degrees with a much lower complexity.

The BGC algorithm can be simply described with an iterative process, as sketched in Algorithm 3.1. At each iteration, each variable node is allowed to connect with at most one measurement node. The measurement node is randomly selected in the 3-rd floor of current subgraph to generate the cycles of length 6, if the floor can be

achieved by the current subgraph. Otherwise, if the subgraph does not include all measurement nodes, a measurement node outside the subgraph will be randomly chosen to avoid generating the cycles of length 4. The procedure above is repeated until no variable node has measurement nodes to update. According to the construction rule, the final generated bipartite graph will hold the following two properties: 1) each subgraph has two and only two floors containing all measurement nodes; 2) any further added edge will lead to the shortest cycles of length 4. This means that the generated bipartite graph indeed achieves its maximum number of edges under the constraint of girth larger than 4. Note that, in this case the column degrees are usually not uniform. The case of uniform column degrees can be easily derived by simply modifying the final iteration of the BGC algorithm. Specifically, if the number of edges selected in final iteration is less than the column number, these edges should be abandoned and then the matrix with maximum uniform degree is derived.

Compared to the PEG algorithm, it can be observed that the BGC algorithm also presents obvious advantages of complexity. Specifically, to select an edge, the BGC algorithm only needs to spread a subgraph with at most 3 floors; in contrast, for PEG algorithm, the subgraph has to be expanded as deep as possible. More importantly, the BGC can directly provide the optimal column degrees, while the PEG algorithm has to construct all possible column degrees.

3.4.3 Best RIP vs. best performance

As stated in the former section, the RIP is derived by considering the worst recovery case. So the best RIP does not certainly implies the best performance, in the setting where the recovery error to some extent is tolerated and a good average performance is preferred. Here, we give a real example, the PEG-constructed $\mathbf{A}_E$ matrix with d slightly larger than d_{max}, which is expected to provide better average performance than the optimal matrix we present. Due to greediness of PEG algorithm, the generated $\mathbf{A}_E$ matrix will contain a relatively few cycles of length 4 as the d is slightly larger than d_{max}. In this case, the nonzero correlations between distinct columns will take the value $1/d$ ($< 1/d_{max}$) with high probability, rather

than s/d ($> 1/d_{max}$), where $2 \leq s \leq d-1$. This implies that with high probability, the submatrix of the $\mathbf{A}_E$ matrix described above will take better orthogonality than the submatrix of optimal matrix. Then it should present better average performance. This conjecture is indeed validated in the following simulations.

3.5 Simulations

The simulations are divided into two parts with two aims. In the first part, the proposed BGC algorithm is compared with the PEG algorithm on the construction of the optimal binary matrix. According to the definition of the optimal binary matrix, it is clear that the larger column degree implies the better performance. In the second part, the performance advantage of the optimal binary matrix over other binary matrices is confirmed.

3.5.1 BGC vs. PEG

The optimal binary matrices with various sizes are constructed with BGC and PEG, whose column degrees are shown in Table 3.1. The optimal cases with nonuniform column degrees are also constructed with BGC, whose average values are denoted by BGCn. As expected, the BGC indeed can provide the nonuniform column degrees with average values larger than d_{max}. As for the construction of uniform cases, the BGC presents the same and even better performance than the PEG. This suggests that the BGC will be preferred in practice due to its significant advantage of complexity, as stated in the former section.

The theoretical estimation of formula (3.6) is also presented in Table 3.1, denoted by TE. Obviously, the estimation behaves pessimistically compared to the real values we can achieve. The difference between theory and practice comes from the fact that the formula (3.6) is derived on the assumption that the matrix is regular, while to obtain relatively large column degrees, this condition is not followed by PEG or BGC.

Table 3.1: For the matrices with girth larger than 4, the maximum column degrees achieved by BGC and PEG are shown. The term BGCn denotes the average of nonuniform column degrees constructed with BGC. The term TE denotes the theoretical estimation of formula 3.6.

m/n			0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
n	100	BGCn	1.45	2.37	3.2	3.99	4.68	5.27	5.93	6.55	7.21	7.69
		BGC	1	2	3	3	4	5	5	6	7	7
		PEG	1	2	3	3	4	5	5	6	7	7
		TE	1.44	2.00	2.50	2.96	3.39	3.79	4.18	4.56	4.90	5.25
	500	BGCn	2.68	4.45	5.99	7.40	8.67	9.86	10.99	12.08	13.07	14.21
		BGC	2	4	5	7	8	9	10	12	13	14
		PEG	2	4	5	7	8	9	10	11	12	13
		TE	2.12	3.12	3.98	4.76	5.48	6.16	6.80	7.41	8.00	8.57
	1000	BGCn	3.48	5.82	7.80	9.59	11.26	12.78	14.36	15.75	17.06	18.45
		BGC	3	5	7	9	11	12	14	15	17	18
		PEG	3	5	7	9	11	12	14	15	16	17
		TE	2.55	3.83	4.91	5.88	6.78	7.63	8.43	9.20	9.93	10.64

3.5.2 Performance of the optimal binary matrix

The performance of the optimal matrix $\mathbf{A}_L$ with $d = d_{max}$, is compared with other $\mathbf{A}_L$ and $\mathbf{A}_E$ matrices. The better matrix should recover larger sparsity k. Here for simulation simplicity, we only test the matrices of size (200, 400). The $\mathbf{A}_L$ matrices with $2 \leq d \leq d_{max}$ are constructed with PEG, and $d_{max} = 7$. Recall that the BGC and PEG generally present the same d_{max} when the column degrees are uniform. The $\mathbf{A}_L$ matrix with nonuniform column degrees is constructed with BGC, denoted by $\mathbf{A}_{Ln}$, whose average column degree is derived as $7.96 > d_{max}$. As stated before, in practice the notion of $\mathbf{A}_E$ covers most known binary sensing matrices. Obviously, we cannot test all of them. Here we just explore the random binary matrix $\mathbf{R}$ with column degrees $2 \leq d \leq 100$, and the PEG-constructed $\mathbf{A}_E$ matrix with $d_{max} < d \leq 100$. As it is known, the random binary matrix is typical in compressed sensing and has achieved comparable performance with Gaussian random matrices [79] [9]. The PEG-constructed $\mathbf{A}_E$ matrix is adopted here because it is expected to present better average performance than the optimal matrix in the former section. For comparison, the performance of Gaussian random matrices, denoted as $\mathbf{G}$, is also provided here. Note that, to reduce the simulation load, $\mathbf{A}_E$ and $\mathbf{R}$ are not tested with all possible column degrees d. However, as shown latter, the samples of d suffice to reflect the performance tendency.

Table 3.2: The largest sparsity k that allows to be recovered with probability larger than 99%. Recall that here $\mathbf{A}_L$ represents the PEG constructed binary matrix with girth larger than 4, $\mathbf{A}_E$ represents the PEG-constructed binary matrix with girth equal to 4, $\mathbf{R}$ denotes the sparse random binary matrices, and $\mathbf{G}$ denotes the Gaussian random matrix. $\mathbf{A}_L$ with $d = 7$ is the optimal binary matrix with uniform column degrees, and $\mathbf{A}_{Ln}$ is the optimal case with nonuniform column degrees. For each class of matrices with various column degrees, the best performance is highlighted in bold.

	d		2	3	4	5	6	**7**	8	9	10	11	12	13	14	15	20	30	40	50	100
OMP	k	$\mathbf{A}_L$	29	70	75	78	80	**81**	-	-	-	-	-	-	-	-	-	-	-	-	-
		$\mathbf{A}_E$	-	-	-	-	-	-	**83**	83	81	80	79	78	78	77	75	74	48	26	2
		$\mathbf{R}$	0	55	69	73	75	**76**	76	76	76	76	76	76	76	76	76	76	76	76	76
		$\mathbf{A_{Ln}}$	81																		
		$\mathbf{G}$	76																		
IHT	k	$\mathbf{A}_L$	1	34	47	53	55	**56**	-	-	-	-	-	-	-	-	-	-	-	-	-
		$\mathbf{A}_E$	-	-	-	-	-	-	**57**	55	53	50	48	47	45	44	37	27	16	7	1
		$\mathbf{R}$	0	14	38	45	**48**	48	47	46	45	44	44	44	43	43	38	30	22	18	3
		$\mathbf{A_{Ln}}$	56																		
		$\mathbf{G}$	53																		
SP	k	$\mathbf{A}_L$	17	62	71	73	74	**75**	-	-	-	-	-	-	-	-	-	-	-	-	-
		$\mathbf{A}_E$	-	-	-	-	-	-	**75**	74	74	73	72	71	71	77	71	70	25	9	1
		$\mathbf{R}$	0	48	65	68	**71**	71	71	71	71	71	71	71	70	70	70	70	70	70	69
		$\mathbf{A_{Ln}}$	75																		
		$\mathbf{G}$	73																		
BP	k	$\mathbf{A}_L$	25	57	**61**	61	61	61	-	-	-	-	-	-	-	-	-	-	-	-	-
		$\mathbf{A}_E$	-	-	-	-	-	-	**62**	61	59	59	58	58	57	57	54	51	36	18	1
		$\mathbf{R}$	0	45	55	**58**	58	58	58	58	57	57	57	56	56	56	55	53	50	49	37
		$\mathbf{A_{Ln}}$	61																		
		$\mathbf{G}$	63																		

To present convincing results, we test four representative recovery algorithms: orthogonal matching pursuit (OMP) algorithm [58] [59], iterative hard thresholding (IHT) algorithm [97], subspace pursuit (SP) algorithm [98] and basis pursuit (BP) algorithm [57]. Each simulation result is derived after 10^4 simulation runs. Both random binary matrices and Gaussian random matrices are randomly generated at each iteration. The sparse signals have nonzero elements i.i.d drawn from $N(0,1)$. And the correct recovery rates are measured with $1-\|\hat{\mathbf{x}}-\mathbf{x}\|_2/\|\mathbf{x}\|_2$.

The simulation results are illustrated in Table 3.2, where the largest sparsity k recovered with a probability larger than 99% is provided. As expected, the optimal matrix $\mathbf{A}_L$ with $d=7$ achieves the best performance among all binary matrices, except for some PEG-constructed $\mathbf{A}_E$ matrices with d slightly larger than d_{max}. Note that, although some $\mathbf{A}_L$ matrices with d slightly less than d_{max}, also present the best k with BP recovery, in fact their recovery rates are less than the optimal matrix. A similar thing also happens to the BGC-constructed $\mathbf{A}_L$ matrix with nonuniform column degrees, which does not present obvious advantages over $\mathbf{A}_L$ with $d=d_{max}$, though it is expected to perform better due to its larger average column degree. As for the fact that the optimal matrix performs a little worse compared to the PEG-constructed $\mathbf{A}_E$ matrices with d slightly larger than d_{max}, it has been explained in the former section. This is because compared to the optimal matrix, the $\mathbf{A}_E$ matrices mentioned above possess smaller average column correlations. Thus it can present better average performance, in the setting where 100% recovery rate is not required.

Moreover, it can be observed that in most cases the optimal binary matrix also presents better performance than Gaussian random matrices. Note that, the sensing performance is generally sensitive to the signs of matrices and signals [99]. In fact, the optimal binary matrix will present much better performance than Gaussian random matrices, if the sparse signals have unsigned elements, rather than the signed $N(0,1)$ elements as in the simulation.

3.6 Conclusion

This chapter has successfully proposed and constructed the optimal binary matrix, which even presents better performance over Gaussian random matrices in simulation. Based on the fact that the best RIP corresponds to the best performance, we are allowed to define the optimal matrix, even though the exact performance still cannot be derived with current theoretical methods. In terms of hardware-friendly implementation, as it did for LDPC codes [100], the optimal binary matrix can also be constructed with quasi-cyclic structure, which generally will suffer some performance loss due to the decrease of the maximum column degree. In fact, as detailed in [101], the result of this chapter can be simply extended to construct the optimal ternary matrix, by assigning the binary values ± 1 with equal probability to the deterministic nonzero positions of optimal binary matrix.

3.7 Proof

3.7.1 Proof of Theorem 3.3.1

Proof. As stated before, the RIC-δ_k can be estimated with the extreme eigenvalues of Gram matrix $\mathbf{A}_\psi^T\mathbf{A}_\psi$ with size $|\psi| = k$. In this chapter we exploit the eigenvalue estimation method in [43]. The k eigenvalues of $\mathbf{A}_\psi^T\mathbf{A}_\psi$ are typically represented with the order $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_k$. In the following part, the two extreme eigenvalues λ_k and λ_1 are successively estimated.

1. Let $\mathbf{B} = \mathbf{A}_\psi^T\mathbf{A}_\psi - \mathbf{I}$, where $\mathbf{I}$ is an identity matrix. Then with definition of $\mathbf{A}_L$, it is known that the symmetric matrix $\mathbf{B} \in \{0, 1/d\}^{k\times k}$ has the diagonal elements $\mathbf{B}_{ii} = 0$ and the off-diagonal elements $\mathbf{B}_{ij,i\neq j} = 0$ or $1/d$. Assume a normalized vector $\mathbf{x} = (x_1, \ldots, x_k)^T$ is the eigenvector corresponding to the minimal eigenvalue $\lambda_k(\mathbf{B})$, which be formulated as

$$\lambda_k(\mathbf{B}) = x^T\mathbf{B}x = \mathbb{1}^T[\mathbf{B} \circ (xx^T)]\mathbb{1}$$

where $\circ$ denotes the Hadamard product and $\mathbb{1} \in \mathbb{R}^k$ is an all-ones vector. Since

$\mathbf{B}$ is symmetric, with simultaneous permutations of the rows and columns of $\mathbf{B}$, we are allowed to suppose $x_i \geq 0$ for $i = 1, \ldots, t$ and $x_i < 0$ for $i = t+1, \ldots, k$. Accordingly, $\mathbf{x}\mathbf{x}^T$ allows to be divided into four parts as below

$$\mathbf{x}\mathbf{x}^T = \begin{bmatrix} \mathbf{X}_{t\times t} & \mathbf{X}_{t\times(k-t)} \\ \mathbf{X}_{(k-t)\times t} & \mathbf{X}_{(k-t)\times(k-t)} \end{bmatrix}$$

where the entries in $\mathbf{X}_{n\times n}$ and $\mathbf{X}_{(k-n)\times(k-n)}$ are nonnegative, while the entries in $\mathbf{X}_{n\times(k-n)}$ and $\mathbf{X}_{(k-n)\times n}$ are nonpositive. Furthermore, we need to define a matrix $\tilde{\mathbf{B}}$ with size same to $\mathbf{B}$

$$\tilde{\mathbf{B}} = \begin{bmatrix} 0 \times \mathbf{J}_{t\times t} & \frac{1}{d} \times \mathbf{J}_{t\times(k-t)} \\ \frac{1}{d} \times \mathbf{J}_{(k-t)\times t} & 0 \times \mathbf{J}_{(k-t)\times(k-t)} \end{bmatrix}$$

where $\mathbf{J}_{a\times b}$ is an all-ones matrix with size $a \times b$. It is easy to deduce that

$$\lambda_k(\tilde{\mathbf{B}}) = \min\{\mathbf{y}^T\tilde{\mathbf{B}}\mathbf{y} : \|\mathbf{y}\|_2 = 1\} \leq \mathbf{x}^T\tilde{\mathbf{B}}x \leq \mathbf{x}^T\mathbf{B}x = \lambda_k(\mathbf{B}).$$

Note that $\tilde{\mathbf{B}}$ preserve a rank not larger than than 2, and so it has at most two nonzero eigenvalues. Considering its trace and Frobenius norm, we have

$$\lambda_k(\tilde{\mathbf{B}}) = -\sqrt{\frac{t(k-t)}{d^2}},\ 0 \leq t \leq k.$$

If k is even, $\lambda_k(\tilde{\mathbf{B}}) \geq -\frac{k}{2d}$, with equality for $t = k/2$.
If k is odd, $\lambda_k(\tilde{\mathbf{B}}) \geq -\frac{\sqrt{k^2-1}}{2d}$, with equality for $t = (k-1)/2$ or $t = (k+1)/2$.

Combining these two cases, it is derived that $\lambda_k(\mathbf{B}) \geq \lambda_k(\tilde{\mathbf{B}}) \geq -\frac{k}{2d}$, with equality for $t = k/2$, k is even. So we have the minimum eigenvalue $\lambda_k(\mathbf{A}_\psi^T\mathbf{A}_\psi) \geq 1 - \frac{k}{2d}$.

2. Let $\mathbf{C} = \mathbf{A}_\psi^T\mathbf{A}_\psi - \frac{d-1}{d}\mathbf{I}$, then $\mathbf{C}_{ii} = 1/d$ and $\mathbf{C}_{ij,i\neq j} = 0$ or $1/d$. Assume a normalized vector $\mathbf{x} = (x_1, \ldots, x_k)^T$ being the eigenvector corresponding to $\lambda_1(\mathbf{C})$. By simultaneous permutations of $\mathbf{C}$ and $\mathbf{x}$, we can suppose $x_i \geq 0$ for $i = 1, \ldots, t$ and $x_i < 0$ for $i = t+1, \ldots, k$, and the maximal eigenvalue is

formulated as

$$\lambda_1(\mathbf{C}) = \mathbf{x}^T\mathbf{C}x = \mathbb{1}'[\mathbf{C} \circ (\mathbf{x}\mathbf{x}^T)]\mathbb{1}.$$

Further define

$$\tilde{\mathbf{C}} = \begin{bmatrix} \frac{1}{d} \times \mathbf{J}_{t\times n} & 0 \times \mathbf{J}_{t\times(k-t)} \\ 0 \times \mathbf{J}_{(k-t)\times t} & \frac{1}{d} \times \mathbf{J}_{(k-t)\times(k-t)} \end{bmatrix}$$

then

$$\lambda_1(\tilde{\mathbf{C}}) = \max\{\mathbf{y}^T\tilde{\mathbf{C}}\mathbf{y} : \|\mathbf{y}\|_2 = 1\} \geq \mathbf{x}^T\tilde{\mathbf{C}}x \geq \mathbf{x}^T\mathbf{C}x = \lambda_1(\mathbf{C}).$$

Note that $\tilde{\mathbf{C}}$ has the rank not more than 2, and equivalently it has at most two nonzero eigenvalues. Considering the trace and Frobenius norm, we have $\lambda_1(\tilde{\mathbf{C}}) = \frac{k+|k-2t|}{2d}$, and then

$$\lambda_1(\mathbf{C}) \leq \lambda_1(\tilde{\mathbf{C}}) \leq \frac{k}{d}$$

with equality for $t = 0$ or $t = k$. Thus, we can further derive

$$\lambda_1(\mathbf{A}_\psi^T\mathbf{A}_\psi) = \lambda_1(\mathbf{C}) + \frac{d-1}{d} \leq \frac{k+d-1}{d}.$$

3. Finally, combining results of 1) and 2), the RIC is derived as

$$\delta_k = \frac{\lambda_1(\mathbf{A}_\psi^T\mathbf{A}_\psi) - \lambda_k(\mathbf{A}_\psi^T\mathbf{A}_\psi)}{\lambda_1(\mathbf{A}_\psi^T\mathbf{A}_\psi) + \lambda_k(\mathbf{A}_\psi^T\mathbf{A}_\psi)} = \frac{3k-2}{4d+k-2}$$

.

□

3.7.2 Proof of Theorem 3.3.2

Proof. To derive the extreme eigenvalues of $\mathbf{A}_\psi^T\mathbf{A}_\psi$, we begin by seeking the extreme eigenvalues of $\mathbf{B} = (\mathbf{A}_\psi^T\mathbf{A}_\psi - \mathbf{I})$, where $\mathbf{I}$ is an identity matrix. Then $\mathbf{B}$ is a symmetric matrix with $\mathbf{B}_{ii} = 0$ and $\mathbf{B}_{ij,i\neq j}$ taking nonzero value $1/d$ with property η, as in Lemma 2.

Suppose

$$\mathbf{Q} = \frac{1}{\sqrt{\eta(1-\eta)}}(d\mathbf{B} - \eta\mathbf{J})$$

where $\mathbf{J}$ is an all-ones matrix. With Wigner semicircle law, the extreme eigenvalues $\frac{1}{\sqrt{k}}\mathbf{Q}$ with $k = |\psi|$, can be approximated as

$$-2 \leq \lambda(\frac{1}{\sqrt{k}}\mathbf{Q}) \leq 2$$

namely,

$$-2\sqrt{k\eta(1-\eta)} \leq \lambda(d\mathbf{B} - \eta\mathbf{J}) \leq 2\sqrt{k\eta(1-\eta)},$$

if $k \to \infty$ [102] [103].

Note that $\mathbf{dB}-\eta\mathbf{J}$ and $\eta\mathbf{J}$ are Hermitian matrices, and $\eta\mathbf{J}$ is positive semi-definite with rank equal to 1. With cauchy interlacing inequality [104], we can obtain

$$\lambda_i(d\mathbf{B} - \eta\mathbf{J}) \leq \lambda_i(d\mathbf{B}) \leq \lambda_{i-1}(d\mathbf{B} - \eta\mathbf{J})$$

for $1 < i \leq k$. Then it can be observed that

$$\lambda_2(\mathbf{B}) \leq \frac{1}{d} \cdot \lambda_1(d\mathbf{B} - \eta\mathbf{J}) \leq \frac{2}{d}\sqrt{k\eta(1-\eta)}$$

and

$$\lambda_k(\mathbf{B}) \geq \frac{1}{d} \cdot \lambda_k(d\mathbf{B} - \eta\mathbf{J}) \geq -\frac{2}{d}\sqrt{k\eta(1-\eta)}$$

As for $\lambda_1(\mathbf{B})$, in [105] it is approximated as

$$\lambda_1(\mathbf{B}) \approx \frac{1}{d}(k\eta + 1).$$

Now the extreme eigenvalues of $\mathbf{A}_\psi^T\mathbf{A}_\psi$ can be approximately formulated as

$$\lambda_1(\mathbf{A}_\psi^T\mathbf{A}_\psi) = \lambda_1(\mathbf{B}) + 1 \leq \frac{1}{d}(k\eta + 1) + 1$$

and

$$\lambda_k(\mathbf{A}_\psi^T\mathbf{A}_\psi) = \lambda_k(\mathbf{B}) + 1 \geq -\frac{2}{d}\sqrt{k\eta(1-\eta)} + 1$$

Finally, the RIC of $\mathbf{A}_\psi^T\mathbf{A}_\psi$ is deduced as

$$\delta_k = \frac{\lambda_1 - \lambda_k}{\lambda_1 + \lambda_k} = \frac{k\eta + 2\sqrt{k\eta(1-\eta)} + 1}{k\eta - 2\sqrt{k\eta(1-\eta)} + 2d + 1}$$

□

3.7.3 Proof of Theorem 3.3.3

The proof is similar to Proof A. Here we just give a sketch.

Proof. Recall that the given sensing matrix $\mathbf{A}$ with size (m, n).

1. If $3 \leq d \leq m/2$, it is known that $(\mathbf{A}_\psi^T\mathbf{A}_\psi)_{ii} = 1$ and $(\mathbf{A}_\psi^T\mathbf{A}_\psi)_{ij,i\neq j} \in \{0, \ldots, s/d\}$, $2 \leq s \leq d-1$.

   (a) Let $\mathbf{B} = \mathbf{A}_\psi^T\mathbf{A}_\psi - \mathbf{I}$, we can derive

   $$\lambda_k(\mathbf{B}) \geq \begin{cases} -sk/2d & \text{if } k \text{ is even} \\ -s\sqrt{k^2-1}/2d & \text{if } k \text{ is odd} \end{cases}$$

   and then

   $$\lambda_k(\mathbf{A}_\psi^T\mathbf{A}_\psi) = 1 + \lambda_k(\mathbf{B}) \geq 1 - \frac{sk}{2d}.$$

   (b) Let $\mathbf{C} = \mathbf{A}_\psi^T\mathbf{A}_\psi - (1 - \frac{s}{d})\mathbf{I}$, we can obtain $\lambda_1(\mathbf{C}) \leq ks/d$, and then

   $$\lambda_1(\mathbf{A}_\psi^T\mathbf{A}_\psi) \leq \frac{(k-1)s + d}{d}.$$

2. if $m/2 < d \leq m-1$, it is known that $(\mathbf{A}_\psi^T\mathbf{A}_\psi)_{ii} = 1$ and $(\mathbf{A}_\psi^T\mathbf{A}_\psi)_{ij,i\neq j} \in \{(2d-m)/d, \ldots, s/d\}$, where $2d - m \leq s \leq d-1$.

   (a) Let $\mathbf{B} = \mathbf{A}_\psi^T\mathbf{A}_\psi - (1 - \frac{2d-m}{d})\mathbf{I}$, it follows that

   $$\lambda_k(\mathbf{B}) \geq \begin{cases} \frac{k(2d-m-s)}{2d} & \text{if } k \text{ is even} \\ \frac{k(2d-m) - \sqrt{(2d-m)^2 - (k^2-1)s^2}}{2d} & \text{if } k \text{ is odd} \end{cases}$$

With $\lambda_k(\mathbf{B}) \geq -\frac{k(2d-m-s)}{2d}$, we have that

$$\lambda_k(\mathbf{A}_\psi^T \mathbf{A}_\psi) \geq \frac{k(2d-m-s)+2(m-d)}{2d}$$

(b) Let $\mathbf{C} = \mathbf{A}_\psi^T \mathbf{A}_\psi - (1-\frac{s}{d})\mathbf{I}$, we have $\lambda_1(\mathbf{C}) \leq ks/d$, and then derive

$$\lambda_1(\mathbf{A}_\psi^T \mathbf{A}_\psi) \leq \frac{(k-1)s+d}{d}.$$

3. Finally, with $\delta_k = \frac{\lambda_1-\lambda_k}{\lambda_1+\lambda_k}$, it can be easily deduced that

$$\delta_k = \begin{cases} \frac{(3k-2)s}{(k-2)s+4d} & \text{if } 3 \leq d \leq \frac{m}{2} \text{ and } 2 \leq s \leq d-1 \\ \frac{(3k-2)s+(k-2)(m-2d)}{(k-2)s-(m-2d)k+2m} & \text{if } \frac{m}{2} < d \leq m-2 \text{ and } 2d-m \leq s \leq d-1 \end{cases}$$

□

4 Random Bernoulli matrices with high compression ratio

In this chapter we study the sensing performance of random Bernoulli matrices with column size n much larger than row size m. It is observed that this kind of matrices will present a performance floor as the compression rate n/m increases. Importantly, the signal sparsity on the performance floor can be reasonably estimated with $\frac{1}{2}(\sqrt{\pi m/2}+1)$.

4.1 Introduction

In compressed sensing, it is natural to seek a sensing matrix with high compression ratio n/m. Empirically, the sensing performance will inevitably degrade with the increase of compression ratio. A question of practical interests then arise: how fast will the performance degrade as the compression ratio increases? This chapter is motivated to address this problem for the random Bernoulli matrix, which is popular in compressed sensing and performs as well as Gaussian ones. Surprisingly, as will be shown in the final simulation, the random Bernoulli matrix approximately presents a 'performance floor' regarding the increasing compression ratio. In other words, the decreasing speed of guaranteed sparsity k is very slow, and can even be ignored in the setting where m is fixed while n tends to infinity. This property enables the significant compression of high-dimensional signal with sparsity lower than the performance floor. Then it becomes interesting to theoretically evaluate the performance floor, namely the guaranteed sparsity k. Unfortunately, the exact

performance estimation is still an open problem in compressed sensing. Currently, the guaranteed sparsity k is often simply estimated with

$$k < \frac{1}{2}(\mu_m(\mathbf{A})^{-1} + 1), \tag{4.1}$$

where the parameter $\mu_m(\mathbf{A})$ represents the *maximum* absolute correlation between distinct columns of $\mathbf{A}$ [25]. However, as will be shown later, the formula (4.1) is only a sufficient condition for perfect recovery, such that the estimated k is usually much smaller than the real value we can achieve in practice. In this chapter, we will prove that the formula (4.1) can be modified to be a sufficient and necessary condition in the limit, if the maximum correlation $\mu_m(\mathbf{A})$ is replaced with the *average* absolute correlation between distinct columns of $\mathbf{A}$, denoted by $\mu_a(\mathbf{A})$. This improved estimation allows us to propose a simple formula $\frac{1}{2}(\sqrt{\pi m/2} + 1)$ to effectively approximate the performance floor of random Bernoulli matrices with fixed row size m. To the best of our knowledge, it is the first time that a theoretical estimation is reported being capable of reflecting the real sensing performance. Thus the contribution of this chapter is of both practical and theoretical interests.

The rest of the chapter is organized as follows. In the next section, by analyzing the proof process of formula (4.1), we demonstrate how the sufficient and necessary condition is approached by the average correlation. In section 4.3, we first calculate the average correlation of random Bernoulli matrix, then estimate its performance floor. The numerical evidence is illustrated and discussed in section 4.4. Finally, this chapter is concluded in section 4.5.

4.2 Estimation methods based on average correlation vs. maximum correlation

This section demonstrates how the formula (4.1) is modified to be a sufficient and necessary condition in the limit by analyzing its proof as shown in Theorem 1.1.2 in chapter 1. For easier reading, the theorem is reviewed below.

Theorem 4.2.1 (Theorem 1.1.2 in Chapter 1)**.** *If*

$$k < \frac{1}{2}(\mu_m(\mathbf{A})^{-1} + 1),$$

a k-sparse signal $\mathbf{x}$ *is the unique recovery of* $\mathbf{y} = \mathbf{A}\mathbf{x}$ *with the* ℓ_0*-minimization problem.*

Proof. We first need to prove a critical property, that is $Spark(\mathbf{A}) \geq 1 + \mu_m(\mathbf{A})^{-1}$. Let subset $\psi \subseteq \{1, 2, ..., n\}$ with cardinality denoted as $|\psi|$, $\mathbf{A}_\psi$ be a submatrix of $\mathbf{A}$ with columns indexed by ψ, and $\mathbf{A}_\psi^T$ be the transpose of $\mathbf{A}_\psi$. Consider the Gram matrix $\mathbf{G} = \mathbf{A}_\psi^T\mathbf{A}_\psi \in \mathbb{R}^{|\psi|\times|\psi|}$. Clearly its diagonal elements $g_{ii} = 1$, and off-diagonal elements $g_{ij} \leq \mu_m(\mathbf{A}) \leq 1$, $i \neq j$. Recall that the columns of $\mathbf{A}_\psi$ are linearly independent if and only if Gram matrix $\mathbf{G}$ has positive determinant, equivalently each eigenvalue is positive. With Gershgorin circle theorem [28], the i-th eigenvalue of $\mathbf{G}$ is bounded in the interval $[g_{ii}-r_i,\ g_{ii}+r_i]$, where $r_i = \sum_{j=1;j\neq i}^{|\psi|} |g_{ij}|$. To render all eigenvalues positive, we only require $1-(|\psi|-1)\mu_m(\mathbf{A}) > 0$. In other words, for any $|\psi| < \mu_m(\mathbf{A})^{-1}+1$, the columns of $\mathbf{A}_\psi$ are linear independent. From the definition of Spark as in Theorem 1.1.1, it can be easily deduced that $Spark(\mathbf{A}) \geq \mu_m(\mathbf{A})^{-1} + 1$. By merging the result with Theorem 1.1.1, we immediately derive the condition $k < \frac{1}{2}(\mu_m(\mathbf{A})^{-1} + 1)$ for the unique solution based on ℓ_0-minimization.

□

Now we focus our attention on the analysis of the proof of Theorem 4.2.1. First, we need to show how the maximum correlation $\mu_m(\mathbf{A})$ is involved. From the proof, it can be observed that the following inequality

$$1 - \sum_{j=1;j\neq i}^{|\psi|} |g_{ij}| > 0 \tag{4.2}$$

must hold to ensure the Gram matrix $\mathbf{A}_\psi^T\mathbf{A}_\psi$ being positive definite, where g_{ij} with $i \neq j$ denotes the correlation between the i-th and j-th columns of submatrix $\mathbf{A}_\psi$. Considering $|g_{ij}| \leq \mu_m(\mathbf{A})$, the condition in formula (4.2) is then simply relaxed to

$$1 - (|\psi| - 1)\mu_m(\mathbf{A}) > 0. \tag{4.3}$$

This relaxation process leads to the sufficient but unnecessary property of Theorem 4.2.1, which makes the estimation of Theorem 4.2.1 far away from the real performance.

To reflect the real performance, it is necessary to reduce the relaxation error between $\sum_{j=1;j\neq i}^{|\psi|}|g_{ij}|$ and $(|\psi|-1)\mu_m(\mathbf{A})$, such that the sufficient and necessary condition can be approached for Theorem 4.2.1. To this end, we propose to replace the maximum correlation $\mu_m(\mathbf{A})$ with the average correlation $\mu_a(\mathbf{A})$. In this case, as proved in Theorem 4.2.2, the relaxation error will be close to zero with high probability as the submatrix size $|\psi|$ increases, if the average correlation of submatrix $\mathbf{A}_\psi$ with high probability can be approximated by the average correlation of $\mathbf{A}$. Intuitively, the condition above should be well satisfied, if the compression ratio of $\mathbf{A}$ is sufficiently large. Thus the proposed method here is exploited to evaluate the performance of random Bernoulli matrix with high compression ratio.

Theorem 4.2.2. *Suppose $a_i \in [0,t]$ is arbitrarily distributed with mean of $e<t$ and variance of σ^2. Then for k elements i.i.d drawn from the distribution, we have $Pr(\sum_{i=1}^{k}a_i = ke)\to 1$ and $Pr(\sum_{i=1}^{k}a_i = kt)\to 0$, if $k\to\infty$.*

Proof. Suppose a binary distribution as below

$$a_i' = \begin{cases} 0 & \text{with probability } 1-p \\ t & \text{with probability } \quad p \end{cases}$$

where $p=e/t$. Then it follows that $Pr(a_i=t)\leq Pr(a_i'=t)$, and

$$Pr(\sum_{i=1}^{k}a_i = kt) \leq Pr(\sum_{i=1}^{k}a_i' = kt) = p^k = (\frac{e}{t})^k$$

According to the law of large numbers, it is known that

$$\lim_{k\to\infty} Pr(|\sum_{i=1}^{k}a_i - ke| < k\varepsilon) \geq 1-\frac{\sigma^2}{k\varepsilon^2}$$

where ε is an arbitrarily small positive constant. Then the conclusion of the theorem can be easily derived. □

4.3 Average column correlation of random Bernoulli matrix

To evaluate the performance floor, this section calculates the average column correlation of random Bernoulli matrices in Theorem 4.3.1. According to the law of large numbers, the average column correlation of random Bernoulli matrix with $n \gg m$ should be equivalent to the expected value of the absolute correlation between two arbitrary Bernoulli vectors. Therefore in Theorem 4.3.1 we only calculate the expected value mentioned above.

Theorem 4.3.1. *Suppose* $\mathbf{v}$ *and* $\mathbf{w}$ *are two distinct* normalized *column vectors of random Bernoulli matrix* $\mathbf{A} \in \mathbb{R}^{m\times n}$ *with i.i.d elements being* $\pm\frac{1}{\sqrt{m}}$ *equiprobably, and* $f(\mathbf{v},\mathbf{w}) = |\mathbf{v}^T\mathbf{w}|$ *denotes the correlation between them, then the expected value of* f *is derived with the following two forms:*

1)

$$\mathbb{E}(f) = \frac{2}{m}\frac{1}{2^m}\left\lceil\frac{m}{2}\right\rceil\binom{m}{\lceil\frac{m}{2}\rceil} \tag{4.4}$$

2)

$$\lim_{m\to\infty}\mathbb{E}(\sqrt{m}f) = \sqrt{\frac{2}{\pi}} \tag{4.5}$$

where $\lceil * \rceil$ *denotes the minimum integer not smaller than* $*$.

Proof. First, following from

$$f = |\mathbf{v}^T\mathbf{w}| = |\sum_{i=1}^{m}(\mathbf{v}_i\mathbf{w}_i)|,$$

f is equivalently written as

$$f = \frac{1}{m}|\sum_{i=1}^{m}\mathbf{z}_i|,$$

where $\mathbf{z}_i$ being ±1 equiprobably. Then the expected value can be formulated as

$$\mathbb{E}(f) = \frac{1}{m}\frac{1}{2^m}\sum_{i=1}^{m}(\binom{m}{i}|m-2i|)$$

where $\binom{m}{i} := \frac{m!}{(m-i)!i!}$. With

$$\binom{m}{i}|m-2i| = \begin{cases} m\binom{m-1}{0} & \text{if } i=0 \\ m\binom{m-1}{m-i-1} - m\binom{m-1}{i-1} & \text{if } 1 \le i \le \frac{m}{2} \\ m\binom{m-1}{i-1} - m\binom{m-1}{m-i-1} & \text{if } \frac{m}{2} < i < m \\ m\binom{m-1}{m-1} & \text{if } i = m \end{cases}$$

one can further derive that

$$\sum_{i=1}^{m}(\binom{m}{i}|m-2i|) = \begin{cases} 2m\binom{m-1}{\frac{m}{2}-1} & \text{if } m \text{ is even} \\ 2m\binom{m-1}{\frac{m-1}{2}} & \text{if } m \text{ is odd} \end{cases}$$

Finally, with $\binom{m-1}{i-1} = \frac{i}{m}\binom{m}{i}$, it follows that

$$\sum_{i=1}^{m}(\binom{m}{i}|m-2i|) = 2\lceil\frac{m}{2}\rceil\binom{m}{\lceil\frac{m}{2}\rceil}$$

The first conclusion of the theorem is thus obtained as

$$\mathbb{E}(f) = \frac{2}{m}\frac{1}{2^m}\lceil\frac{m}{2}\rceil\binom{m}{\lceil\frac{m}{2}\rceil}$$

We now turn to proving the second conclusion. According to Stirling's approximation:

$$m! = \sqrt{2\pi m}\left(\frac{m}{e}\right)^m \exp(\lambda_m),\ 1/(12m+1) < \lambda_m < 1/(12m)$$

$\mathbb{E}(f)$ can be described as

$$\mathbb{E}(f) = \frac{1}{2^m}\frac{m!}{\frac{m}{2}!\frac{m}{2}!} = \sqrt{\frac{2}{\pi m}}\exp(\lambda_m - 2\lambda_{\frac{m}{2}})$$

if m is even; otherwise,

$$\begin{aligned}\mathbb{E}(f) &= \frac{m+1}{m}\frac{1}{2^m}\frac{m!}{\frac{m+1}{2}!\frac{m-1}{2}!} \\ &= \sqrt{\frac{2}{\pi m}}\left(\frac{m^2}{m^2-1}\right)^{\frac{m}{2}}\exp(\lambda_m - \lambda_{\frac{m+1}{2}} - \lambda_{\frac{m-1}{2}})\end{aligned}$$

Then we have $\lim_{m\to\infty} \mathbb{E}(\sqrt{m}f) = \sqrt{\frac{2}{\pi}}$, whenever m is even or odd. The proof is completed. □

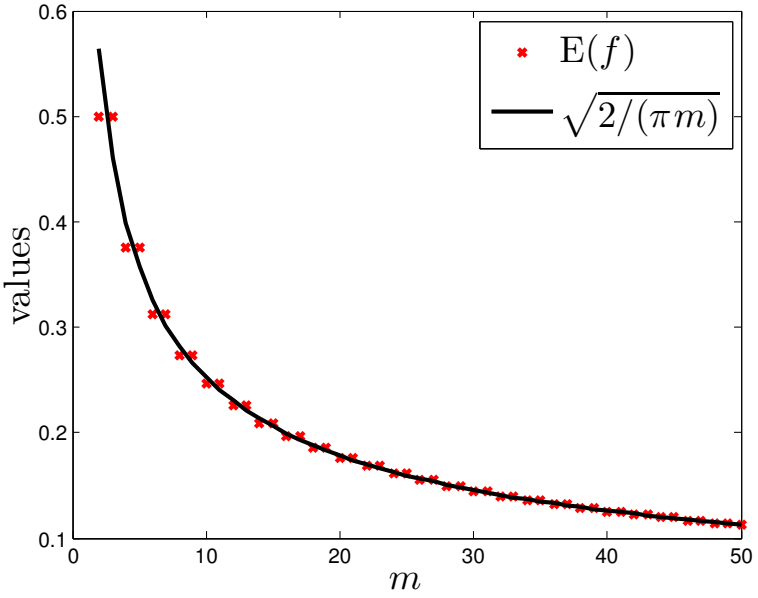


Figure 4.1: The values of $\mathbb{E}(f)$ in formula (4.4) and $\sqrt{2/(\pi m)}$ over varying m.

Note that the formula (4.5) in fact converges very fast, and can be satisfied with a relatively small m (on the order of tens). This implies that the expected value $\mathbb{E}(f)$ in formula (4.4) allows to be approximately written as $\mathbb{E}(f) = \sqrt{2/(\pi m)}$. For confirmation, the coincidence between $\mathbb{E}(f)$ and $\sqrt{2/(\pi m)}$ is illustrated in Figure 4.3. Then the average correlation $\mu_a(\mathbf{A})$ of random Bernoulli matrices with $n \gg m$ can also be approximated as $\sqrt{2/(\pi m)}$. In this case, the guaranteed sparsity on the performance floor of random Bernoulli matrices can be estimated with

$$k_{avr} = \frac{1}{2}(\sqrt{\pi m/2} + 1) \tag{4.6}$$

which is derived by replacing $\mu_m(\mathbf{A})$ with $\mu_a(\mathbf{A}) = \sqrt{2/(\pi m)}$ in formula (4.1). As it is expected in the former section, k_{avr} should be close to the real performance floor. Note that the maximum correlation of random Bernoulli matrices without same columns is equal to 1. For comparison, in the following simulation we also consider the performance floor estimated with $\mu_m(\mathbf{A}) = 1$, which is simply derived

as

$$k_{max} = \frac{1}{2}(1+1) = 1. \tag{4.7}$$

by incorporating $\mu_m(\mathbf{A}) = 1$ into formula (4.1). As will be shown later, k_{avr} performs much better than k_{max}.

4.4 Numerical Simulations

In this section, we will show that the random Bernoulli matrix indeed approximately presents a performance floor, which can be effectively estimated with k_{avr}. To illustrate the performance floor, the guaranteed sparsity k of random Bernoulli matrices with fixed m and increasing n is derived in Table 4.1. Before analyzing the data, we first briefly introduce the simulation setting. The sparse signal $\mathbf{x}$ with sparsity k is randomly generated in each simulation, and recovered with subspace pursuit algorithm [67]. The recovery rate is measured with 1-$||\hat{\mathbf{x}} - \mathbf{x}||_2/||\mathbf{x}||_2$. Note that here we only consider the largest k that can be recovered with rate larger than 0.99, because the perfect recovery is hard to be validated with simulation unless we can enumerate all possible distributions of k nonzero elements. Each result in Table 4.1 is derived after 10000 simulation runs.

Note that in Table 4.1 the compression ratio n/m exponentially increases, while the decreasing speed of k is very slow and can even be ignored compared with the fast increasing n. Specifically, as n increases, the sparsity k will decrease in a step not greater than 1. The relevant results are highlighted in red in Table 4.1. This implies that there indeed approximately exists a performance floor for each row size m. For better understanding, we further draw their performance curves in Figure 4.2. Note that, due to the limitation of computer memory, as shown in Table 4.1, we cannot test enough samples n to describe the performance floor, especially as m increases. Here, the performance floor defined for each m is quantified only with the mean of the first five k among the results labeled in red in Table 4.1. The quantified result is denoted with p_f. It is important to note that the gradual performance degradation is inevitable with the fast increasing of n. In fact, the whole performance floor is hard to be accurately reflected with the parameter p_f which only considers five

Table 4.1: The largest k guaranteed by random Bernoulli matrices of size (m, n) with recovery rates larger than 0.99. For each m, all k with decreasing step smaller than 2 are highlighted in bold red.

n/m		2^1	2^2	2^3	2^4	2^5	2^6	2^7	2^8	2^9	2^{10}	2^{11}	2^{12}	2^{13}
m	6^2	5	**3**	**3**	**2**	**2**	**2**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
	8^2	12	9	7	**5**	**5**	**4**	**3**	**3**	**3**	**2**	**2**	**2**	**2**
	10^2	22	16	12	10	**8**	**7**	**6**	**5**	**5**	**4**	**4**	**4**	**3**
	12^2	34	24	19	16	13	**11**	**10**	**9**	**8**	**7**	**6**	**6**	**5**
	14^2	50	35	27	22	19	16	14	**12**	**11**	**10**	**9**	**9**	**8**
	16^2	68	50	38	30	26	22	19	17	**15**	**14**	**13**	**12**	**11**

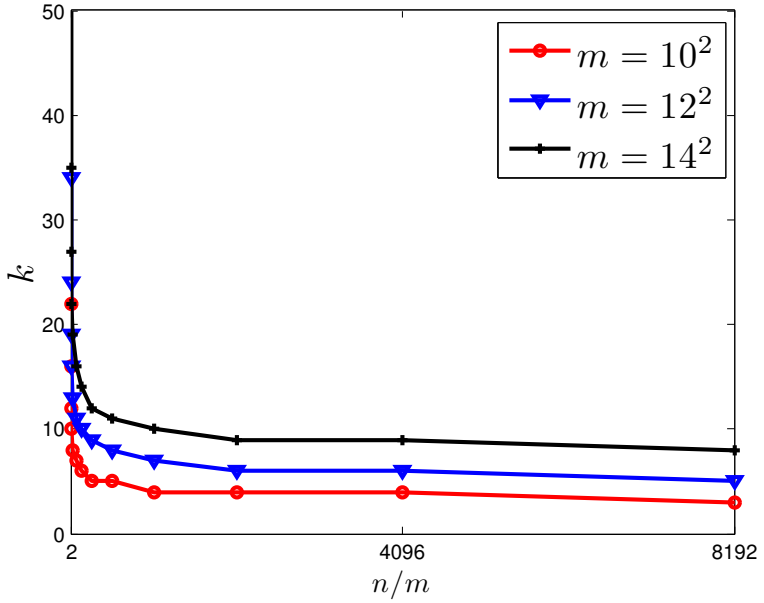


Figure 4.2: The performance curves of three random Bernoulli matrices from Table 4.1, with m=10^2, 12^2 and 14^2.

relatively good samples, especially when the matrix row size m is relatively large. As m increases, p_f should behave better than the real performance floor, since with five samples it can only consider few relatively large k and ignores most other smaller k on the performance floor. In this case, it is reasonable to infer that the distance between p_f and the estimation k_{avr} will increase with the increasing of m, if k_{avr} is close to the real performance. This conjecture is validated in the following simulation.

In Figure 4.3, we compare the two estimations k_{avr} and k_{max} against the performance floor measured with p_f. As it is expected, the estimation k_{avr} based on average correlation performs much better than the estimation k_{max} based on maximum cor-

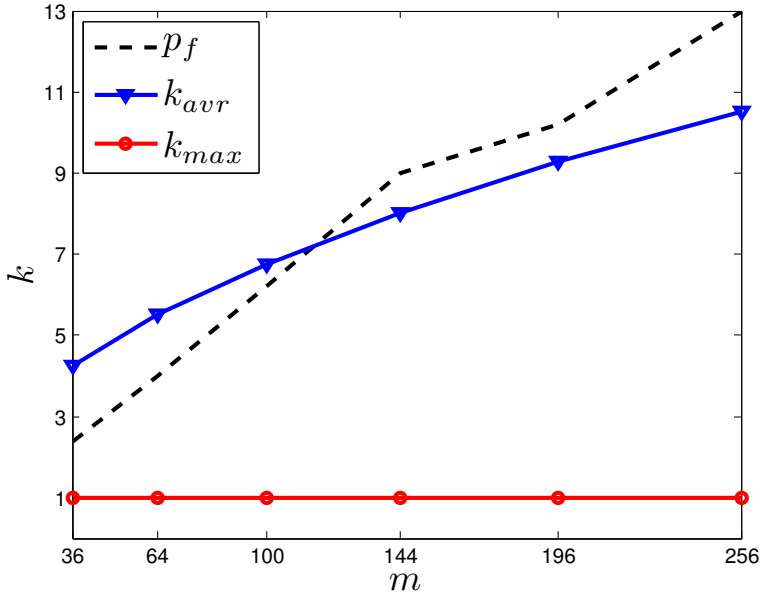


Figure 4.3: The performance floor estimated with p_f, and the theoretical estimations k_{avr} in formula (4.6) and k_{max} in formula (4.7).

relation. Precisely, the estimation k_{avr} is very close to p_f while the estimation k_{max} is of no practical use. From Figure 4.3, it is also observed that the error between p_f and k_{avr} tends to increase with the increasing of m. As stated before, this is due to the fact that p_f tends to behave better than the real performance floor as m increases. In fact, the error between k_{avr} and real performance floor should be smaller than the result shown in Figure 4.3.

4.5 Conclusion

This chapter has shown that the random Bernoulli matrix approximately presents a performance floor regarding the increasing compression ratio, which enables the significant compression of high-dimensional sparse signals. More importantly, we successfully estimated the performance floor by exploring the average correlation between distinct columns of random Bernoulli matrix, instead of the traditional maximum column correlation. Empirically, the result of this chapter also applies to Gaussian random matrix.

Part II

Random Projection

5 Sparse matrix based random projection for classification

As a typical dimensionality reduction technique, random projection can be simply implemented with linear projection, while preserving the pairwise distances of high-dimensional data with high probability. Considering this technique is mainly exploited for the task of classification, this chapter is developed to study the construction of random matrix from the viewpoint of feature selection, rather than of traditional distance preservation. This yields a somewhat surprising theoretical result, that is, the sparse random matrix with exactly one nonzero element per column, can present better feature selection performance than other more dense matrices, if the projection dimension is not much smaller than the number of feature elements. The theoretical conjecture is confirmed with extensive classification experiments.

5.1 Introduction

Random projection attempts to project a set of high-dimensional data into a low-dimensional subspace without distortion on pairwise distance. This brings attractive computational advantages on the collection and processing of high-dimensional signals. In practice, it has been successfully applied in numerous fields concerning categorization, as shown in [106] and the references therein. Currently the theoretical study of this technique mainly falls into one of the following two topics. One is concerned with the construction of random matrix in terms of distance preservation.

In fact, this problem has been sufficiently addressed along with the emergence of Johnson-Lindenstrauss (JL) lemma [7]. The other popular topic is about the design of classifier combined with random projection, as detailed in [107] and the references therein. Specifically, it may be worth mentioning that, recently the performance consistency of SVM on random projection is proved by exploiting the underlying connection between JL lemma and compressed sensing [108] [9].

Based on distance preservation, Gaussian random matrices [109] and a few of sparse random matrices [8, 110, 111] have been sequentially proposed for random projection. In terms of implementation complexity, it is clear that the sparse random matrix is more attractive. Unfortunately, as it will be detailed in section 5.2.2, theoretically the sparser matrix tends to yield weaker distance preservation. This fact largely weakens our interests on the pursuit of sparser random matrix. However, it is necessary to mention a problem ignored for a long time, that is random projection is mainly used for various tasks of classification, which prefer to maximize the distances between different classes, rather than merely preserve their distances. In this sense, it may be interesting to study random projection from the viewpoint of feature selection, even with some loss on distance preservation. Of course, the JL lemma cannot be absolutely ignored, and in fact it is still the premise of conducting classification, as it promises the stability of data structure during random projection.

In this chapter, we indeed derive the random matrix with the best feature selection performance, by analyzing the relation between the feature distribution of high-dimensional data and the sparsity of random matrix. The proposed matrix presents currently the most sparse structure with only one random nonzero position per column. Theoretically, it is expected to provide better classification performance over other more dense matrices, if the projection dimension is not much smaller than the number of feature elements. This conjecture is confirmed with extensive experiments on both synthetic and real data.

The rest of the chapter is organized as follows. In section 5.2, the JL lemma is first introduced, and then the distance preservation property of sparse random matrix over varying sparsity is evaluated. In section 5.3, a theoretical frame is proposed to evaluate the relation between the feature distribution of high-dimensional data

and the sparsity of random matrix. According to the theoretical results above, the sparse matrix with better performance over other more dense matrices is derived and analyzed in section 5.4. In section 5.5, the performance advantage of the proposed sparse matrix is verified by performing binary classification on both synthetic data and real data. The real data include three representative datasets in dimension reduction: face image, DNA microarray and text document. Finally, this chapter is concluded in section 5.6.

5.2 Preliminaries

This section first briefly reviews JL lemma, and then evaluates the distance preservation of sparse random matrix over varying sparsity.

For easy reading, we begin by introducing some basic notations for this chapter. A random matrix is denoted by $\mathbf{R} \in \mathbb{R}^{k\times d}$, $k < d$. r_{ij} is used to represent the element of $\mathbf{R}$ at the i-th row and the j-th column, and $\mathbf{r} \in \mathbb{R}^d$ indicates the row vector of $\mathbf{R}$. Considering this chapter is concerned with binary classification, in the following study we tend to define two samples $\mathbf{v} \in \mathbb{R}^d$ and $\mathbf{w} \in \mathbb{R}^d$, randomly drawn from two different patterns of high-dimensional datasets $\mathcal{V} \subset \mathbb{R}^d$ and $\mathcal{W} \subset \mathbb{R}^d$, respectively. The inner product between two vectors is typically written as $\langle \mathbf{v}, \mathbf{w} \rangle$. To distinguish from scalar variable, the vector is written in bold. In the proofs of the following theorems, we typically use $\Phi(*)$ to denote the cumulative distribution function of $N(0, 1)$. The minimal integer not less than $*$, and the maximum integer not larger than $*$ are denoted with $\lceil * \rceil$ and $\lfloor * \rfloor$.

5.2.1 Johnson-Lindenstrauss (JL) lemma

The distance preservation of random projection is supported by JL lemma. In the past decades, several variants of JL lemma have been proposed in [112–114]. For the convenience of the proof of the following Lemma 5.2.1, here we recall the version of [114] in the following Theorem A.2.1. It is clear that the basic requirement of JL lemma is to let $\mathbb{E}(r_{ij}) = 0$ and $\mathbb{E}(r_{ij}^2) = 1$. In addition, to obtain a relatively good distance preservation, JL lemma is expected to possess a tight concentration in the

Theorem A.2.1.

Theorem 5.2.1. *[114] Consider random matrix* $\mathbf{R} \in \mathbb{R}^{k\times d}$*, with each entry* r_{ij} *chosen independently from a distribution that is symmetric about the origin with* $\mathbb{E}(r_{ij}^2) = 1$*. For any fixed vector* $\mathbf{v} \in \mathbb{R}^d$*, let* $\mathbf{v}' = \frac{1}{\sqrt{k}}\mathbf{R}\mathbf{v}$*.*

- *Suppose* $B = \mathbb{E}(r_{ij}^4) < \infty$*. Then for any* $\epsilon > 0$*,*

$$\Pr(\|\mathbf{v}'\|^2 \leq (1-\epsilon)\|\mathbf{v}\|^2) \leq e^{-\frac{(\epsilon^2-\epsilon^3)k}{2(B+1)}} \tag{5.1}$$

- *Suppose* $\exists L > 0$ *such that for any integer* $m > 0$*,* $\mathbb{E}(r_{ij}^{2m}) \leq \frac{(2m)!}{2^m m!}L^{2m}$*. Then for any* $\epsilon > 0$*,*

$$\begin{aligned}\Pr(\|\mathbf{v}'\|^2 \geq (1+\epsilon)L^2\|\mathbf{v}\|^2) &\leq ((1+\epsilon)e^{-\epsilon})^{k/2} \\ &\leq e^{-(\epsilon^2-\epsilon^3)\frac{k}{4}}\end{aligned} \tag{5.2}$$

5.2.2 Sparse random projection matrices

Up to now, only a few random matrices are theoretically proposed for random projection. They can be roughly classified into two typical classes. One is the Gaussian random matrix with entries i.i.d dawn from $N(0,1)$, and the other is the sparse random matrix with elements satisfying the distribution below:

$$r_{ij} = \sqrt{q} \times \begin{cases} 1 & \text{with probability } 1/2q \\ 0 & \text{with probability } 1 - 1/q \\ -1 & \text{with probability } 1/2q \end{cases} \tag{5.3}$$

where q is allowed to be 2, 3 [8] or $\sqrt{d}$ [110]. Apparently the larger q indicates the higher sparsity.

Naturally, an interesting question arises: can we continue improving the sparsity of random projection? Unfortunately, as illustrated in Lemma 5.2.1, the concentration of JL lemma will decrease as the sparsity increases. In other words, the higher sparsity leads to weaker performance on distance preservation. However, as it will be disclosed in the following part, the classification tasks involving random projection are more sensitive to feature selection rather than to distance preservation.

Lemma 5.2.1. *Consider a class of random matrices $R \in \mathbb{R}^{k\times d}$, with each entry r_{ij} of the distribution as in formula* (5.3)*, where $q = k/s$ and $1 \leq s \leq k$ is an integer. Then these matrices satisfy JL lemma with different levels: the more sparse matrix (with smaller s) presents worse expectation on the pairwise distance preservation.*

Proof. With formula (5.3), it is easy to derive that the proposed matrices satisfy the distribution defined in Theorem A.2.1. In this sense, they also obey JL lemma if the two constraints corresponding to formulas 5.1 and 5.2 could be further proved. For the first constraint corresponding to formula (5.1):

$$\begin{aligned} B &= \mathbb{E}(r_{ij}^4) \\ &= (\sqrt{k/s})^4 \times (s/2k) + (-\sqrt{k/s})^4 \times (s/2k) \\ &= k/s < \infty \end{aligned} \tag{5.4}$$

then it is approved.

For the second constraint corresponding to formula (5.2):

for any integer $m > 0$, derive $\mathbb{E}(r^{2m}) = (k/s)^{m-1}$, and

$$\frac{\mathbb{E}(r_{ij}^{2m})}{(2m)!L^{2m}/(2^m m!)} = \frac{2^m m! k^{m-1}}{s^{m-1}(2m)!L^{2m}}.$$

Since $(2m)! \geq m!m^m$,

$$\frac{\mathbb{E}(r_{ij}^{2m})}{(2m)!L^{2m}/(2^m m!)} \leq \frac{2^m k^{m-1}}{s^{m-1}m^m L^{2m}},$$

let $L = (2k/s)^{1/2} \geq \sqrt{2}(k/s)^{(m-1)/2m}/\sqrt{m}$, further derive

$$\frac{\mathbb{E}(r_{ij}^{2m})}{(2m)!L^{2m}/(2^m m!)} \leq 1.$$

Thus $\exists L = (2k/s)^{1/2} > 0$ such that

$$\mathbb{E}(r_{ij}^{2m}) \leq \frac{(2m)!}{2^m m!}L^{2m}$$

for any integer $m > 0$. Then the second constraint is also proved.

Consequently, it is deduced that, as s decreases, B in formula (5.4) will increase, and

subsequently the boundary error in formula (5.1) will get larger. And this implies that the sparser the matrix is, the worse the JL property. □

5.3 Theoretical Framework

As it will be shown latter, the feature selection performance can be simply indicated by the products between the difference between two distinct high-dimensional vectors and the sampling/row vectors of random matrix. For the convenience of analysis, we first assume a general distribution for the feature difference between two distinct high-dimensional vectors in subsection 5.3.1, and then in subsection 5.3.2, analyze the products mentioned above with respect to the sparsity of sampling vectors, as illustrated in Theorems 5.3.1, 5.3.2 and 5.3.3. Note that to make the thesis more readable, the proofs of the three theorems mentioned above are included in the section Proof 5.7.

5.3.1 Difference between two distinct high-dimensional vectors

From the viewpoint of feature selection, the random projection is expected to maximize the difference between two arbitrary samples $\mathbf{v}$ and $\mathbf{w}$ from two different datasets $\mathcal{V}$ and $\mathcal{W}$, respectively. Usually the difference is measured with the Euclidean distance denoted by $\|\mathbf{Rz}\|_2$, $\mathbf{z} = \mathbf{v} - \mathbf{w}$. Then the search for a good random projection is equivalent to seeking the distribution of the row vector $\hat{\mathbf{r}}$ such that

$$\hat{\mathbf{r}} = \underset{\mathbf{r}}{\operatorname{argmax}}\{|\langle \mathbf{r}, \mathbf{z} \rangle|\}, \tag{5.5}$$

as the row vectors of $\mathbf{R}$ are mutually independent. So in the following part we need to evaluate only the distribution of row vectors. For convenient analysis, the two classes of high-dimensional data are further ideally divided into two parts, $\mathbf{v} = [\mathbf{v}^f \ \mathbf{v}^r]$ and $\mathbf{w} = [\mathbf{w}^f \ \mathbf{w}^r]$, where $\mathbf{v}^f$ and $\mathbf{w}^f$ denote the feature elements containing the discriminative information between $\mathbf{v}$ and $\mathbf{w}$ such that $\mathbb{E}(v_i^f - w_i^f) \neq 0$, while $\mathbf{v}^r$ and $\mathbf{w}^r$ represent the redundant elements such that $\mathbb{E}(v_i^r - w_i^r) = 0$ with a tiny variance. Sub-

sequently, $\mathbf{r} = [\mathbf{r}^f\ \mathbf{r}^r]$ and $\mathbf{z} = [\mathbf{z}^f\ \mathbf{z}^r]$ are also segmented into two parts corresponding to the coordinates of feature elements and redundant elements, respectively. Then the task of random projection can be reduced to maximizing $|\langle \mathbf{r}^f, \mathbf{z}^f \rangle|$, which implies that the redundant elements have no impact on the feature selection. Therefore, for simpler expression, in the following part the high-dimensional data is assumed to have only feature elements except for specific explanation, and the superscript f is simply dropped. Note that, in this chapter the minimization of the difference between intra-class samples is not considered, and their difference is ideally assumed to be zero.

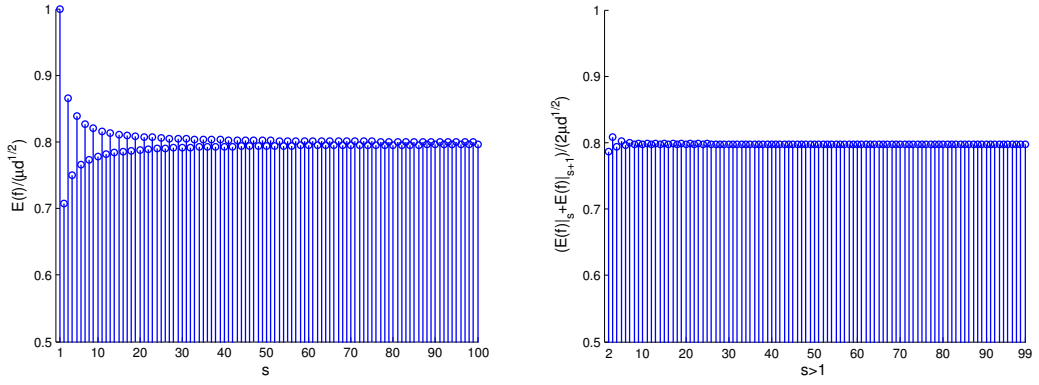
To explore the desired $\hat{\mathbf{r}}_i$ in formula (5.5), it is necessary to know the distribution of $\mathbf{z}$. However, in practice the distribution is hard to be characterized since the locations of feature elements are usually unknown. As a result, we have to make a relaxed assumption on the distribution of $\mathbf{z}$. For a given real dataset, the values of v_i and w_i should be limited. This allows us to assume that their difference z_i is also bounded in amplitude, and acts as some unknown distribution. For the sake of generality, in this thesis $|z_i|$ is regarded as approximately satisfying a Gaussian distribution with a random sign. Then the distribution of z_i can be formulated as

$$z_i = \begin{cases} x & \text{with probability } 1/2 \\ -x & \text{with probability } 1/2 \end{cases} \tag{5.6}$$

where $x \in N(\mu, \sigma^2)$, μ is a positive number, and $\Pr(x > 0) = 1 - \epsilon$, $\epsilon = \Phi(-\frac{\mu}{\sigma})$ is a small positive number.

5.3.2 Products between high-dimensional vectors and random sampling vectors with varying sparsity

This subsection mainly tests the feature selection performance of random row vector with varying sparsity. For the sake of comparison, Gaussian random vectors are also evaluated. Recall that under the basic requirement of JL lemma, that is $\mathbb{E}(r_{ij}) = 0$ and $\mathbb{E}(r_{ij}^2) = 1$, Gaussian matrix has elements i.i.d drawn from $N(0, 1)$, and sparse random matrix has elements distributed as in formula (5.3) with $q \in$



(a) (b)

Figure 5.1: The process of $\frac{1}{\mu\sqrt{d}}\mathbb{E}(f)$ converging to $\sqrt{2/\pi}$ (≈ 0.7979) with increasing s is described in (a); and in (b) the average value of two $\frac{1}{\mu\sqrt{d}}\mathbb{E}(f)$ with adjacent s (> 1), namely $\frac{1}{2\mu\sqrt{d}}(\mathbb{E}(f)|_s + \mathbb{E}(f)|_{s+1})$, is approved very close to $\sqrt{2/\pi}$. Note that $\mathbb{E}(f)$ is calculated with the formula provided in Theorem 5.3.1.

$\{d/s : 1 \leq s \leq d, s \in \mathbb{N}\}$.

Then from the following Theorems 5.3.1, 5.3.2 and 5.3.3, we present two crucial random projection results for the high-dimensional data with feature difference $|z_i|$ varying within a certain range:

- Random matrices will achieve the best feature selection performance as only one feature element is sampled by each row vector; in other words, the solution to the formula (5.5) is obtained when $\mathbf{r}$ randomly has $s = 1$ nonzero elements;
- The desired sparse random matrix mentioned above can also obtain better feature selection performance than Gaussian random matrices.

Note that, for better understanding, we first prove in Theorem 5.3.1 a relatively simple case where $z_i \in \{\pm\mu\}$, and then in Theorem 5.3.2 expand to a more general case as shown in formula (5.6). The performance of Gaussian matrix for $z_i \in \{\pm\mu\}$ is illustrated in Theorem 5.3.3.

Theorem 5.3.1. *Let* $\mathbf{r} = [r_1, ..., r_d]$ *randomly have* $1 \leq s \leq d$ *nonzero elements taking values* $\pm\sqrt{d/s}$ *with equal probability, and* $\mathbf{z} = [z_1, ..., z_d]$ *with elements being* $\pm\mu$ *equiprobably, where* μ *is a positive constant. Given* $f(\mathbf{r}, \mathbf{z}) = |\langle \mathbf{r}, \mathbf{z} \rangle|$, *there are three results regarding the expected value of* $f(r_i, z)$*:*

1) $\mathbb{E}(f) = 2\mu\sqrt{\frac{d}{s}}\frac{1}{2^s}\lceil\frac{s}{2}\rceil C_s^{\lceil\frac{s}{2}\rceil}$;

2) $\mathbb{E}(f)|_{s=1} = \mu\sqrt{d} > \mathbb{E}(f)|_{s>1}$;

3) $\lim_{s\to\infty} \frac{1}{\sqrt{d}}\mathbb{E}(f) \to \mu\sqrt{\frac{2}{\pi}}$.

Proof. Please see Proof 5.7.1. □

Remark on Theorem 5.3.1: This theorem discloses that the best feature selection performance is obtained, when only one feature element is sampled by each row vector. In contrast, the performance tends to converge to a lower level as the number of sampled feature elements increases. However, in practice the desired sampling process is hard to be implemented due to the few knowledge of feature location. As it will be detailed in section 5.4, what we can really implement is to sample only one feature element with high probability. Note that with the proof of this theorem, it can also be proved that if s is odd, $\mathbb{E}(f)$ fast decreases to $\mu\sqrt{2d/\pi}$ with increasing s; in contrast, if s is even, $\mathbb{E}(f)$ quickly increases towards $\mu\sqrt{2d/\pi}$ as s increases. But for arbitrary two adjacent s larger than 1, their average value on $\mathbb{E}(f)$, namely $(\mathbb{E}(f)|_s + \mathbb{E}(f)|_{s+1})/2$, is very close to $\mu\sqrt{2d/\pi}$. For clarity, the values of $\mathbb{E}(f)$ over varying s are calculated and shown in Figure 5.1, where instead of $\mathbb{E}(f)$, $\frac{1}{\mu\sqrt{d}}\mathbb{E}(f)$ is described since only the varying s is concerned. The specific character of $\mathbb{E}(f)$ ensures that one can still achieve better performance over others by sampling $s = 1$ element with a relative high probability, along with the occurrence of a sequence of s slightly larger than 1 while being even or odd equiprobably.

Theorem 5.3.2. *Let* $\mathbf{r} = [r_1, ..., r_d]$ *randomly have* $1 \le s \le d$ *nonzero elements taking values* $\pm\sqrt{d/s}$ *with equal probability, and* $\mathbf{z} = [z_1, ..., z_d]$ *with elements distributed as in formula* (5.6). *Given* $f(\mathbf{r}, \mathbf{z}) = |\langle \mathbf{r}, \mathbf{z} \rangle|$, *it is derived that:*

$$\mathbb{E}(f)|_{s=1} > \mathbb{E}(f)|_{s>1}$$

if $(\frac{9}{8})^{\frac{3}{2}}[\sqrt{\frac{2}{\pi}} + (1 + \frac{\sqrt{3}}{4})\frac{2}{\pi}(\frac{\mu}{\sigma})^{-1}] + 2\Phi(-\frac{\mu}{\sigma}) \le 1$.

Proof. Please see Proof 5.7.2. □

Remark on Theorem 5.3.2: This theorem expands Theorem 5.3.1 to a more general case where $|z_i|$ is allowed to vary in some range. In other words, there is an

upper bound on $\frac{\sigma}{\mu}$ for $\mathbb{E}(f)|_{s=1} > \mathbb{E}(f)|_{s>1}$, since $\Phi(-\frac{\mu}{\sigma})$ decreases monotonically with respect to $\frac{\mu}{\sigma}$. Clearly the larger upper bound for $\frac{\sigma}{\mu}$ allows more variation of $|z_i|$. In practice the real upper bound should be larger than that we have derived as a sufficient condition in this theorem.

Theorem 5.3.3. *Let* $\mathbf{r} = [r_1, ..., r_d]$ *have elements i.i.d drawn from* $N(0, 1)$*, and* $\mathbf{z} = [z_1, ..., z_d]$ *with elements being* $\pm\mu$ *equiprobably, where* μ *is a positive constant. Given* $f(\mathbf{r}, \mathbf{z}) = |\langle \mathbf{r}, \mathbf{z} \rangle|$*, its expected value* $\mathbb{E}(f) = \mu\sqrt{\frac{2d}{\pi}}$.

Proof. Please see Proof 5.7.3. □

Remark on Theorem 5.3.3: Comparing this theorem with Theorem 5.3.1, clearly the row vector with Gaussian distribution shares the same feature selection level with sparse row vector with a relatively large s. This explains why in practice the sparse random matrices usually can present comparable classification performance with Gaussian matrix. More importantly, it implies that the sparsest sampling process provided in Theorem 5.3.1 should outperform Gaussian matrix on feature selection.

5.4 Proposed sparse random matrix

The theorems in section 5.3 have proved that the best feature selection performance can be obtained, if only one feature element is sampled by each row vector of random matrix. It is now interesting to know if the condition above can be satisfied in the practical setting, where the high-dimensional data consists of both feature elements and redundant elements, namely $\mathbf{v} = [\mathbf{v}^f \ \mathbf{v}^r]$ and $\mathbf{w} = [\mathbf{w}^f \ \mathbf{w}^r]$. According to the theoretical condition mentioned above, it is known that the row vector $\mathbf{r} = [\mathbf{r}^f \ \mathbf{r}^r]$ can obtain the best feature selection, only when $||\mathbf{r}^f||_0 = 1$, where the quasi-norm ℓ_0 counts the number of nonzero elements in $\mathbf{r}^f$. Let $\mathbf{r}^f \in \mathbb{R}^{d_f}$, and $\mathbf{r}^r \in \mathbb{R}^{d_r}$, where $d = d_f + d_r$. Then the desired row vector should have d/d_f uniformly distributed nonzero elements such that $\mathbb{E}(||\mathbf{r}^f||_0) = 1$. However, in practice the desired distribution for row vectors is often hard to be determined, since for a real dataset the number of feature elements is usually unknown.

In this sense, we are motivated to propose a general distribution for the matrix elements, such that $||\mathbf{r}^f||_0 = 1$ holds with high probability in the setting where the feature distribution is unknown. In other words, the random matrix should hold the distribution maximizing the ratio $\Pr(||\mathbf{r}^f||_0 = 1)/\Pr(||\mathbf{r}^f||_0 \in \{2, 3, ..., d_f\})$. In practice, the desired distribution implies that the random matrix has exactly one nonzero position per column, which can be simply derived as below. Assume a random matrix $\mathbf{R} \in \mathbb{R}^{k\times d}$ randomly holding $1 \leq s \leq k$ nonzero elements per *column*[1], equivalently sd/k nonzero elements per *row*, then one can derive that

$$\begin{aligned}
&\Pr(||\mathbf{r}^f||_0 = 1)/\Pr(||\mathbf{r}^f||_0 \in \{2, 3, ..., d_f\}) \\
&= \frac{\Pr(||\mathbf{r}^f||_0 = 1)}{1 - \Pr(||\mathbf{r}^f||_0 = 0) - \Pr(||\mathbf{r}^f||_0 = 1)} \\
&= \frac{C_{d_f}^1 C_{d_r}^{sd/k-1}}{C_d^{sd/k} - C_{d_r}^{sd/k} - C_{d_f}^1 C_{d_r}^{sd/k-1}} \\
&= \frac{d_f d_r!}{\frac{d!(d_r-sd/k+1)!}{sd/k(d-sd/k)!} - \frac{d_r!(d_r-sd/k+1)}{sd/k} - d_f d_r!}
\end{aligned} \tag{5.7}$$

From the last equation in formula (5.7), it can be observed that sd/k is inversely proportional to the value of formula (5.7). Hence we have to set $s = 1$ to maximize the value. This indicates that the desired random matrix has only one nonzero element per column.

The proposed random matrix with exactly one nonzero element per column presents two obvious advantages, as detailed below.

- In complexity, the proposed matrix clearly presents much higher sparsity than existing random projection matrices. Note that, theoretically the very sparse random matrix with $q = \sqrt{d}$ [110] has higher sparsity than the proposed matrix when $k < \sqrt{d}$. However, in practice the case $k < \sqrt{d}$ is usually not of practical interest, due to the weak performance caused by large compression rate d/k ($> \sqrt{d}$).
- In performance, it can be derived that the proposed matrix outperforms other more dense matrices, if the projection dimension k is not much smaller than the

1. Note that in the former section the term s is used to represent the number of nonzero elements per *row*.

number d_f of feature elements included in the high-dimensional vector. To be specific, with Figure 5.1, it can be derived that the dense matrices with column weight $s > 1$ share comparable feature selection performance, because as s increases they tend to sample more than one feature element (namely $||\mathbf{r}^f||_0 > 1$) with higher probability. Then the proposed matrix with $s = 1$ will present better performance than them, if k ensures $||\mathbf{r}^f||_0 = 1$ with high probability, or equivalently the ratio $\Pr(||\mathbf{r}^f||_0 = 1)/\Pr(||\mathbf{r}^f||_0 \in \{2, 3, ..., d_f\})$ being relatively large. As shown in formula (5.7), the condition above can be better satisfied as k increases. Inversely, as k decreases, the feature selection advantage of the proposed matrix will degrade. Recall that the proposed matrix is weaker than other more dense matrices on distance preservation, as demonstrated in section 5.2.2. This means that the proposed matrix will perform worse than others when its feature selection advantage is not obvious. In other words, there should exist a lower bound for k to ensure the performance advantage of the proposed matrix, which is also verified in the following experiments. It can be roughly estimated that lower bound of k should be on the order of d_f, since for the proposed matrix with column weight $s = 1$, the $k = d_f$ leads to $\mathbb{E}(||\mathbf{r}^f||_0) = d/k \times d_f/d = 1$. In practice, the performance advantage seemingly can be maintained for a relatively small $k(< d_f)$. For instance, in the following experiments on synthetic data, the lower bound of k is as small as $d_f/10$. This recalls the fact that to obtain performance advantage, we merely require $\Pr(||\mathbf{r}^f||_0 = 1)$ being relatively large rather than being equal to 1, as detailed in the remark on Theorem 5.3.1.

5.5 Experiments

5.5.1 Setup

This section verifies the feature selection advantage of the proposed currently most sparse matrix (MSM) over other popular matrices, by conducting binary classification on both synthetic data and real data. Here the synthetic data with labeled

feature elements is provided to specially observe the relation between the projection dimension and feature number, as well as the impact of redundant elements. The real data involve three typical datasets in the area of dimensionality reduction: face image, DNA microarray and text document. As for the binary classifier, the classical support vector machine (SVM) based on Euclidean distance is adopted. For comparison, we test three popular random matrices: Gaussian random matrix (GM), sparse random matrix (SM) as in formula (5.3) with $q = 3$ [8] and very sparse random matrix (VSM) with $q = \sqrt{d}$ [110].

The simulation parameters are introduced as follows. It is known that the repeated random projection tends to improve the feature selection, so here each classification decision is voted by performing 5 times random projection [115]. The correct classification rate at each projection dimension k is derived with 100000 simulation runs. In each simulation, four matrices are tested with the same samples. The projection dimension k decreases uniformly from the high dimension d. Moreover, it is necessary to note that, for some datasets containing more than two classes of samples, the SVM classifier randomly selects two classes to conduct binary classification in each simulation. For each class of data, one half of samples are randomly selected for training, and the rest for testing.

5.5.2 Synthetic data

The synthetic data is designed for evaluating two factors below:

– the relation between the lower bound of projection dimension k and the feature dimension d_f

– the negative influence of redundant elements, which are ideally assumed to be zero in the previous theoretical proofs.

Then two classes of synthetic data with d_f feature elements and $d - d_f$ redundant elements are generated in two steps:

1) randomly build a vector $\tilde{\mathbf{v}} \in \{\pm 1\}^d$, then define a vector $\tilde{\mathbf{w}}$ distributed as $\tilde{w}_i = -\tilde{v}_i$, if $1 \leq i \leq d_f$, and $\tilde{w}_i = \tilde{v}_i$, if $d_f < i \leq d$;

2) generate two classes of datasets $\mathcal{V}$ and $\mathcal{W}$ by i.i.d sampling $v_i^f \in N(\tilde{v}_i, \sigma_f^2)$ and $w_i^f \in N(\tilde{w}_i, \sigma_f^2)$, if $1 \leq i \leq d_f$; and $v_i^r \in N(\tilde{v}_i, \sigma_r^2)$ and $w_i^r \in N(\tilde{w}_i, \sigma_r^2)$, if

$d_f < i \leq d$.

Subsequently, the distributions on pointwise distance can be approximately derived as $|v_i^f - w_i^f| \in N(2, 2\sigma_f^2)$ for feature elements and $(v_i^r - w_i^r) \in N(0, 2\sigma_r^2)$ for redundant elements, respectively. To be close to reality, we introduce some unreliability for feature elements and redundant elements by adopting relatively large variances. Precisely, in the simulation σ_f is fixed to 8 and σ_r varies in the set $\{8, 12, 16\}$. Note that, the probability of $(v_i^r - w_i^r)$ converging to zero will decrease as σ_r increases. Thus the increasing σ_r will be a challenge for our previous theoretical conjecture derived on the assumption of $(v_i^r - w_i^r) = 0$. As for the size of the dataset, the data dimension d is set to 2000, and the feature dimension $d_f = 1000$. Each dataset consists of 100 randomly generated samples.

Table 5.1 shows the correct classification performance of four types of matrices over evenly varying projection dimension k. The results provide two positive clues. First, the proposed matrix preserves obvious advantage over others, even when k is relatively small, for instance, k/d_f is allowed to be small as 1/10 when $\sigma_r = 8$. Second, with the interference of redundant elements, the proposed matrix still outperforms others, which implies that the previous theoretical result is also applicable to the real case where the redundant elements cannot be simply neglected. Obviously the synthetic simulation is far from being enough due to the simple assumption on data distribution. Hence we will have to perform a more wide test on real data in next subsection.

5.5.3 Real data

Datasets

1) Face image
   - AR [116] : as in [117], a subset of 2600 frontal faces from 50 males and 50 females are examined. For some persons, the faces were taken at different times, varying the lighting, facial expressions (open/closed eyes, smiling/not smiling) and facial details (glasses/no glasses). There are 6 faces with dark glasses and 6 faces partially disguised by scarfs among 26 faces per person.

Table 5.1: Correct classification rates on the synthetic data which have $d = 2000$ and redundant elements suffering from three different varying levels σ_r. The best performance is highlighted in bold. Recall that the acronyms GM, SM, VSM and MSM represent Gaussian random matrix, sparse random matrix with $q = 3$, very sparse rand matrix with $q = \sqrt{d}$, and the proposed most sparse random matrix, respectively.

	k	50	100	200	400	600	800	1000	1500	2000
$\sigma_r = 8$	GM	70.44	67.93	84.23	93.31	95.93	97.17	97.71	98.35	98.74
	SM	**70.65**	67.90	84.43	93.03	95.97	96.86	97.78	98.36	98.80
	VSM	70.55	68.05	84.46	93.19	96.00	96.99	97.68	98.38	98.76
	MSM	70.27	**68.09**	**84.66**	**94.22**	**97.11**	**98.03**	**98.67**	**99.37**	**99.57**
$\sigma_r = 12$	GM	**64.89**	**63.06**	76.08	85.04	88.46	90.21	91.16	92.68	93.32
	SM	64.67	62.66	75.85	85.03	88.30	90.09	91.21	92.70	93.30
	VSM	65.17	62.95	76.12	85.14	88.80	90.46	91.37	92.88	93.64
	MSM	64.85	63.00	**76.82**	**88.41**	**93.51**	**96.12**	**97.59**	**99.13**	**99.68**
$\sigma_r = 16$	GM	60.90	59.42	70.13	78.26	81.70	83.82	84.74	86.50	87.49
	SM	60.86	59.58	69.93	78.04	81.66	83.85	84.79	86.55	87.39
	VSM	60.98	**59.87**	70.27	78.49	81.98	84.36	85.27	86.98	87.81
	MSM	**61.09**	59.29	**71.58**	**84.56**	**91.65**	**95.50**	**97.24**	**98.91**	**99.30**

- Extended Yale B [118, 119]: this dataset includes about 2414 frontal faces of 38 persons, which suffer varying illumination changes.
- FERET [120]: this dataset consists of more than 10000 faces from more than 1000 persons taken in largely varying circumstances. The database is further divided into several sets which are formed for different evaluations. Here we evaluate the 1984 *frontal* faces of 992 persons each with 2 faces separately extracted from sets *fa* and *fb*.
- GTF [121]: in this dataset, 750 images from 50 persons were captured at different scales and orientations under variations in illumination and expression. So cropped faces suffer from serious pose variation.
- ORL [122]: it contains 40 persons each with 10 faces. Besides slightly varying lighting and expressions, the faces also undergo slight changes on pose.

2) DNA microarray

- Colon [123]: this is a dataset consisting of 40 colon tumors and 22 normal colon tissue samples. 2000 genes with highest intensity across the samples are considered.
- ALML [124]: this dataset contains 25 samples taken from patients suffering from acute myeloid leukemia (AML) and 47 samples from patients suffering

Table 5.2: Correct classification rates on five face datasets with dimension $d = 1200$. For each projection dimension k, the best performance is highlighted in bold. Recall that the acronyms GM, SM, VSM and MSM represent Gaussian random matrix, sparse random matrix with $q = 3$, very sparse random matrix with $q = \sqrt{d}$, and the proposed most sparse random matrix, respectively.

	k	30	60	120	240	360	480	600
AR	GM	**98.67**	99.04	99.19	99.24	99.30	99.28	99.33
	SM	98.58	99.04	99.21	99.25	99.31	99.30	99.32
	VSM	98.62	99.07	99.20	99.27	99.30	99.31	99.34
	MSM	98.64	**99.10**	**99.24**	**99.35**	**99.48**	**99.50**	**99.58**
Ext-YaleB	GM	97.10	**98.06**	98.39	98.49	98.48	98.45	98.47
	SM	97.00	98.05	98.37	98.49	98.48	98.45	98.47
	VSM	97.12	98.05	98.36	98.50	98.48	98.45	98.48
	MSM	**97.15**	**98.06**	**98.40**	**98.54**	**98.54**	**98.57**	**98.59**
FERET	GM	86.06	86.42	86.31	86.50	86.46	86.66	86.57
	SM	86.51	86.66	87.26	88.01	88.57	89.59	90.13
	VSM	**87.21**	87.61	89.34	91.14	92.31	93.75	93.81
	MSM	87.11	**88.74**	**92.04**	**95.38**	**96.90**	**97.47**	**97.47**
GTF	GM	96.67	97.48	97.84	98.06	98.09	98.10	98.16
	SM	96.63	97.52	97.85	98.06	98.09	98.13	98.16
	VSM	**96.69**	**97.57**	97.87	98.10	98.13	98.14	98.16
	MSM	96.65	97.51	**97.94**	**98.25**	**98.40**	**98.43**	**98.53**
ORL	GM	94.58	95.69	96.31	96.40	96.54	96.51	96.49
	SM	94.50	95.63	96.36	96.38	96.48	96.47	96.48
	VSM	94.60	**95.77**	96.33	96.35	96.53	96.55	96.46
	MSM	**94.64**	95.75	**96.43**	**96.68**	**96.90**	**97.04**	**97.05**

from acute lymphoblastic leukemia (ALL). Each sample is expressed with 7129 genes.

– Lung [125] : this dataset contains 86 lung tumor and 10 normal lung samples. Each sample holds 7129 genes.

3) Text document [126] [2]

– TDT2: the recently modified dataset includes 96 categories of total 10212 documents/samples. Each document is represented with vector of length 36771. Here we adopt the first 19 categories each with more than 100 documents, such that each category is tested with 100 randomly selected documents.

– 20Newsgroups(version 1): there are 20 categories of 18774 documents in this dataset. Each document has vector dimension 61188. Since the documents are not equally distributed in the 20 categories, we randomly select 600

2. Publicly available at `http://www.cad.zju.edu.cn/home/dengcai/Data/TextData.html`

documents for each category, which is nearly the maximum number we can assign to all categories.

– RCV1: the original dataset contains 9625 documents each with 29992 distinct words, corresponding to 4 categories with 2022, 2064, 2901, and 2638 documents respectively. To reduce computation, we randomly select only 1000 documents for each category.

Table 5.3: Correct classification rates on three DNA datasets with dimension $d = 2000$. For each projection dimension k, the best performance is highlighted in bold. Recall that the acronyms GM, SM, VSM and MSM represent Gaussian random matrix, sparse random matrix with $q = 3$, very sparse random matrix with $q = \sqrt{d}$, and the proposed most sparse random matrix, respectively.

	k	50	100	200	400	600	800	1000	1500
Colon	GM	77.16	77.15	77.29	77.28	77.46	77.40	77.35	77.55
	SM	**77.23**	77.18	77.16	77.36	77.42	77.42	77.39	77.54
	VSM	76.86	77.19	77.34	77.52	77.64	77.61	77.61	77.82
	MSM	76.93	**77.34**	**77.73**	**78.22**	**78.51**	**78.67**	**78.65**	**78.84**
ALML	GM	**65.11**	66.22	66.96	67.21	67.23	67.24	67.28	67.37
	SM	65.09	66.16	66.93	67.25	67.22	67.31	67.31	67.36
	VSM	64.93	67.32	68.52	69.01	69.15	69.16	69.25	69.33
	MSM	65.07	**68.38**	**70.43**	**71.39**	**71.75**	**71.87**	**72.00**	**72.11**
Lung	GM	98.74	98.80	98.91	98.96	98.95	98.96	98.95	98.97
	SM	98.71	98.80	98.92	98.97	98.96	98.98	98.97	98.97
	VSM	**98.81**	99.21	99.48	99.57	99.58	99.61	99.61	99.61
	MSM	98.70	**99.48**	**99.69**	**99.70**	**99.69**	**99.72**	**99.68**	**99.65**

Results

Three types of representative high-dimensional datasets are tested for random projection over evenly varying projection dimension k. The datasets are first briefly introduced, and then the results are illustrated and analyzed. Note that, the simulation is developed to compare the feature selection performance of different random projections, rather than to obtain the best performance. So to reduce the simulation load, the original high-dimensional data is uniformly downsampled to a relatively low dimension. Precisely, the face image, DNA, and text are reduced to the dimensions 1200, 2000 and 3000, respectively. Note that, in terms of JL lemma, the original high dimension is allowed to be reduced to arbitrary values (not limited to 1200, 2000 or 3000), since theoretically the distance preservation of random projection is

Table 5.4: Correct classification rates on three Text datasets with dimension $d = 3000$. For each projection dimension k, the best performance is highlighted in bold. Recall that the acronyms GM, SM, VSM and MSM represent Gaussian random matrix, sparse random matrix with $q = 3$, very sparse random matrix with $q = \sqrt{d}$, and the proposed most sparse random matrix, respectively.

	k	150	300	600	900	1200	1500	2000
TDT2	GM	**83.64**	83.10	82.84	82.29	81.94	81.67	81.72
	SM	83.61	82.93	83.10	82.28	81.92	81.55	81.76
	VSM	82.59	82.55	82.72	82.2	81.74	81.47	81.78
	MSM	82.52	**83.15**	**84.06**	**83.58**	**83.42**	**82.95**	**83.35**
Newsgroup	GM	**75.35**	**74.46**	72.27	71.52	71.34	70.63	69.95
	SM	75.21	74.43	72.29	71.30	71.07	70.34	69.58
	VSM	74.84	73.47	70.22	69.21	69.28	68.28	68.04
	MSM	74.94	74.20	**72.34**	**71.54**	**71.53**	**70.46**	**70.00**
RCV1	GM	85.85	86.20	81.65	78.98	78.22	78.21	78.21
	SM	**86.05**	86.19	81.53	79.08	78.23	78.14	78.19
	VSM	86.04	86.14	81.54	78.57	78.12	78.05	78.04
	MSM	85.75	**86.33**	**85.09**	**83.38**	**82.30**	**81.39**	**80.69**

independent of the size of high-dimensional data [8].

Tables 5.2, 5.3 and 5.4 illustrate the classification performance of four classes of matrices on three typical high-dimensional data: face image, DNA microarray and text document. It can be observed that, all results are consistent with the theoretical conjecture stated in section 5.4, that the proposed matrix should outperform other matrices as k increases. This verifies that the theoretically proposed matrix is indeed applicable to the typical high-dimensional data. Note that, although the theoretical result concerning best feature selection is derived with a general assumption on the distribution of feature difference, it cannot be ensured to be followed by all possible real data. In this sense, the proposed matrix needs more test to expand to more areas of application.

5.6 Conclusion

This chapter has shown that random projection can achieve its best feature selection performance, when only one feature element of high-dimensional data is considered at each sampling. However, in practice the number of feature elements is usually unknown, and so the aforementioned best sampling process is hard to be implemented. To achieve the best sampling process with high probability, we

practically propose a class of sparse random matrices with exactly one nonzero element per column. The proposed matrix shows better performance than other more dense matrices on the classification experiments based on face images, DNA microarray and text document. Therefore, it can be argued that the proposed random projection is competitive on both complexity and performance.

5.7 Proof

5.7.1 Proof of Theorem 5.3.1

Proof. Due to the sparsity of $\mathbf{r}$ and the symmetric property of both r_j and z_j, the function $f(\mathbf{r}, \mathbf{z})$ can be equivalently transformed to a simpler form, that is $f(x) = \mu\sqrt{\frac{d}{s}}|\sum_{i=1}^{i=s} x_i|$ with x_i being ± 1 equiprobably. With the simplified form, three results of this theorem are sequentially proved below.

1) First, it can be easily derived that

$$\mathbb{E}(f(x)) = \mu\sqrt{\frac{d}{s}}\frac{1}{2^s}\sum_{i=1}^{s}(C_s^i|s-2i|)$$

then the solution to $\mathbb{E}(f(x))$ turns to calculating $\sum_{i=1}^{s}(C_s^i|s-2i|)$, which can be deduced as

$$\sum_{i=1}^{s}(C_s^i|s-2i|) = \begin{cases} 2sC_{s-1}^{\frac{s}{2}-1} & if\ s\ is\ even \\ 2sC_{s-1}^{\frac{s-1}{2}} & if\ s\ is\ odd \end{cases}$$

by summing the piecewise function

$$C_s^i|s-2i| = \begin{cases} sC_{s-1}^0 & if\ i=0 \\ sC_{s-1}^{s-i-1} - sC_{s-1}^{i-1} & if\ 1 \le i \le \frac{s}{2} \\ sC_{s-1}^{i-1} - sC_{s-1}^{s-i-1} & if\ \frac{s}{2} < i < s \\ sC_{s-1}^{s-1} & if\ i=s \end{cases}$$

Further, with $C_{s-1}^{i-1} = \frac{i}{s}C_s^i$, it can be deduced that

$$\sum_{i=1}^{s}(C_s^i|s-2i|) = 2\lceil\frac{s}{2}\rceil C_s^{\lceil\frac{s}{2}\rceil}$$

Then the fist result is obtained as

$$\mathbb{E}(f) = 2\mu\sqrt{\frac{d}{s}}\frac{1}{2^s}\lceil\frac{s}{2}\rceil C_s^{\lceil\frac{s}{2}\rceil}$$

2) Following the proof above, it is clear that $\mathbb{E}(f(x))|_{s=1} = f(x)|_{s=1} = \mu\sqrt{d}$. As for $\mathbb{E}(f(x))|_{s>1}$, it is evaluated under two cases:

– if s is odd,

$$\frac{\mathbb{E}(f(x))|_s}{\mathbb{E}(f(x))|_{s-2}} = \frac{\frac{2}{\sqrt{s}}\frac{1}{2^s}\frac{s+1}{2}C_s^{\frac{s+1}{2}}}{\frac{2}{\sqrt{s-2}}\frac{1}{2^{s-2}}\frac{s-1}{2}C_{s-2}^{\frac{s-1}{2}}} = \frac{\sqrt{s(s-2)}}{s-1} < 1$$

namely, $\mathbb{E}(f(x))$ decreases monotonically with respect to s. Clearly, in this case $\mathbb{E}(f(x))|_{s=1} > \mathbb{E}(f(x))|_{s>1}$;

– if s is even,

$$\frac{\mathbb{E}(f(x))|_s}{\mathbb{E}(f(x))|_{s-1}} = \frac{\frac{2}{\sqrt{s}}\frac{1}{2^s}\frac{s}{2}C_s^{\frac{s}{2}}}{\frac{2}{\sqrt{s-1}}\frac{1}{2^{s-1}}\frac{s}{2}C_{s-1}^{\frac{s}{2}}} = \sqrt{\frac{s-1}{s}} < 1$$

which means $\mathbb{E}(f(x))|_{s=1} > \mathbb{E}(f(x))|_{s>1}$, since $s-1$ is odd number for which $\mathbb{E}(f(x))$ monotonically decreases.

Therefore the proof of the second result is completed.

3) The proof of the third result is developed by employing Stirling's approximation [127]

$$s! = \sqrt{2\pi s}(\frac{s}{e})^s e^{\lambda_s}, \quad 1/(12s+1) < \lambda_s < 1/(12s).$$

Precisely, with the formula of $\mathbb{E}(f(x))$, it can be deduced that

– if s is even,

$$\mathbb{E}(f(x)) = \mu\sqrt{ds}\frac{1}{2^s}\frac{s!}{\frac{s}{2}!\frac{s}{2}!} = \mu\sqrt{\frac{2d}{\pi}}e^{\lambda_s - 2\lambda_{\frac{s}{2}}}$$

– if s is odd,

$$\mathbb{E}(f(x)) = \mu\sqrt{d}\frac{s+1}{\sqrt{s}}\frac{1}{2^s}\frac{s!}{\frac{s+1}{2}!\frac{s-1}{2}!} = \mu\sqrt{\frac{2d}{\pi}}(\frac{s^2}{s^2-1})^{\frac{s}{2}}e^{\lambda_s-\lambda_{\frac{s+1}{2}}-\lambda_{\frac{s-1}{2}}}$$

Clearly $\lim_{s\to\infty}\frac{1}{\sqrt{d}}\mathbb{E}(f(x)) \to \mu\sqrt{\frac{2}{\pi}}$ holds, whenever s is even or odd.

□

5.7.2 Proof of Theorem 5.3.2

Proof. Due to the sparsity of $\mathbf{r}$ and the symmetric property of both r_j and z_j, it is easy to derive that $f(\mathbf{r},\mathbf{z}) = |\langle \mathbf{r},\mathbf{z}\rangle| = \sqrt{\frac{d}{s}}|\sum_{j=1}^{s} z_j|$. This simplified formula will be studied in the following proof. To present a readable proof, here we review the distribution shown in formula (5.6)

$$z_j \sim \begin{cases} N(\mu,\sigma) & \text{with probability } 1/2 \\ N(-\mu,\sigma) & \text{with probability } 1/2 \end{cases}$$

where for $x \in N(\mu,\sigma)$, $\Pr(x>0) = 1-\epsilon$, $\epsilon = \Phi(-\frac{\mu}{\sigma})$ is a tiny positive number. For notational simplicity, the subscript of random variable z_j is dropped in the following proof. To ease the proof of the theorem, we first need to derive the expected value of $|x|$ with $x \sim N(\mu,\sigma^2)$:

$$\begin{aligned}
\mathbb{E}(|x|) &= \int_{-\infty}^{\infty}\frac{|x|}{\sqrt{2\pi}\sigma}e^{\frac{-(x-\mu)^2}{2\sigma^2}}dx \\
&= \int_{-\infty}^{0}\frac{-x}{\sqrt{2\pi}\sigma}e^{\frac{-(x-\mu)^2}{2\sigma^2}}dx + \int_{0}^{\infty}\frac{x}{\sqrt{2\pi}\sigma}e^{\frac{-(x-\mu)^2}{2\sigma^2}}dx \\
&= -\int_{-\infty}^{0}\frac{x-\mu}{\sqrt{2\pi}\sigma}e^{\frac{-(x-\mu)^2}{2\sigma^2}}dx + \int_{0}^{\infty}\frac{x-\mu}{\sqrt{2\pi}\sigma}e^{\frac{-(x-\mu)^2}{2\sigma^2}}dx \\
&+ \mu\int_{0}^{\infty}\frac{1}{\sqrt{2\pi}\sigma}e^{\frac{-(x-\mu)^2}{2\sigma^2}}dx - \mu\int_{-\infty}^{0}\frac{1}{\sqrt{2\pi}\sigma}e^{\frac{-(x-\mu)^2}{2\sigma^2}}dx \\
&= \frac{\sigma}{\sqrt{2\pi}}e^{-\frac{(x-\mu)^2}{2\sigma^2}}|_{-\infty}^{0} - \frac{\sigma}{\sqrt{2\pi}}e^{-\frac{(x-\mu)^2}{2\sigma^2}}|_{0}^{\infty} + \mu\Pr(x>0) - \mu\Pr(x<0) \\
&= \sqrt{\frac{2}{\pi}}\sigma e^{-\frac{\mu^2}{2\sigma^2}} + \mu(1-2\Pr(x<0)) \\
&= \sqrt{\frac{2}{\pi}}\sigma e^{-\frac{\mu^2}{2\sigma^2}} + \mu(1-2\Phi(-\frac{\mu}{\sigma}))
\end{aligned}$$

which will be used many a time in the following proof. Then we are ready to prove the theorem below.

1) This part presents the expected value of $f(r_i, z)$ for the cases $s = 1$ and $s > 1$.
   - if $s = 1$, $f(\mathbf{r}, \mathbf{z}) = \sqrt{d}|z|$; with the the probability density function of z:

$$p(z) = \frac{1}{2}\frac{1}{\sqrt{2\pi}\sigma}e^{\frac{-(z-\mu)^2}{2\sigma^2}} + \frac{1}{2}\frac{1}{\sqrt{2\pi}\sigma}e^{\frac{-(z+\mu)^2}{2\sigma^2}}$$

   one can derive that

$$\begin{aligned}\mathbb{E}(|z|) &= \int_{-\infty}^{\infty}|z|p(z)d_z \\ &= \frac{1}{2}\int_{-\infty}^{\infty}\frac{|z|}{\sqrt{2\pi}\sigma}e^{\frac{-(z-\mu)^2}{2\sigma^2}}dz + \frac{1}{2}\int_{-\infty}^{\infty}\frac{|z|}{\sqrt{2\pi}\sigma}e^{\frac{-(z+\mu)^2}{2\sigma^2}}dz\end{aligned}$$

   with the previous result on $\mathbb{E}(|x|)$, it is further deduced that

$$\mathbb{E}(|z|) = \sqrt{\frac{2}{\pi}}\sigma e^{-\frac{\mu^2}{2\sigma^2}} + \mu(1 - 2\Phi(-\frac{\mu}{\sigma}))$$

   Recall that $\Phi(-\frac{\mu}{\sigma}) = \epsilon$, so

$$\mathbb{E}(f) = \sqrt{d}\mathbb{E}(|z|) = \sqrt{\frac{2d}{\pi}}\sigma_\mu e^{-\frac{\mu^2}{2\sigma^2}} + \mu\sqrt{d}(1 - 2\Phi(-\frac{\mu}{\sigma})) \approx \mu\sqrt{d}$$

   if ϵ is tiny enough as illustrated in formula (5.6).
   - if $s > 1$, $f(\mathbf{r}, \mathbf{z}) = \sqrt{\frac{d}{s}}|\sum_{j=1}^{s} z|$; let $t = \sum_{j=1}^{s} z$, then according to the symmetric distribution of z, t holds $s + 1$ different distributions:

$$t \sim N((s - 2i)\mu, s\sigma^2) \text{ with probability } \frac{1}{2^s}C_s^i$$

   where $0 \leq i \leq s$ denotes the number of z drawn from $N(-\mu, \sigma^2)$. Then the PDF of t can be described as

$$p(t) = \frac{1}{2^s}\sum_{i=0}^{s} C_s^i \frac{1}{\sqrt{2\pi s}\sigma}e^{\frac{-(t-(s-2i)\mu)^2}{2s\sigma^2}}$$

then,

$$\begin{aligned}\mathbb{E}(|t|) &= \int_{-\infty}^{\infty} |t| p(t) dt \\ &= \frac{1}{2^s} \sum_{i=0}^{s} C_s^i \int_{-\infty}^{\infty} |t| \frac{1}{\sqrt{2\pi s}\sigma} e^{\frac{-(t-(s-2i)\mu)^2}{2s\sigma^2}} dt \\ &= \frac{1}{2^s} \sum_{i=0}^{s} C_s^i \{ \sqrt{\frac{2s}{\pi}} \sigma e^{\frac{-(s-2i)^2\mu^2}{2s\sigma^2}} + \mu|s-2i|[1-2\Phi(\frac{-|s-2i|\mu}{\sqrt{s}\sigma})] \}\end{aligned}$$

subsequently, the expected value of $f(r_i, z)$ can be expressed as

$$\begin{aligned}\mathbb{E}(f) = \mu\sqrt{\frac{d}{s}} \frac{1}{2^s} \sum_{i=0}^{s} (C_s^i |s-2i|) + \sigma\sqrt{\frac{2d}{\pi}} \frac{1}{2^s} \sum_{i=0}^{s} C_s^i e^{\frac{-(s-2i)^2\mu^2}{2s\sigma^2}} \\ - 2\mu\sqrt{\frac{d}{s}} \frac{1}{2^s} \sum_{i=0}^{s} [C_s^i |s-2i| \Phi(\frac{-|s-2i|\mu}{\sqrt{s}\sigma})]\end{aligned}$$

2) This part derives the upper bound of the aforementioned $\mathbb{E}(f)|_{s>1}$. For simpler expression, the three factors of above expression for $\mathbb{E}(f)|_{s>1}$ are sequentially represented by f_1, f_2 and f_3, and then are analyzed, respectively.

– for $f_1 = \mu\sqrt{\frac{d}{s}} \frac{1}{2^s} \sum_{i=0}^{s} (C_s^i |s-2i|)$, it can be rewritten as

$$f_1 = 2\mu\sqrt{\frac{d}{s}} \frac{1}{2^s} C_s^{\lceil \frac{s}{2} \rceil} \lceil \frac{s}{2} \rceil$$

– for $f_2 = \sigma\sqrt{\frac{2d}{\pi}} \frac{1}{2^s} \sum_{i=0}^{s} C_s^i e^{\frac{-(s-2i)^2\mu^2}{2s\sigma^2}}$, first, we can bound

$$\begin{cases} e^{\frac{-(s-2i)^2\mu^2}{2s\sigma^2}} < \exp(-\frac{\mu^2}{\sigma^2}) & \text{if } i < \alpha \text{ or } i > \alpha \\ e^{\frac{-(s-2i)^2\mu^2}{2s\sigma^2}} \leq 1 & \text{if } \alpha \leq i \leq s-\alpha \end{cases}$$

where $\alpha = \lceil \frac{s-\sqrt{s}}{2} \rceil$. Take it into f_2,

$$\begin{aligned} f_2 &< \sigma\sqrt{\frac{2d}{\pi}} \frac{1}{2^s} \sum_{i=0}^{\alpha-1} C_s^i e^{\frac{-\mu^2}{\sigma^2}} + \sigma\sqrt{\frac{2d}{\pi}} \frac{1}{2^s} \sum_{i=s-\alpha+1}^{s} C_s^i e^{\frac{-\mu^2}{2\sigma^2}} + \sigma\sqrt{\frac{2d}{\pi}} \frac{1}{2^s} \sum_{i=\alpha}^{s-\alpha} C_s^i \\ &< \sigma\sqrt{\frac{2d}{\pi}} e^{\frac{-\mu^2}{\sigma^2}} + \sigma\sqrt{\frac{2d}{\pi}} \frac{1}{2^s} \sum_{i=\alpha}^{s-\alpha} C_s^i \end{aligned}$$

Since $C_s^i \leq C_s^{\lceil s/2 \rceil}$,

$$\begin{aligned} f_2 &< \sigma\sqrt{\frac{2d}{\pi}}e^{\frac{-\mu^2}{\sigma^2}} + \sigma\sqrt{\frac{2d}{\pi}}\frac{1}{2^s}(\lfloor\sqrt{s}\rfloor + 1)C_s^{\lceil s/2 \rceil} \\ &\leq \sigma\sqrt{\frac{2d}{\pi}}e^{\frac{-\mu^2}{\sigma^2}} + \sigma\sqrt{\frac{2d}{\pi}}\frac{1}{2^s}\sqrt{s}C_s^{\lceil s/2 \rceil} + \sigma\sqrt{\frac{2d}{\pi}}\frac{1}{2^s}C_s^{\lceil s/2 \rceil} \\ &\leq \sigma\sqrt{\frac{2d}{\pi}}e^{\frac{-\mu^2}{\sigma^2}} + \sigma\sqrt{\frac{2d}{\pi}}\frac{1}{2^s}\frac{2}{\sqrt{s}}C_s^{\lceil s/2 \rceil}\lceil\frac{s}{2}\rceil + \sigma\sqrt{\frac{2d}{\pi}}\frac{1}{2^s}C_s^{\lceil s/2 \rceil} \end{aligned}$$

with Stirling's approximation,

$$f_2 < \begin{cases} \sqrt{\frac{2d}{\pi}}\sigma e^{\frac{-\mu^2}{2\sigma^2}} + \sqrt{d}\frac{2}{\pi}\sigma e^{\lambda_s - 2\lambda_{s/2}} + \sqrt{\frac{d}{s}}\frac{2}{\pi}\sigma e^{\lambda_s - 2\lambda_{s/2}} & \text{if } s \text{ is even} \\ \sqrt{\frac{2d}{\pi}}\sigma e^{\frac{-\mu^2}{2\sigma^2}} + \sqrt{d}\frac{2\sigma}{\pi}(\frac{s^2}{s^2-1})^{\frac{s}{2}}e^{\lambda_s - \lambda_{\frac{s+1}{2}} - \lambda_{\frac{s-1}{2}}} & \\ +\sqrt{d}\frac{2\sigma}{\pi}\frac{\sqrt{s}}{s+1}(\frac{s^2}{s^2-1})^{\frac{s}{2}}e^{\lambda_s - \lambda_{\frac{s+1}{2}} - \lambda_{\frac{s-1}{2}}} & \text{if } s \text{ is odd} \end{cases}$$

– for $f_3 = -2\mu\sqrt{\frac{d}{s}}\frac{1}{2^s}\sum_{i=0}^{s}[C_s^i|s-2i|\Phi(\frac{-|s-2i|\mu}{\sqrt{s}\sigma})]$, with the previous defined α,

$$\begin{aligned} f_3 &\leq -2\mu\sqrt{\frac{d}{s}}\frac{1}{2^s}\sum_{i=\alpha}^{s-\alpha}[C_s^i|s-2i|\Phi(\frac{-|s-2i|\mu}{\sqrt{s}\sigma})] \\ &\leq -2\mu\sqrt{\frac{d}{s}}\frac{1}{2^s}\sum_{i=\alpha}^{s-\alpha}[C_s^i|s-2i|\Phi(\frac{-\mu}{\sigma})] \\ &= -2\mu\epsilon\sqrt{\frac{d}{s}}\frac{1}{2^s}\sum_{i=\alpha}^{s-\alpha}[C_s^i|s-2i|] \\ &= -2\mu\epsilon\sqrt{\frac{d}{s}}\frac{1}{2^s}(2sC_{s-1}^{\lceil\frac{s}{2}-1\rceil} - 2sC_{s-1}^{\alpha-1}) \\ &= -4\mu\epsilon\sqrt{ds}\frac{1}{2^s}(C_{s-1}^{\lceil\frac{s}{2}-1\rceil} - C_{s-1}^{\alpha-1}) \\ &\leq 0 \end{aligned}$$

finally, we can further deduce that

$$\mathbb{E}(f)|_{s>1} = f_1 + f_2 + f_3$$

$$< \begin{cases} 2\mu\frac{1}{2^s}\sqrt{\frac{d}{s}}C_s^{\lceil\frac{s}{2}\rceil} + \frac{2\sigma}{\pi}\sqrt{d}e^{\lambda_s - 2\lambda_{\frac{s}{2}}} + \sqrt{\frac{2d}{\pi}}\sigma e^{\frac{-\mu^2}{2\sigma^2}} + \sqrt{\frac{d}{s}}\frac{2}{\pi}\sigma e^{\lambda_s - 2\lambda_{s/2}} \\ -4\mu\epsilon\sqrt{ds}\frac{1}{2^s}(C_{s-1}^{\lceil\frac{s}{2}-1\rceil} - C_{s-1}^{\alpha-1}) & \text{if } s \text{ is even} \\ \\ 2\mu\frac{1}{2^s}\sqrt{\frac{d}{s}}C_s^{\lceil\frac{s}{2}\rceil} + \frac{2\sigma}{\pi}\sqrt{d}\frac{s^2}{s^2-1}^{\frac{s}{2}}e^{\lambda_s - \lambda_{\frac{s+1}{2}} - \lambda_{\frac{s-1}{2}}} + \sqrt{\frac{2d}{\pi}}\sigma e^{\frac{-\mu^2}{2\sigma^2}} \\ +\sqrt{d}\frac{2\sigma}{\pi}\frac{\sqrt{s}}{s+1}(\frac{s^2}{s^2-1})^{\frac{s}{2}}e^{\lambda_s - \lambda_{\frac{s+1}{2}} - \lambda_{\frac{s-1}{2}}} - 4\mu\epsilon\sqrt{ds}\frac{1}{2^s}(C_{s-1}^{\lceil\frac{s}{2}-1\rceil} - C_{s-1}^{\alpha-1}) & \text{if } s \text{ is odd} \end{cases}$$

$$= \begin{cases} (\sqrt{\frac{2d}{\pi}}\mu + \frac{4\sigma}{\pi}\sqrt{d})e^{\lambda_s - 2\lambda_{\frac{s}{2}}} + \sqrt{\frac{2d}{\pi}}\sigma e^{\frac{-\mu^2}{2\sigma^2}} + \sqrt{\frac{d}{s}}\frac{2}{\pi}\sigma e^{\lambda_s - 2\lambda_{s/2}} \\ -4\mu\epsilon\sqrt{ds}\frac{1}{2^s}(C_{s-1}^{\lceil\frac{s}{2}-1\rceil} - C_{s-1}^{\alpha-1}) & \text{if } s \text{ is even} \\ \\ (\sqrt{\frac{2d}{\pi}}\mu + \frac{4\sigma}{\pi}\sqrt{d})(\frac{s^2}{s^2-1})^{\frac{s}{2}}e^{\lambda_s - \lambda_{\frac{s+1}{2}} - \lambda_{\frac{s-1}{2}}} + \sqrt{\frac{2d}{\pi}}\sigma e^{\frac{-\mu^2}{2\sigma^2}} \\ +\sqrt{d}\frac{2\sigma}{\pi}\frac{\sqrt{s}}{s+1}(\frac{s^2}{s^2-1})^{\frac{s}{2}}e^{\lambda_s - \lambda_{\frac{s+1}{2}} - \lambda_{\frac{s-1}{2}}} - 4\mu\epsilon\sqrt{ds}\frac{1}{2^s}(C_{s-1}^{\lceil\frac{s}{2}-1\rceil} - C_{s-1}^{\alpha-1}) & \text{if } s \text{ is odd} \end{cases}$$

3) This part discusses the condition for

$$\mathbb{E}(f)|_{s>1} < \mathbb{E}(f)|_{s=1} = \sqrt{\frac{2d}{\pi}}\sigma e^{-\frac{\mu^2}{2\sigma^2}} + \mu\sqrt{d}(1 - 2\Phi(-\frac{\mu}{\sigma}))$$

by further relaxing the upper bound of $\mathbb{E}(f)|_{s>1}$.

– if s is even, since $f_3 \leq 0$,

$$\begin{aligned} \mathbb{E}(f)|_{s>1} &< (\sqrt{\frac{2d}{\pi}}\mu + \frac{2\sigma}{\pi}\sqrt{d})e^{\lambda_s - 2\lambda_{\frac{s}{2}}} + \sqrt{\frac{d}{s}}\frac{2}{\pi}\sigma e^{\lambda_s - 2\lambda_{s/2}} + \sqrt{\frac{2d}{\pi}}\sigma e^{\frac{-\mu^2}{2\sigma^2}} \\ &\leq (\sqrt{\frac{2d}{\pi}}\mu + \frac{2\sigma}{\pi}\sqrt{d}) + \sqrt{\frac{d}{s}}\frac{2}{\pi}\sigma + \sqrt{\frac{2d}{\pi}}\sigma e^{\frac{-\mu^2}{2\sigma^2}} \\ &= \mu\sqrt{d}(\sqrt{\frac{2}{\pi}} + (1 + \frac{1}{\sqrt{s}})\frac{2\sigma}{\pi\mu}) + \sqrt{\frac{2d}{\pi}}\sigma e^{\frac{-\mu^2}{2\sigma^2}} \end{aligned}$$

Clearly $\mathbb{E}(f)|_{s>1} < \mathbb{E}(f)|_{s=1}$, if $\sqrt{\frac{2}{\pi}} + (1 + \frac{1}{\sqrt{2}})\frac{2\sigma}{\pi\mu} \leq 1 - 2\Phi(-\frac{\mu}{\sigma})$. This condition is well satisfied when $\mu >> \sigma$, since $\Phi(-\frac{\mu}{\sigma})$ decreases monotonically with increasing μ/σ.

– if s is odd, with $f_3 \leq 0$,

$$\mathbb{E}(f)|_{s>1} < (\sqrt{\frac{2d}{\pi}}\mu + \frac{2\sigma}{\pi}\sqrt{d})(\frac{s^2}{s^2-1})^{\frac{s}{2}} + \sqrt{d}\frac{2\sigma}{\pi}\frac{\sqrt{s}}{s+1}(\frac{s^2}{s^2-1})^{\frac{s}{2}} + \sqrt{\frac{2d}{\pi}}\sigma e^{\frac{-\mu^2}{2\sigma^2}}$$

It can be proved that $(\frac{s^2}{s^2-1})^{\frac{s}{2}}$ decreases monotonically with respect to s. This yields that

$$\mathbb{E}(f)|_{s>1} < (\sqrt{\frac{2d}{\pi}}\mu + (1+\frac{\sqrt{3}}{4})\frac{2\sigma}{\pi}\sqrt{d})(\frac{3^2}{3^2-1})^{\frac{3}{2}} + \sqrt{\frac{2d}{\pi}}\sigma e^{\frac{-\mu^2}{2\sigma^2}}$$

in this case $\mathbb{E}(f)|_{s>1} < \mathbb{E}(f)|_{s=1}$, if $(\frac{9}{8})^{\frac{3}{2}}(\sqrt{\frac{2}{\pi}} + (1+\frac{\sqrt{3}}{4})\frac{2\sigma}{\pi\mu}) \leq 1 - 2\Phi(-\frac{\mu}{\sigma})$.

Summarizing above two cases for s , finally

$$\mathbb{E}(f)|_{s>1} < \mathbb{E}(f)|_{s=1}, \text{ if } (\frac{9}{8})^{\frac{3}{2}}[\sqrt{\frac{2}{\pi}} + (1+\frac{\sqrt{3}}{4})\frac{2}{\pi}(\frac{\mu}{\sigma})^{-1}] + 2\Phi(-\frac{\mu}{\sigma}) \leq 1$$

□

5.7.3 Proof of Theorem 5.3.3

First, one can rewrite $f(\mathbf{r}, \mathbf{z}) = |\Sigma_{j=1}^{j=d}(r_j z_j)| = \mu|x|$, where $x \in N(0, d)$, since i.i.d $r_j \in N(0,1)$ and $z_j \in \{\pm\mu\}$ with equal probability. Then one can prove that

$$\begin{aligned}
\mathbb{E}(|x|) &= \int_{-\infty}^{0} \frac{-x}{\sqrt{2\pi d}} e^{-\frac{x^2}{2d}} dx + \int_{0}^{\infty} \frac{x}{\sqrt{2\pi d}} e^{-\frac{x^2}{2d}} dx \\
&= 2\int_{0}^{\infty} \frac{\sqrt{d}}{\sqrt{2\pi}} e^{-\frac{x^2}{2d}} d\frac{x^2}{2d} \\
&= 2\sqrt{d}\int_{0}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\alpha} d\alpha \\
&= \sqrt{\frac{2d}{\pi}}
\end{aligned}$$

Finally, it is derived that $\mathbb{E}(f) = \mu\mathbb{E}(|x|) = \mu\sqrt{\frac{2d}{\pi}}$.

Part III

Sparse Representation

6 Single object tracking

Sparse representation has recently been widely studied in the area of visual object tracking due to its good performance on object recognition. Up to now, little attention has been paid to the complexity of sparse representation, while most works focus merely on improving performance. By reducing the computation load related to sparse representation, this paper proposes by far the most computationally efficient tracking approach based on sparse representation, which empirically can obtain at least hundreds of times computation gains over current popular approaches involving sparse representation. The proposed approach simply consists of two stages of sparse representation, one is for object detection and the other one for object validation. In practice, it presents favorable tracking performance over state-of-the-art.

6.1 Introduction

Object tracking is a challenging task in computer vision community, since an object usually suffers from appearance changes and unpredictable motion. Recently, sparse representation has been introduced in this area for its robustness in representing objects with a wide range of corruption [128]. In terms of complexity, this method is also competitive since it only involves simple products of vectors and matrices. This chapter will show that the computation related to sparse representation can be significantly reduced within the simple tracking-by-detection scheme instead of the popular particle filter framework.

Most tracking works up to date explore sparse representation within the frame-

work of particle filter, in which it is used to measure the similarity between each particle and template/dictionary. It is clear that sparse representation introduces considerable computation load, since it has to be executed for each particle while empirically the number of particles is usually large (eg. 600 particles in [129–131]). Moreover, the complexity of sparse solution based on ℓ_1-regularization is also worth paying much attention. As it is known, the greedy solution algorithms of sparse representation hold the complexity of $\mathcal{O}(kn)$, where n is the total number of atoms in the dictionary and k is the number of atoms selected for sparse representation. Unfortunately, to obtain accurate similarity measure, the two above complexity parameters, k and n practically tend to be large. For instance, the value of n often reaches thousands, because most tracking works incline to incorporate a high-dimensional trivial template, denoted as $[I \ -I]$, into the dictionary to approximate noises or occlusion, where I is an identity matrix whose size is equal to the dimension of object feature.

Apart from computation cost challenge, sparse representation also cannot ensure reliable performance on similarity measure due to the potential overfitting solution. Exactly speaking, the particle with large corruption is probably represented with small error, since the sparsity of the solution cannot be known beforehand and the overfitting seems to be inevitable.

Since the framework of particle filter is computationally expensive for the application of sparse representation, we are motivated to develop a simple but efficient tracking-by-detection scheme by exploring the potential of sparse representation on object retrieval rather than on similarity measure. The proposed scheme consists of two-step sparse presentation for object detection and validation. In the first step the most possible object is retrieved by the largest sparse coefficient instead of investigating all candidate particles with similarity measure; and in the second step the detected object is further validated with a binary classifier based on sparse representation.

Compared to traditional framework of particle filter, the proposed scheme holds two obvious advantages on complexity. First, the potential object could be obtained by performing sparse representation only once, while to build a Monte-Carlo model for motion estimation, particle filter has to weight each particle with sparse rep-

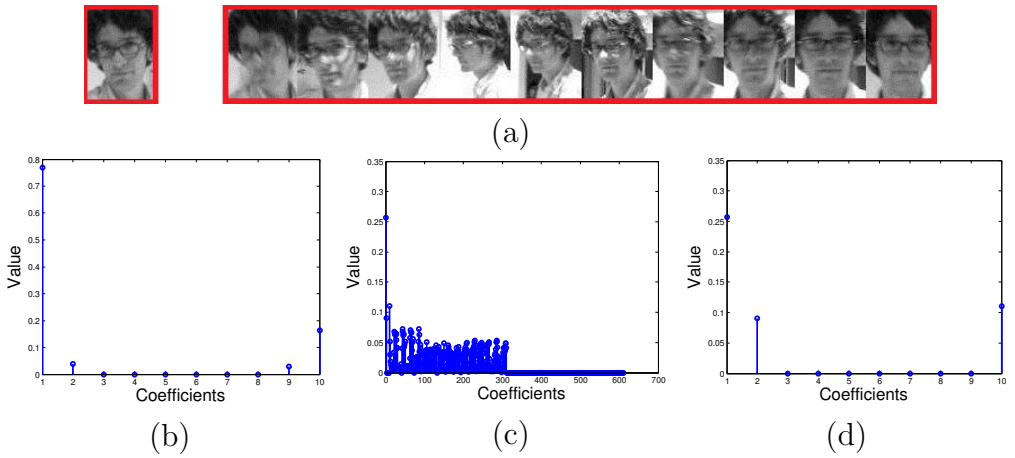


Figure 6.1: Examples of an object (left in (a)) sparsely approximated by a dictionary of ten objects (right in (a)) with sparse solution in (b), or by a dictionary combining the ten objects and a trivial template $[I - I]$ with sparse solution in (c). Recall that the identity matrix I has size equal to the size of object feature. Here each object is represented with a 300×1 vector. This means that the column size of dictionary is 10 in (b) and 310 in (c). The sparse solution in (b) has 4 nonzero entries and the representation error is about 10^{-1}. In contrast, the solution in (c) holds 300 nonzero coefficients, thereby lowering the representation error to 10^{-3}. For clarity, the first ten coefficients in (c) corresponding to above ten objects are detailed in (d). Comparing (b) and (d), one can observe that the sample most similar to the test object can be determined by the largest coefficient in (b) even with much higher representation error.

resentation; second, as the example shown in Figure 6.1, compared to similarity measure, object retrieval based on the largest coefficient allows to be successfully implemented with both a smaller size dictionary and a higher representation error, such that the solution complexity of sparse representation is also significantly reduced. In addition, the binary classifier based on sparse representation also shows obvious computation advantage over other traditional classifiers, e.g., SVM, since it only involves simple operations of matrix-vector product. More significantly, with low-complexity implementation, the proposed approach still presents favorable performance.

The rest of this chapter is organized as follows. In the next section, tracking works related to sparse representation are briefly reviewed. In section 6.3, the sparse representation based classification is introduced. In section 6.4, the proposed tracking scheme with two-step sparse representation is described and analyzed. In section 6.5, extensive experiments are conducted with comparison to the state-of-the-art.

Finally, a conclusion is given in section 6.6.

6.2 Related Work

Extensive literature has been proposed on object tracking. We here review the major work related to sparse representation. Until now, most tracking works involving sparse representation are implemented within the framework of particle filter by measuring the weight of particles with representation error. Therefore, generally these works are developed based on two major goals: improving the robustness of sparse representation and reducing the complexity of the sparse solution.

Mei and Ling [132] first explore sparse representation into an on-line tracking system, and introduce the trivial template to approximate noise and occlusion. Later, to improve the high-dimensional feature selection, Liu et al. [133] attempt to learn discriminative high-dimensional feature using dynamic sparsity group. To reduce the sensitivity to background noise in selected object area, Wang et al. [129] and Jia et al. [134] apply the sparse coding histogram based on local patches to describe object. Zhong et al. [135] propose a collaborative model that weights particles by combining the confidences of local descriptor and holistic representation. In a particle filter framework, computation cost is a great challenge as it grows linearly with the number of particles. Thus, a few works have been proposed to alleviate this problem. Li et al. [136] perform sparse representation in compression domain. Mei et al. [130] discard insignificant samples by limiting their linear least square errors before resampling particles with more computationally expensive ℓ_1-regularization. Based on the accelerated proximal gradient approach, Bao et al. develop a fast solver for the ℓ_1-regularization problem [137]. Liu and Sun [138] attempt to weight each particle only with corresponding sparse coefficient such that sparse representation is allowed to be conducted only once. This method seems very attractive in complexity. However, it should be noted that, in theory the magnitude of each coefficient cannot be ensured 'proportional' to the similarity/correlation between corresponding particle and query object. Precisely, in terms of least square, it can be derived that the 'proportion' exists only when the sub-dictionary corresponding

to sparse coefficients is orthogonal. Obviously the condition above is hard to satisfy in practice. Furthermore, it should be strengthened that we cannot detect the case where object is out of the scene with the magnitudes of sparse coefficients. Zhang et al. [139] propose to jointly represent particles by using multi-task learning to explore the interdependencies between particles; In addition, to detect occlusion, the nonzero coefficients in trivial template are proposed to locate occluded pixels in [130]. This novel technique seems impractical and imprecise, since both the error bound for sparse solution and the threshold for coefficient value defining occlusion are hard to be determined empirically. For the overfitting case in Figure 6.1(c), all pixels are likely to be classified as occlusion though in fact there is no occlusion. Apart from the tracking work involving particle filter, Liu et al. [140] once attempt to use only mean-shift algorithm to search local patches in a fixed window, with sparse coding histogram as feature.

Adaptive appearance model is the crucial development of recent object tracking work. This technique is initially developed to model dynamic appearance by online object sample updating [132] or learning [141]. To avoid identity drift caused by false or unreliable samples accumulating in template, the ground-truth detected manually or automatically from the first frame is further modeled to retain the identity of object. Our work also adopts similar scheme to enhance the reliability of object detection and validation.

Existing tracking methods can be roughly categorized as either generative or discriminative. The generative method tries to find the potential object area most similar to object appearance model. Most trackers [131, 141–143] within the framework of particle filter belong to this method. The discriminative method formulates tracking as a binary classification problem that distinguishes object from background [144–147]. Clearly, the former method is suitable for adapting to variable object appearances, while the latter is robust to identity drift. In this chapter, both previous advantages are efficiently combined into the tracking-by-detection scheme of two-step sparse representation.

6.3 Sparse representation-based classification

This section briefly introduces sparse representation-based classification with the typical face recognition as example. Let vector $\mathbf{y} \in \mathbb{R}^{m\times 1}$ denote a test face, and matrix $\mathbf{D} = [D_{G_1}, D_{G_2}, \ldots, D_{G_N}] \in \mathbb{R}^{m\times n}$ be a dictionary consisting of N classes of labeled face vectors, where the i-th sub-matrix $D_{G_i} = [D_{i_1}, D_{i_2}, \ldots, D_{i_{n_i}}]$ includes n_i samples and $\sum_{i=1}^{N} n_i = n$. Then we ideally suppose that test face can be approximated by a linear combination of few labeled face vectors, namely

$$y = \mathbf{D}\beta + \epsilon \tag{6.1}$$

where β is required to hold at most $k \ll n$ nonzero *positive* entries; and ϵ is a tolerated error. Subsequently, the feature vector $\mathbf{y}$ is viewed as close to the subspace of labeled samples corresponding to the nonzero entries of β. In other words, it can be identified as the class

$$\hat{i} = \underset{i}{\operatorname{argmax}}\{\delta_i(\beta)|1 \le i \le N\}, \tag{6.2}$$

where $\delta_i(\beta)$ is a function that sums the elements of β corresponding to D_{G_i}. The solution to k-sparse vector β now can be simply derived with greedy algorithms of complexity $\mathcal{O}(mnk)$, such as OMP [58] or LARS [148]. Note that, to reduce representation error, most works [129, 130, 132–134, 136, 139] tend to add identity matrices $[I - I] \in R^{m\times 2m}$ to the dictionary $\mathbf{D}$, thereby dramatically increasing the solution complexity.

It is worth mentioning a special case where the test face is novel and out of the database. In this case, the nonzero entries of β empirically incline to scatter among some different classes instead of focusing on some special class. So the novel face can be detected by a threshold

$$\max\{\delta_i(\beta)|1 \le i \le N\} < \gamma \sum_{j=1}^{n} \beta_j \tag{6.3}$$

where $0 < \gamma < 1$ is an empirical parameter. This character enables the capability of

outlier detection of the classifier we will propose in section 6.2.

6.4 Proposed tracking scheme

In this section, we first detail the tracking scheme simply based on two-step sparse representation, and then quantify its advantage of complexity.

6.4.1 Random projection-based feature selection

In real tracking scenarios, it is usually hard to obtain ideal object features. Thus, color histogram and raw image have been two kinds of popular features. Here, the random projection of raw image is adopted as feature [115], because the integration of random projection and sparse representation is competitive both in complexity and performance [128]. Let $\mathbf{R} \in \mathbb{R}^{d \times m}$ be a random projection matrix, $d < m$. Incorporating random projection, formula (6.1) is reformulated as

$$\mathbf{R}y = \mathbf{RD}\beta + \epsilon \tag{6.4}$$

which simply implements feature selection with dimensionality reduction. Usually, the random projection matrix $\mathbf{R}$ tends to be generated by sampling entries from Gaussian distribution. For simplicity, here we apply a more sparse version, which holds only one nonzero entry taking values ± 1 with equal probability in each column. This kind of matrix has shown better overall performance on feature selection than other more dense matrices [10], and also performs well in the following tracking experiments.

Note that, despite holding low implementation complexity, random projection clearly is not the best feature selection tool in terms of performance. However, considering the variation of object appearance, the feature comparison based on the sum of few randomly selected pixels should be more reasonable than the conventional pixel-wise comparison. This also explains why it presents satisfactory performance in the following experiments.

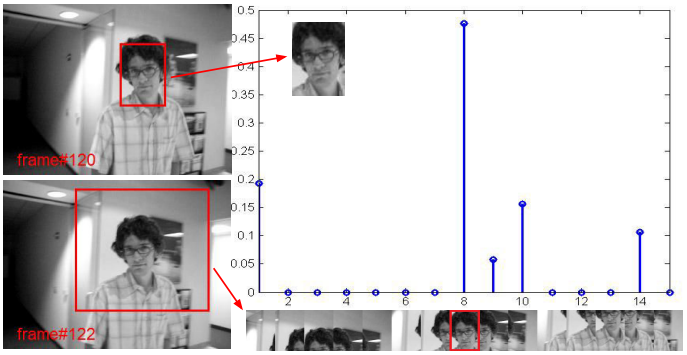


Figure 6.2: Labeled object from former frame can be sparsely approximated by a dictionary consisting of overlapped local patches in some area of current frame. The local patch corresponding to the largest coefficient is prone to indicate the position of estimated object.

6.4.2 Object detection

In this subsection, object detection is simply regarded as a process of retrieving known object in the current frame with sparse representation. As illustrated in Figure 6.2, suppose that the known/query object extracted from former adjacent frame, denoted as y, could be sparsely approximated by a dictionary, represented by $\mathbf{D}$, consisting of overlapping local patches extracted from current frame. Then with formula (6.4), the position of candidate object will be indicated by the local patch of $\mathbf{D}$ corresponding to the largest component of β. In this sense, the performance of retrieval is determined by the reliability of query object y extracted from former frame. Here, we define a special dictionary $\mathbf{Y} = [Y_s\ Y_d]$ to further model query object y. As it appears in the state-of-the-art [131,135], Y_s denotes static appearance model consisting of ground-truth manually or automatically detected from the first frame as well as its perturbations with small Gaussian noise, and Y_d represents dynamic appearance model collecting some object samples extracted from recent frames. To represent object variations while avoiding identity drift, in our method a set of query samples, rather than one, are randomly selected from above two models to vote final result. In other words, the average of sparse solutions from several query objects is used to define the candidate object area. For details, please see Algorithm 6.1. The random selection of several samples results from the following two reasons: 1) it is hard to ensure that the samples of the dictionary are all reliable. Further, if

Algorithm 6.1 Object detection.

Definitions: Let $\mathbf{Y} = [Y_s \; Y_d]$ be a set of known object samples from former frames, where Y_s represents the subset of static samples from the initial frames and Y_d denote the subset of dynamic samples from the recent frames, and $\mathbf{D}$ be the dictionary consisting of overlapping local patches extracted from some regions of current frame, with a fixed space layout. And then suppose $\{y^j : 1 \leq j \leq N_c\}$ is a set of N_c query samples randomly selected from Y_s or Y_d.

1. The average of sparse coefficient $\bar{\beta} = \frac{1}{N_c}\sum_{j=1}^{N_c}\beta^j$ is derived with respect to $\mathbf{R}y^j = \mathbf{RD}\beta^j + \epsilon$, where $\mathbf{R}$ is random projection matrix for feature selection;
2. The candidate object is located by the local patch with index $\hat{i} = \text{argmax}_i\{\bar{\beta}_i\}$, where $\bar{\beta}_i$ indicates the i-th component of vector $\bar{\beta}$.

there are corrupted samples, they tend to occur over consecutive frames. In this sense, randomness may lead to the selection of more reliable samples on the whole; 2) random projection theoretically should get better feature selection performance, if it could be conducted several times rather than only once [115]. Thus, the random projection of several samples will improve our feature selection performance.

In the following experiments, 10 samples from static model and 5 samples from dynamic model, are randomly selected for object retrieval. This means that identity preservation is more important than dynamic adaptation. Considering the balance between computation and performance, random projection is carried out only once for each sample. Of course, if the number of samples is limited, we should increase the times of repeating random projection, as we will do in the following subsection. Overall, this indicates that here the sparse representation is conducted 15 times for object detection, which is still much less than hundreds of times that particle filter usually requires [129–131]. Likewise, considering the continuity of object moving between adjacent frames, the search region usually can be limited to a relatively small area, e.g. twice or three times the object size. And the local patches are extracted with a sliding overlapping window of same size as the former tracked object. Empirically, the number of local patches could be limited at the level of tens, and thus the computation cost for sparse representation is relatively small.

Note that, the confidence of the detected object cannot be ensured with the largest coefficient. For instance, even though object has disappeared in the scenario, the object location scheme still presents an 'object area' with the largest coefficient

Algorithm 6.2 Object validation and template updating.

Definitions: Let y_c denote the candidate object derived in Algorithm 6.1, and $\mathbf{Z} = [Z_{G_p}\ Z_{G_n}]$ be the dictionary consisting of object samples and background samples, where the G_p and G_n denote the index sets corresponding to above two classes of samples. Note that $Z_{G_p} \subseteq \mathbf{Y}$ of Algorithm 6.1.

1. The average of sparse solutions $\bar{\beta} = [\bar{\beta}_{G_p}\ \bar{\beta}_{G_n}] = \frac{1}{N_r}\sum_{j=1}^{N_r}\beta^j$ is derived with $\mathbf{R}_j y_c = \mathbf{R}_j\mathbf{Z}\beta^j + \epsilon$, where $\mathbf{R}_j$ denotes the j-th random projection with $j \in \{1, 2, ..., N_r\}$, and correspondingly β^j is the sparse solution;
2. The candidate object is labeled, if $\text{argmax}_i\{\bar{\beta}_i\} \in G_p$ and $0.5 < ||\bar{\beta}_{G_p}||_1/||\bar{\beta}||_1 < 0.8$, where $\bar{\beta}_i$ denotes the i-th element of vector $\bar{\beta}$;
3. The candidate object is labeled and updated for dynamic object model Y_d, and its neighboring background patches are updated for Z_{G_n}, if $\text{argmax}_i\{\bar{\beta}_i\} \in G_p$ and $||\bar{\beta}_{G_p}||_1/||\bar{\beta}||_1 > 0.8$;
4. If both steps 2 and 3 cannot be performed, object is assumed to keep still or move with a constant velocity;
5. If step 4 is performed for numerous consecutive frames, object detection in Algorithm 1 is performed again in a larger region.

of sparse solution. In this sense, we have to develop a binary classifier to further validate the detected object.

6.4.3 Object validation and template updating

In this subsection, a binary classifier based on sparse representation is developed for discriminating object from background or outliers. Sparse representation is adopted here for the following four reasons:

1) it is computationally competitive since it only involves simple operations of matrix-vector product;

2) the decision can be easily derived in terms of the distribution of sparse coefficients;

3) compared to traditional binary classifiers, it has an exclusive advantage, that is the outlier which cannot be previously trained can also be detected since in this case the sparse coefficients tend to scatter rather than focus [128]. In practice, this also allows to train a relative small dictionary of positive and negative samples;

4) in practice the discrimination between object and background seems to be

a multi-class classification problem rather than a binary classification problem, since complex and dynamic background usually involves kinds of feature subspaces in which some ones might be close to object feature. In this case, two opposite half-spaces trained by traditional binary classifiers, like SVM [26], probably overlap with each other, thereby deteriorating the classification performance. In contrast, sparse representation is immune to this problem because it partially explores the similarity between individual samples rather than directly dividing the samples space into two parts [1].

Next we move on to detailing the proposed classifier. Similar with formula (6.4), a candidate object, denoted by y_c, is assumed to be sparsely approximated by a dictionary/template $\mathbf{Z} = [Z_{G_p}\ Z_{G_n}]$

$$\mathbf{R}y_c = \mathbf{RZ}\beta + \epsilon, \tag{6.5}$$

where Z_{G_p} and Z_{G_n} denote the subset of positive samples and the subset of negative samples, respectively; and two subscripts G_p and G_n correspond to two subsets of column indexes of positive samples and negative samples in dictionary $\mathbf{Z}$. The positive samples Z_{G_p} come from aforementioned static and dynamic appearance models $\mathbf{Y}$, and the negative samples Z_{G_n} are collected by a sliding overlapping window from the neighborhood of tracked object, where partial object region is included as opposed to relatively complete object region in positive samples. Correspondingly, the sparse solution β is also divided into two parts: $\beta = [\beta_{G_p}\ \beta_{G_n}]$. Note that, here β also allows high sparsity, i.e. it holds at most 10 nonzero entries in our experiments, since the subsequent binary classification based on object retrieval is insensitive to representation error. In terms of binary classification, *two* rules are used to define the *positive* result. *One* is that the largest coefficient of sparse solution β corresponds to a positive sample of Z_{G_p}; namely, $\text{argmax}_i\{\beta_i\} \in G_p$. And *the other* is that the sparse coefficients corresponding to positive samples, β_{G_p}, take higher energy than the sparse coefficients corresponding to negative samples, β_{G_n}; that is, $||\beta_{G_p}||_1/||\beta||_1 > 0.5$. Empirically, the latter criterion is more strict than the former since it measures the similarity between the candidate object and the positive sub-

space instead of individual positive samples. In our method, to obtain a relatively fluent tracking trajectory, an object is allowed to be labeled when above two criteria are satisfied. But for template updating, in the following experiments the second criterion is imposed on a stricter threshold $||\beta_{G_p}||_1/||\beta||_1 > 0.8$ for more reliable features. In practice the threshold value is allowed to be tuned empirically. But in the following experiments it is fixed to 0.8. Note that random projection is required to be carried out several times to achieve better feature selection performance for the unique candidate object [115], e.g., it is repeated 5 times in our experiments. And so the sparse solution β is finally derived by averaging. In addition, it should be mentioned that the decision of the classifier is based on sparse coefficients rather than on representation error. Therefore it can also be implemented with a relatively high representation error. The whole flow of object validation is illustrated in Algorithm 6.2.

It is necessary to recall that the proposed classifier holds an exclusive advantage, because it can efficiently detect the outliers which cannot be previously trained. In other words, for the potential outliers, like significant object appearance change and dynamic background, sparse coefficients incline to scatter among positive and negative subspaces rather than focus on one of them, such that $||\beta_{G_p}||_1/||\beta||_1 \approx 0.5$. In this case if we apply a much stricter threshold for template updating, e.g., $||\beta_{G_p}||_1/||\beta||_1 > 0.8$, the outliers can be successfully detected and excluded.

In addition, considering the outliers can be excluded efficiently, we need not build a robust background model because the untrained background elements could be detected as outliers. Clearly this will significantly reduce the load of updating the background model, which probably accumulates a great amount of information over time. In our experiments, only 100 samples are adopted for background model and 50 samples for object model. The samples are updated with the simple way of replacing the oldest with the novel. By the way, we also test the k-means algorithm for samples updating, which seems more suitable for collecting the redundant background samples [135]. However, it does not show the desired advantage over the simple way of replacing the oldest one with the novel. This verifies the conjecture that the proposed classifier is not sensitive to the robustness of background model.

6.4.4 Computation cost related to sparse representation

As stated in the former introduction, there are two major factors affecting the computation cost related to sparse representation. One is the product of the dimension n of dictionary and the number k of selected atoms for sparse representation, namely $\mathcal{O}(kn)$; and the other is the times of repeating sparse representation. Compared to traditional tracking works involving sparse representation, the proposed tracking scheme enjoys obvious advantages in terms of above two factors. For instance, on one hand, in the proposed scheme, object retrieval based sparse representation significantly reduces both the dimension of dictionary and the number of selected atoms, since it is insensitive to representation error. Precisely, 1) the high dimensional trivial template $[I \ -I]$ here is not required, which shares the same size with object feature; 2) few atoms are allowed for sparse representation, e.g., the maximum of k is set to 10 in our experiments; conversely, for traditional tracking methods, k tends to increase limitlessly until the representation error stops decreasing or achieves some given threshold. On the other hand, the frequency of repeating sparse representation is also obviously reduced. Exactly speaking, here sparse representation is required to be conducted only several tens of times, e.g., 20 times in our experiments. Furthermore, if object feature is reliable enough, the proposed scheme allows to perform sparse representation only once at each step. In contrast, traditional methods usually have to conduct sparse representation for each particle, and empirically this indicates hundreds of times of sparse representation in terms of the number of particles exploited in practical applications [129–131].

6.5 Experiments

We evaluate the proposed approach on ten challenging videos, among which eight are publicly available [1] and two are produced by ourselves. For comparison, we perform four typical tracking algorithms with adaptive appearance models: IVT tracker [141], L1 tracker [132], PLS tracker [131] and SCM tracker [135]. The for-

1. http://www.cvg.rdg.ac.uk/PETS2009/; http://www.gris.informatik.tu-darmstadt.de/~aandriye/data.html; http://faculty.ucmerced.edu/mhyang/pubs.html

mer two trackers are early presented for adapting appearance variations by on-line updating or learning template subspace, and the latter two trackers are most recent works that further explore object ground-truth for object identity model, and also show favorable performance over other state-of-the-art methods. For fair comparison, in our experiments the four trackers are all implemented with their original codes, and initialized with same parameters. As for parameters tuning, we use the original parameters for L1 tracker and PLS tracker. Since IVT tracker and SCM tracker both have provided several options of parameters for some popular videos, we adopt the original parameters for the same videos, and select proper parameter options for our own videos, e.g., using their parameters for 'head' and 'pedestrian' to separately track 'head' and 'pedestrian' of our videos. Note that, some methods here perform worse than their original papers. This might be explained by the slight changes of the initial areas of tracked objects.

The proposed algorithm is mainly implemented with Matlab code. LARS algorithm of SPAMS package [149] is used for sparse solution, for which the threshold on reconstruction error is set to 10^{-3} and the maximum number of nonzero entries is no more than 10. To reduce computation and memory load, extracted subimage of same size with tracked object, is first resized to 32×32 and then compressed to a 200×1 vector by random projection for simple computation of sparse representation. The whole scheme achieves the speed of about 3 fps on a PC with 2.67Hz double cores, which is faster than recent work [134] also programmed with Matlab and SPAMS package. Note that, compared to existing tracking works involving sparse representation [137], our software is not the best in speed, though the proposed scheme has been proved significantly reducing the computation cost related to sparse representation. This problem results from the inefficiency of Matlab code on the feature selection. Precisely, during feature selection, the generation of high dimensional random matrix and the multiplication between the dictionary and random matrix are both time-consuming for serial Matlab code. However, these problems will turn into advantages in parallel circuit implementation.

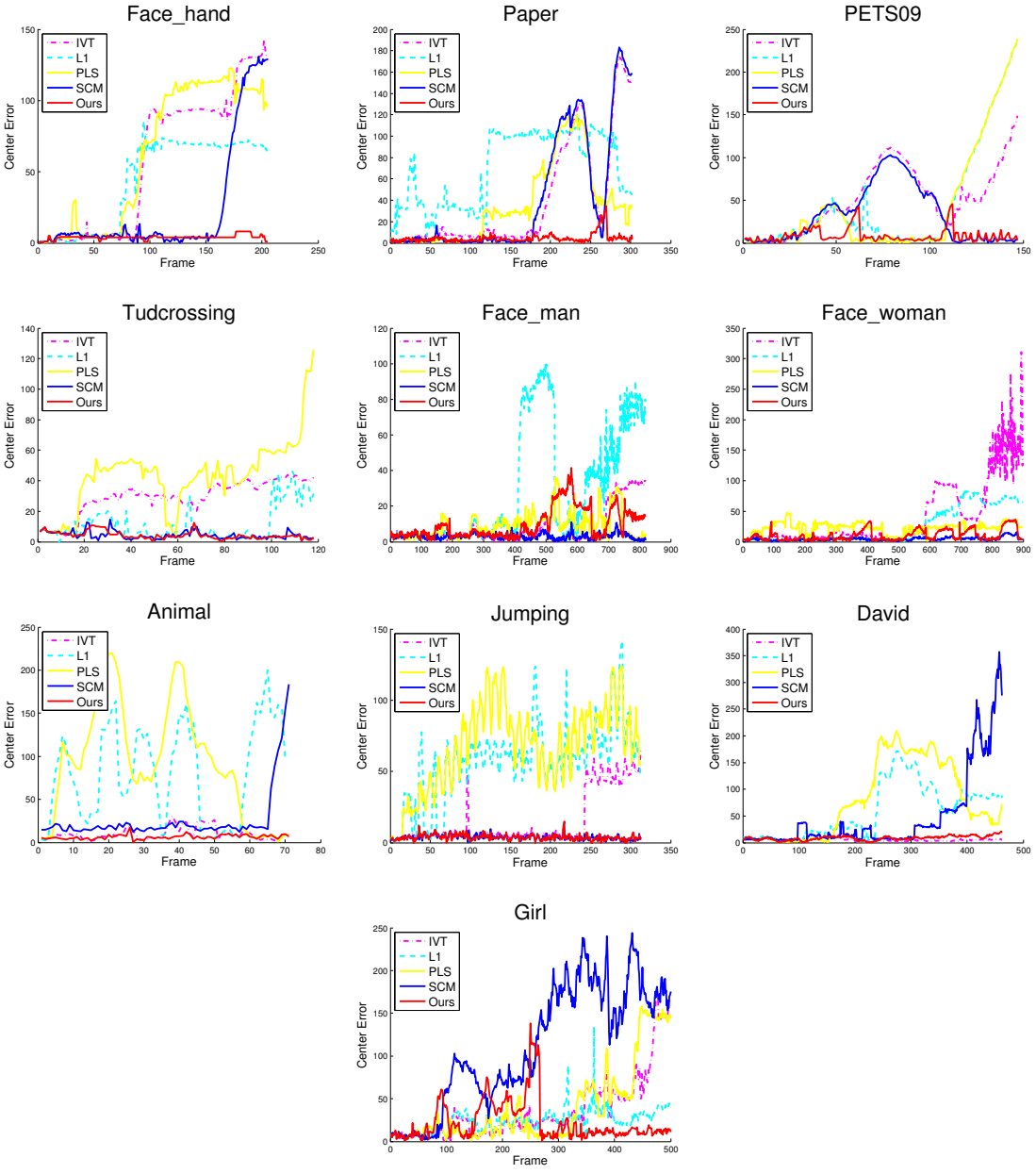


Figure 6.3: Quantitative evaluation of the trackers in terms of center location error on all test sequences.

Table 6.1: Average center location errors (in pixel). The best and second best results are shown in red and blue fonts, respectively.

Video clip	IVT	L1	PLS	SCM	Ours
Face_hand	57.80	43.98	62.08	**23.82**	**4.08**
Paper	38.77	70.42	**34.77**	40.89	**4.45**
PETS09	51.26	45.04	42.98	**32.12**	**8.47**
Tudcrossing	29.11	11.49	43.11	**4.36**	**4.23**
Face_man	**7.41**	28.96	9.091	**3.06**	9.74
Face_woman	39.99	24.06	22.57	**4.68**	**10.45**
Animal	10.02	81.02	91.94	25.75	**6.39**
Jumping	15.25	55.68	66.93	**3.73**	**3.77**
David	**5.49**	58.32	77.73	47.20	**8.65**
Girl	36.50	**24.53**	39.51	110.49	**22.68**

Table 6.2: Average overlap rates between target region and ground-truth. The best and second best results are shown in red and blue fonts, respectively.

Video clip	IVT	L1	PLS	SCM	Ours
Face_hand	0.40	0.35	0.35	**0.70**	**0.93**
Paper	0.48	0.12	0.44	**0.54**	**0.82**
PETS09	0.20	**0.39**	0.38	0.38	**0.64**
Face_man	**0.56**	0.36	0.52	**0.67**	**0.56**
Face_woman	0.46	0.54	0.43	**0.82**	**0.67**
Tudcrossing	0.15	0.54	0.09	**0.69**	**0.74**
Animal	**0.74**	0.23	0.21	0.57	**0.84**
Jumping	0.52	0.11	0.07	**0.80**	**0.82**
David	**0.64**	0.26	0.25	0.37	**0.39**
Girl	0.37	**0.48**	0.39	0.13	**0.51**

6.5.1 Quantitative evaluation

We measure the tracking accuracy of aforementioned algorithms based on the center location error and the overlap rate. The center location error is the Euclidean distance between the central points of tracked object R_T and the corresponding manually labeled ground-truth R_G. The error plots on all test sequences are shown in Figure 6.3, and the average errors are given in Table 6.1. The overlap rate [150], defined as $area(R_T \cap R_G)/area(R_T \cup R_G)$, evaluates the success rate. Table 6.2 illustrates the average overlap rates for ten videos. As Figure 6.3 illustrates, it is easy to state that the proposed approach achieves relatively persistent and stable tracking, and obtains better overall performance than the other four trackers on the ten videos. However, it does not obtain the best average performance on all videos as shown in Tables 6.1 and 6.2, since we simply exploit a fixed-size rectangle of raw image to represent the object. On one hand, it is sensitive to severe scale

change; on the other hand, in terms of the definition of overlap rate, it tends to suffer from small overlap rate as tracked rectangle area is usually not consistent with ground-truth in size. For instance, even if we successfully capture the face in sequence *david*, the overlap rate is still small as our rectangle is usually larger than the face area. However, this problem could be addressed by applying more advanced representation method in the future.

6.5.2 Qualitative evaluation

In addition to quantitative evaluation, the qualitative evaluation, as illustrated in Figure 6.4, is performed based on several typical deformations that the videos often suffer from.

Occlusion: Complete or partial occlusion is a major challenge for tracking systems with adaptive appearance model since it usually leads to identity drift of the template. In fact, the proposed approach owns obvious advantages for long-time and heavy occlusion for its robust tracking-by-detection scheme. To better highlight the performance, we produce two more challenging videos against long-time complete occlusions: sequences *face_hand* and *paper*. In the sequence *face_hand*, target face is completely occluded with hands for a long period. In the sequence *paper*, the target paper is also completely occluded twice by other paper. In addition, for complete occlusion, two challenging examples about pedestrians are given by sequences *PETS09* and *Tud_crossing*; and the case on face occlusion in the sequence *girl* is also evaluated. For partial occlusions, two examples of partial face occlusion are proposed by sequences *face_man* and *face_woman*.

The proposed approach performs well on aforementioned videos without identity drift. In contrast, other four trackers all fail when long-time complete occlusion occurs or the occlusion shares similar feature with target. For instance, in the sequence *face_hand*, IVT, L1 and PLS early drift to background when short-time occlusion occurs, and SCM finally drifts to hands when the face is covered by hands during fifty frames. In the sequence *paper*, it seems difficult for them to distinguish between two papers. L1 begins to lose track as target paper moves, while other three trackers absolutely drift to occlusion. Among these four trackers, SCM shows better

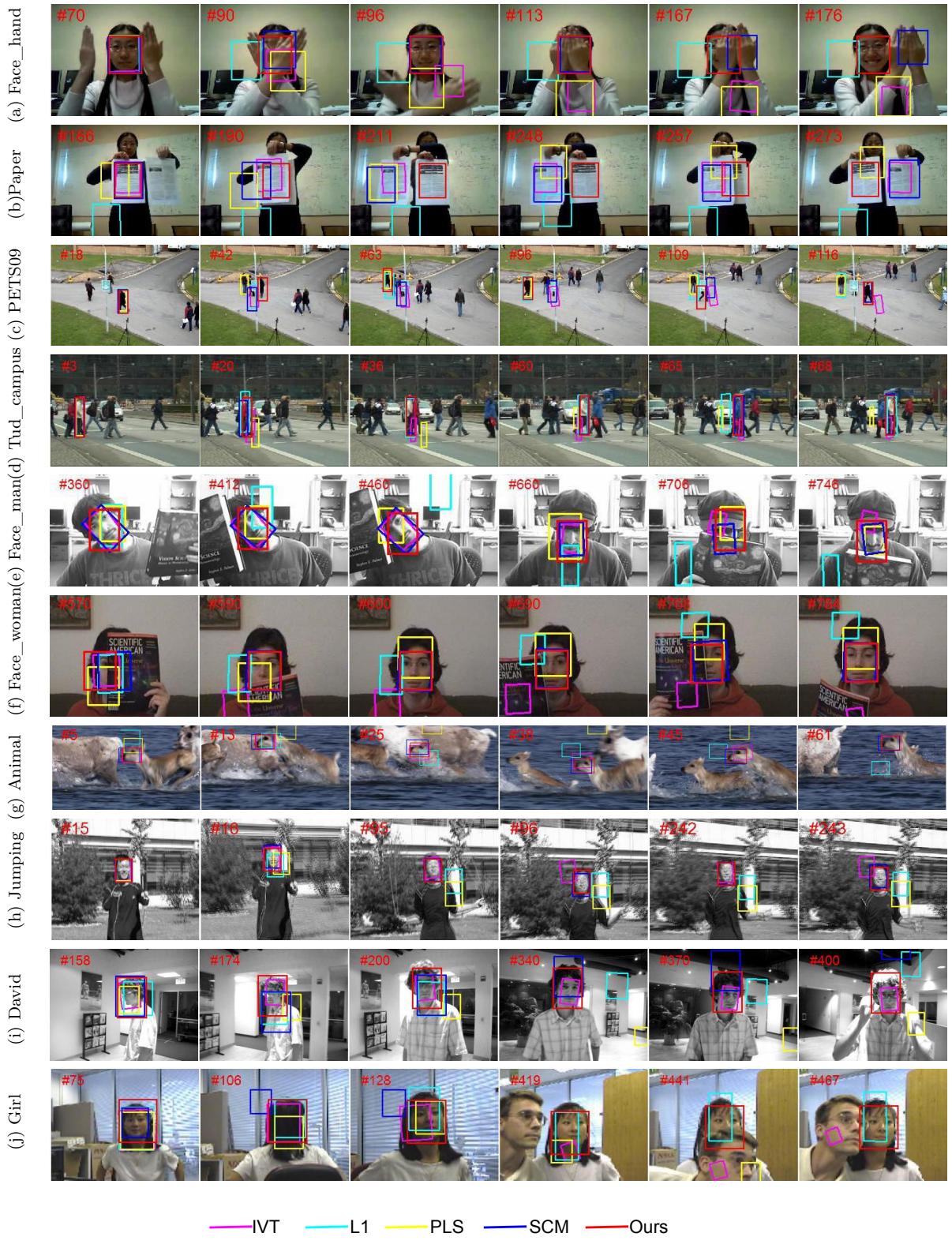


Figure 6.4: Tracking examples of five methods on ten challenging videos.

overall performance on identity preservation thanks to exploring the ground-truth to search the object. In the sequence *PETS09*, it successfully recovers the target lost during the occlusion. Nevertheless, it remains sensitive to long-time occlusion as the sequence *face_hand* illustrates. This implies that in fact the occlusion cannot be detected efficiently by SCM and the attribute of dynamic object model will gradually change over time.

Motion & Blur: Fast or abrupt motion is a great challenge for tracking systems with particle filter due to their continuous motion estimation parameters. To cope with this problem, they usually have to exploit more particles holding the distribution of larger variance. For the proposed approach with tracking-by-detection system, unexpected motion can be easily caught by a larger retrieval region. As for motion blur, it is usually regarded as a challenge for object recognition. For the proposed approach without accurate motion estimation, abrupt motion cannot be caught if the recognition fails in the first step. However, fluent tracking result in sequences *animal* and *jumping* show that our approach works well in this case. This indicates that random projection of raw image within sparse representation to some extent to is immune to the blur. SCM also addresses this problem well with sparse coding histogram as feature. In contrast, for the sequence *jumping*, L1 and PLS early drift from the target when the blur emerges in frame 16, while IVT keeps on tracking to frame 243 by more advanced incremental subspace learning method.

Scale & Rotation: There are drastic scales as well as in-plane and out-of-plane rotations in both sequences *david* and *girl*. It is clear that our approach is sensitive to scale changes since it is not addressed in object representation. But the proposed approach can successfully detect failed detection caused by severe scaling and rotation in object validation. In other words, this prevents identity drift of dynamic object appearance model. So in the sequences *david* and *girl*, the proposed approach successfully recaptures the object after severe appearance changes and tracks it up to the end. In contrast, for severe appearance changes caused by out-of-plane rotation, like the transformation from face to the back of head in sequence *girl*, other four trackers are likely to drift for ever.

Illumination: In theory, sparse representation based on normalized image vec-

tor is insensitive to illumination changes. So here it is not an important performance index. In the sequence *david*, an object walks out from the dark room to the area with spot lights in the early few frames without scaling and rotation, the five trackers all present perfect results.

Overall performance: The proposed approach shows better overall performance on the ten videos benefiting from two advantages. First, the robust recognition of sparse representation based on static and dynamic features ensures the fluency of tracking trajectory. Second, the robustness of the proposed classifier on outlier detection efficiently retains the purity of object appearance model. Although SCM and PLS both explore ground-truth to identify the object, they lack efficient method to detect and prevent outliers from updating dynamic object model. In fact, SCM obviously outperforms PLS in our experiments. This can be partially explained by the fact that SCM explores both static and dynamic features to weight particles while PLS only applies dynamic features to search particles. The other reason may be the difference between the features they apply. IVT and L1 cannot cope with severe appearance changes since their template updating mechanisms are not developed to maintain object identity.

6.6 Conclusion

This chapter has presented an efficient and effective tracking-by-detection scheme by exploring the sparse representation for object retrieval and object validation. Compared to conventional tracking methods involving sparse representation, the proposed approach achieves significant computation gains by reducing both the times of repeating sparse representation and the computation load of sparse solution. In particular, if the feature is robust enough, the object can be tracked by conducting only two times of sparse representation. It is necessary to strengthen that the computation gain is quantified in this paper, though our experiment simply implemented with Matlab code does not present the desired speed.

Extensive experiments on challenging sequences show that the proposed approach performs favorably on identity preservation. For instance, although the scaling and

rotation of object appearance are not specifically addressed in the proposed scheme, we can successfully recapture the object after long-time occlusion or serious deformation. The robust tracking performance indicates that sparse representation is indeed a powerful tool for both object retrieval and classification. Specifically, with scattering sparse coefficients, the binary classifier based on sparse representation can easily detect the outliers which are not included in current training dictionary. This enables the classifier to work well even when the dictionary contains relatively few background samples.

7 Multiple objects tracking

Multi-object tracking is a technique that locates and recognizes a number of objects in some sequential video frames. Compared to single object tracking, it presents more challenges on objects discrimination. In this chapter, we will show that the sparse representation can also be used to recognize multiple objects with dynamic appearances [13].

7.1 Introduction

As reviewed in last chapter, sparse representation has been extensively studied for single object tracking, while the application to multi-object tracking has not received attention. In this chapter, we attempt to test the capability of sparse representation on the classification of multiple objects with dynamic appearances. The tracking-by-detection scheme is simply applied as follows. First, the labeled samples from previous frames are collected and updated in the dictionary, and then the samples detected in a new frame are sparsely represented with the dictionary. These new samples can be identified with the labeled samples of the dictionary in terms of the distribution of sparse coefficients, as done in the former chapter. For simplicity, we only test some videos with static scenes which can be addressed using simple object detection techniques. In the following parts, we detail the tracking scheme and present the experimental results.

(a) (b)

Figure 7.1: Tracking results only with color feature (a) and with feature integrating color and 2-dimensional coordinates (b). In (a), *object* 6 switches into *object* 2 after 13 frames due to similar appearance.

7.2 Tracking scheme

7.2.1 Object detection and representation

For static camera without dense scenes, background subtraction is efficient for body area detection. Here in terms of computation cost, we still use traditional method [151] based on statistically modeling and pixel-wise subtraction instead of these complex methods with tiny performance gain [152] [153]. Two cascaded RGB histograms are used to represent the upper and lower parts of human body. Furthermore, to discriminate objects sharing similar appearance, the location information, normalized 2-dimensional coordinates of the object center, is applied to represent object by concatenating it with normalized RGB vector. Its benefit is simply illustrated in Figure 7.1. And the weight ratio between the two kinds of vectors is tuned empirically.

7.2.2 Overlapping

With background subtraction, the overlapping objects are subject to be detected as one object. And objects with tiny overlap usually can be detected by obvious size variation as shown in Figure 7.2(a). To avoid falsely updating samples or defining novel objects in the dictionary, detected overlap is not processed and nor labeled in our tracking scheme. As for the undetected overlap as Figure 7.2(b) shows, it will be recognized as a unique object. Empirically, the overlap tends to be linearly approximated by the objects it includes during sparse representation. Thus, the overlap is likely to be defined as the larger object it includes in our experiments. In

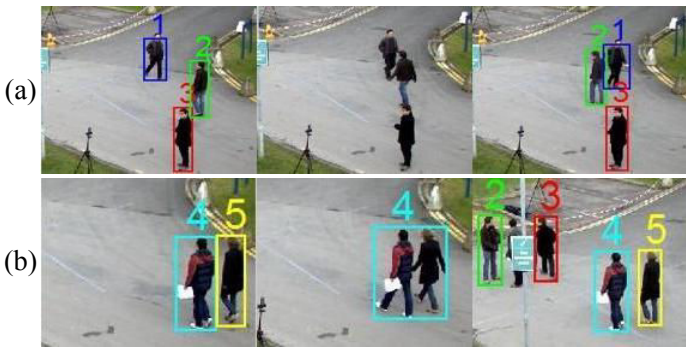


Figure 7.2: The obvious overlap detected in (a) is not recognized or labeled. And the undetected overlap in (b) is usually recognized as the object at the forefront.

this sense, we naturally avoid defining the overlap as novel object.

7.2.3 Online dictionary updating

Online dictionary updating attempts to store and train recently recognized samples which are expected to be most similar to incoming test objects. This is critical for object recognition in tracking, since objects usually suffer from serious variation over time. Here the dictionary of labeled samples is simply updated by replacing the old samples with the new samples of the same class. To enhance recognition rates and avoid false samples accumulation in the dictionary, we further give some updating rules:

- to avoid identity switch, the initial detected object sample is always stored in the dictionary.
- at the beginning of experiments, the dictionary is expanded by perturbing the initial sample with small Gaussian noise .
- the training samples of the same class share the same location information from the most recently updated sample.
- to avoid false recognition or novel object initialization, detected overlap is not updated.

Note that the popular dictionary learning algorithms for reducing recovery error [154] [155] are not employed here, since empirically the accurate sparse representation is not required for object classification.

7.3 Experiments

There are few benchmark videos for multi-object tracking, especially the videos with few overlapped scenes that background subtraction can deal with. For comparison, we start experiment with a classic video from PETS'09, which has been evaluated by two state-of-the-art works, ETHZ [156] and EPFL [157]. ETHZ implements a robust tracking-by-detection scheme by combing body detection and particle filter. EPFL attempts to explore object appearance from a global view with multiple calibrated cameras. To further verify our performance, we also evaluate proposed approach on some sequences from PETS'06. (For video results, please refer to `http://youtu.be/SLyABs_nJeg`.)

Multi-object tracking usually faces three challenges: object switch during overlapping, new object initialization and re-recognition of re-entering objects. In the following part, we will briefly introduce two videos and then discuss the results in terms of aforementioned challenges.

7.3.1 Database PETS'09.

This video with 795 frames is recorded in a campus at 7 fps from a high view point. Ten persons walk in and out of scene, and some of them are similar in color. So it is a challenge for recognition by appearance. In the following comparison, the persons are labeled by the sequence number corresponding to their entries into the scene. In our results, ten persons are labeled with a number, and their initial samples are displaced on the top of each frame, as Figure 7.4 shows. In ETHZ and in EPFL, they are discriminated separately with color-box and number.

Results. Figure 7.3 illustrates two examples about identity switch between objects of similar appearances. In fact, the proposed approach shows better performance for discriminating objects on the whole video. This mainly benefits from the features involving object center coordinates. In the proposed approach, one special case is that objects 4 and 5 are labeled together as object 4 for a long time due to overlap as shown in Figure 7.2(b). However, object 5 can be successfully recovered when they

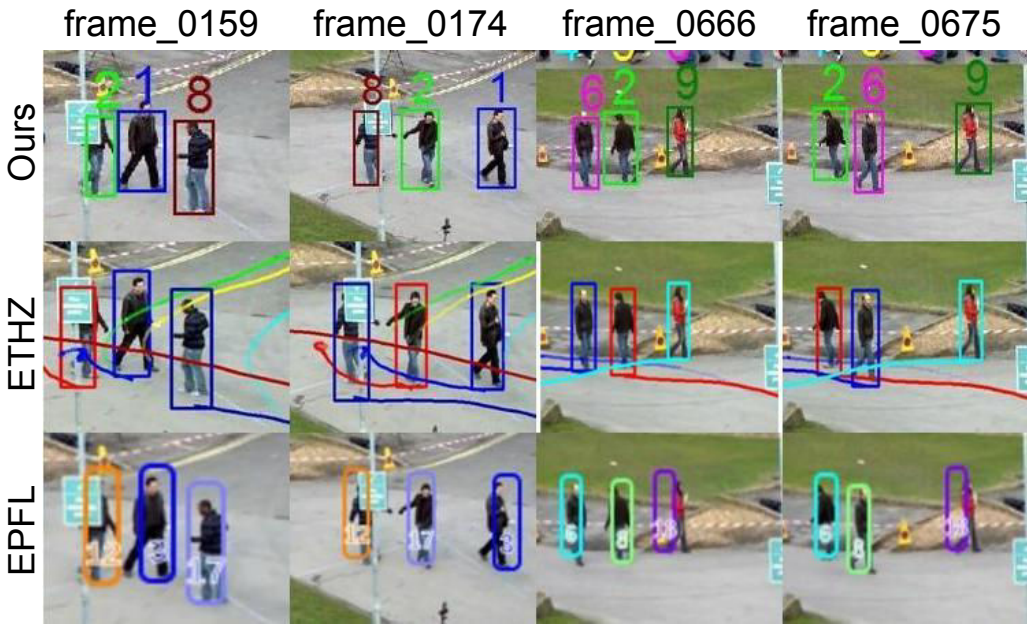


Figure 7.3: Two examples on identity switch caused by overlapping. EPFL switches the identities of *objects* 12 and 17 in the first two frames, and exchange *objects* 6 and 8 in the last two frames. Conversely, the proposed approach and ETHZ work well.

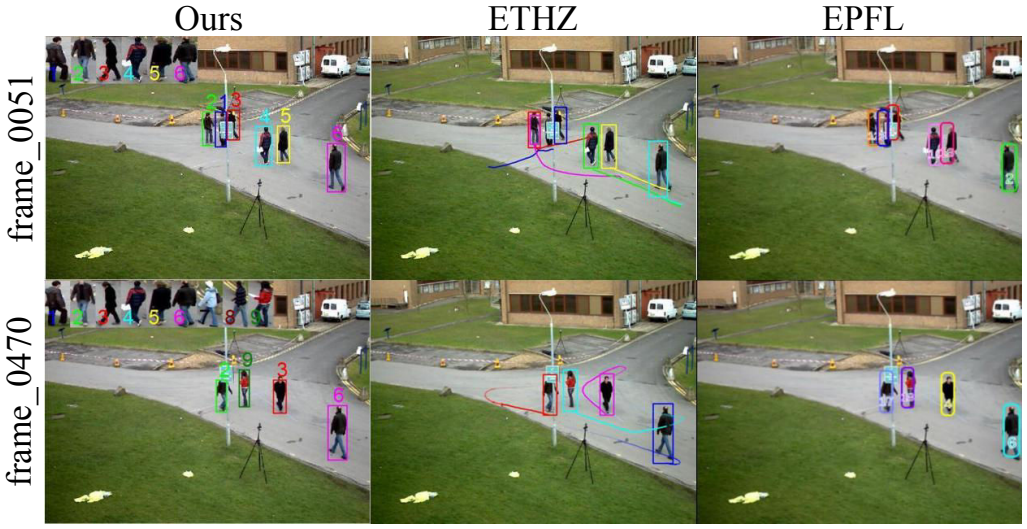


Figure 7.4: Examples on object initialization and re-recognition. In the frame_-0470, *object* 6 re-enters and *object* 9 first enters (referring our labels). *0bject* 6 is not recovered in ETHZ and EPFL. And *object* 9 is incorrectly initialized to *object* 6 in ETHZ. In contrast, the proposed approach performs well on above two cases.

separate. In Figure 7.4, we give one example about object re-recognition and novel object initialization, in which proposed approach works well while other two methods

Table 7.1: Correct occurrences for objects entering or re-entering in scene (PETS'09). Value 0 indicates false object initialization.

Objects	1	2	3	4	5	6	7	8	9	10
Num. of entries	4	2	2	2	2	2	1	1	2	1
Ours	**4**	**2**	**2**	1	1	**2**	**1**	**1**	**2**	**1**
ETHZ	4	2	2	1	1	1	1	1	0	0
EPFL	2	1	1	1	1	1	1	1	2	1

Figure 7.5: Tracking results for PETS'06. The bottom displays 11 objects that we initialize and track successfully. Our results have no identity switch or incorrect *novel* object initialization.

fail. In fact, proposed approach shows best performance for persistent identity tracking in the whole video, as confirmed in Table 7.1. Otherwise, it should be recalled that the proposed tracking process is not very fluent due to object detection failure as well as the lack of motion estimation.

7.3.2 Database PETS'06

We select a relatively crowded scene from S7.T6.B4 (frame _01685 to frame_-01985), in which 11 objects suffer from serious size variation and illumination variation, and some of them also share similar color. For example, *objects* 4, 5 and 6 are hard to be distinguished by naked eyes when they walk away.

Results. As Figure 7.5 shows, the proposed approach successfully detects and initializes these 11 objects. Furthermore, there is no identity switch caused by occlusion or incorrect object initialization. This result further proves the robustness

of sparse representation for multi-object classification.

7.4 Conclusion

This chapter has shown preliminary but promising results on the application of sparse representation to multiple objects tracking. By exploiting a simple tracking-by-detection scheme, sparse representation even presents better recognition performance than state-of-the-art. In the future, the simple tracking scheme can be further improved. For instance, the background will avoid being falsely detected as novel object, if the background samples are incorporated to the dictionary of sparse representation.

8 Conclusion

Dimension reduction is an important research direction in diverse research areas involving the process of high-dimensional data. In this thesis, we have studied three popular dimension reduction techniques and presented some significant results.

First, we have deterministically proposed and constructed the optimal binary compressed sensing matrix, which is undoubtedly a groundbreaking result for the application of compressed sensing. Importantly, it is expect to present comparable or even better performance than Gaussian random matrices for most possible distributions of sparse signals. The proposed construction method can also be borrowed to simply construct the optimal *random* ternary matrix through assigning the nonzero elements of optimal binary matrix to ± 1 with equal probability. To obtain more hardware-friendly structure, the binary matrix could also be constructed with quasi-cyclic matrix structure, which generally will lead to some loss of performance. Besides matrix construction, we also considered another interesting problem: the performance estimation of sensing matrices of high compression rates. Due to limited theoretical level, we have studied only the popular random Bernoulli matrix, which presents a performance floor as the compression rate increases. Luckily, the performance floor allows to be effectively estimated when we simply modify the traditional estimation method of sufficient condition towards the one of sufficient and necessary condition. Empirically, the random Gaussian matrix also holds a performance floor, which is hoped to be effectively estimated in the future.

For the second technique of random projection, the main challenge also arises from the matrix construction, which has been deeply studied in the past decade. The

thesis has restudied this problem from a novel point of view, that is from feature selection rather than from traditional distance preservation. Then a better classification performance has been theoretically derived with a random matrix much sparser than existing random projection matrices. Obviously, this result is competitive on both performance and complexity for the classification task involving random projection. In the future, the theoretically proposed random projection matrix needs more tests to expand to more application areas.

Unlike the two techniques above, the questions of sparse representation often occur along with specific applications. In this thesis, we considered this technique only in the context of visual object tracking. A simple but efficient single-object tracking scheme has been successfully proposed to decrease the computation load of sparse representation while without introducing obvious performance loss. The proposed tracking scheme could be further improved on the fluency of tracking trajectory if the affine transform of object is considered. Moreover, we also studied the application potential of sparse representation in the context of multi-object tracking, where sparse representation is simply used for object classification and achieves favorable performance. Maybe this result will help to develop an efficient multi-object tracking scheme in the future.

A

Résumé étendu en français

Introduction

Avec le développement de la science et de la technologie, les données de dimensions élevées sont de plus en plus populaires dans notre vie quotidienne. Par exemple, dans le domaine du traitement du signal et de la biostatistique, il est maintenant très commun de traiter des donnéesm telles que l'image, la vidéo ou l'ADN, dont les dimensions sont de l'ordre du million. Tout cela apporte souvent de grands défis pour le calcul et le stockage. Pour surmonter ce que l'on appelle la malédiction de la dimension, il faut souvent recourir à des techniques de réduction de la dimension, qui visent à projeter ces données de grande dimension sur un espace de relativement faible dimension tout en conservant les informations d'intérêt. Ce thème de réduction de la dimension a en effet reçu une attention considérable dans divers domaines de recherche tels que les statistiques, la bio-informatique, le traitement du signal, la vision par ordinateur, les machines à apprentissage etc. Dans la pratique, la notion de réduction de la dimension est principalement utilisée pour désigner les techniques de projection linéaires ou non linéaires visant des caractéristiques ou des échantillons de données, avec par exemple, les méthodes d'analyse de composantes principales (ACP) et également l'ACP basé sur le noyau, alors que d'un point de vue plus large, elle devrait également couvrir le problème de la compression de données. Dans cette thèse, nous allons concentrer notre attention sur trois techniques populaires de réduction de la dimension : l'acquisition parcimonieuse, la projection aléatoire et la représentation parcimonieuse, qui se voient toutes concernées par le

problème de la projection linéaire à travers une matrice. La contribution principale de cette thèse par rapport à ces techniques sera présentée brièvement dans les trois parties suivantes.

A.1 Acquisition parcimonieuse

A.1.1 Présentation du problème

L' acquisition parcimonieuse, ou échantillonnage parcimonieux, est une nouvelle technique visant à acquérir et récupérer des signaux réduits avec beaucoup moins de mesures que nécessite le taux de Nyquist classique [3] [4], ce qui offre évidemment une perspective remarquable dans la pratique, car tout naturellement les signaux font apparaître en général des structures clairsemées dans le domaine du temps, de l'espace ou des fréquences. Dans la pratique, la tâche majeure pour cette technique consiste à construire une matrice de réduction avec de bonnes performances. On sait que certaines matrices aléatoires, i.i.d avec des éléments extraits de certaines distributions bien connues, telles que la distribution gaussienne ou la distribution de Bernoulli, peuvent fournir bonnes performances de réduction et cela avec une forte probabilité. Cependant, en termes de mise en œuvre matérielle, il est évident que la matrice binaire zéro-un avec une structure déterministe est plus attrayante au niveau du calcul. Bien que certains travaux ont récemment été mis au point pour construire ce type de matrices, la matrice binaire optimale reste inconnue. Dans cette thèse, nous allons résoudre ce problème.

A.1.2 Fondamentaux

Notons $\mathbf{x} \in \mathbb{R}^n$ un vecteur avec $k \ll n$ éléments non nuls, appelé signal k-parcimonieux ou ayant une parcimonie k. Soit $\mathbf{A} \in \mathbb{R}^{m \times n}$ une matrice de réduction avec $m \ll n$, l'acquisition parcimonieuse dit que le vecteur de grande dimension $\mathbf{x}$ peut être parfaitement récupéré avec à la fois $\mathbf{A}$ et peu de mesures linéaires $\mathbf{y} = \mathbf{A}\mathbf{x}$. La solution peut être obtenue en résolvant un probleme de minimisation basé sur la

norme-ℓ_1

$$\min_{\hat{\mathbf{x}}} ||\hat{\mathbf{x}}||_1 \text{ s.t. } \mathbf{y} = \mathbf{A}\hat{\mathbf{x}}, \tag{A.1}$$

si le vecteur $\mathbf{x}$ est suffisamment clairsemé et tel que la propriété d'isométrie restreinte (RIP) puisse être satisfaite par la matrice $\mathbf{A}$.

Avant de détailler RIP, nous avons besoin de définir d'abord un paramètre dit d'isométrie restreinte constante (RIC) comme ci-dessous :

Definition A.1.1 (RIC)**.** *Le RIC d'une matrice $\mathbf{A} \in \mathbb{R}^{m\times n}$ est défini comme le plus petit $\delta_k \in (0, 1)$ tel que l'inégalité*

$$(1 - \delta_k)||\mathbf{x}||^2 \leq ||\mathbf{A}\mathbf{x}||^2 \leq (1 + \delta_k)||\mathbf{x}||^2 \tag{A.2}$$

est vérifié pour tous les signaux $\mathbf{x} \in \mathbb{R}^n$ k-parcimonieux.

Le RIP stipule que si δ_k de $\mathbf{A}$ est assez petit, le signal k-parcimonieux peut être bien récupéré avec un problème de minimization ℓ_1. Notons que $\delta_i \leq \delta_j$, si $i < j$ [32] [33]. Ainsi, dans la pratique, une matrice dont δ_k est relativement petit est préférable car elle offre une relativement grande parcimonie k. Notons également que la formule A.2 peut être réécrite comme

$$1 - \delta_k \leq \frac{\|\mathbf{A}_\psi \mathbf{x}\|^2}{\|\mathbf{x}\|^2} \leq 1 + \delta_k \tag{A.3}$$

qui est valable pour tout $|\psi| = k$ et $\mathbf{x} \in \mathbb{R}^{|\psi|}$, où $\mathbf{A}_\psi \in \mathbb{R}^{m\times|\psi|}$ est une sous-matrice de $\mathbf{A}$ avec des colonnes indiquées par $\psi \subseteq \{1, 2, ..., n\}$, et où $\mathbf{A}_\psi^T$ dénote le transposé de $\mathbf{A}_\psi$. Ceci implique que la solution pour δ_k peut être dérivée en limitant les valeurs propres extrêmes de $\mathbf{A}_\psi^T\mathbf{A}_\psi$,

A.1.3 Méthodes

En cherchant le meilleur RIP, pour une matrice de taille donnée, la répartition optimale de zéro-un est déterminée par le graphe biparti avec autant d'arêtes que possible, mais sans longueur de cycles égale à 4. Dans la partie suivante, nous présentons d'abord la notion de graphe biparti, puis proposons un algorithme efficace pour construire le graphe biparti souhaité.

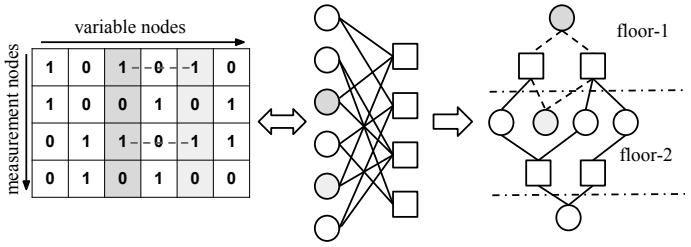


FIGURE A.1 – De gauche à droite : une matrice binaire, le graphe biparti correspondant et un sous-graphe étendu à partir d'un noeud de variable. Les noeuds de variable et les nœuds de mesure sont indiquées avec le cercle et le carré, respectivement. Si deux noeuds de variable partagent deux mêmes positions non nulles dans leurs colonnes respectives, ils forment un cycle le plus court de longueur 4 (lignes pointillées), comme les lignes pointillés montrées dans le sous-graphe.

Comme illustré dans la figure A.1, un graphe biparti est composé de deux classes de nœuds, que nous noterons les nœuds de variable et les nœuds de mesure. Au graphe est associé une matrice binaire, en laissant deux catégories de nœuds correspondant aux colonnes et lignes de la matrice binaire, respectivement. Les frontières entre deux catégories de nœuds sont déterminés par la position des éléments non nuls de la matrice binaire. Pour chaque nœud de variable, un sous-graphe avec plusieurs étages, comme illustré dans la figure A.1, peut être généré en traversant tous les nœuds connectés.

Le sous-graphe comprend souvent des chemins fermés, appelés les cycles. La longueur du cycle est mesurée avec le nombre d'arêtes, qui ne peut prendre que des valeurs paires supérieures ou égal à 4.

Parmi tous les sous-graphes, la longueur de cycle le plus court est définie comme la circonférence du graphe biparti. Empiriquement, quand le nombre d'arêtes augmente, les cycles plus courts sont inévitables, et la circonférence devient immédiatement plus petite. On note que, dans un sous-graphe comme sur la figure A.1, si le nœud de variable de racine est en outre relié à un nœud de mesure inclus dans le f-ième étage du graphe, où $f > 1$, les nouveaux cycles générés auront la longueur $2f$. Cette propriété nous permet de proposer un algorithme glouton pour construire le graphe biparti souhaité.

L'algorithme de construction proposé peut être simplement décrit avec un processus itératif. Dans chaque itération, chaque nœud de variable est autorisé à se

connecter tout au plus à un nœud de mesure. Le nœud de mesure est choisi de manière aléatoire dans le 3-ième étage du sous-graphe courant pour générer les cycles de longueur 6, si cet étage peut être réalisé par le sous-graphe courant. Sinon, si le sous-graphe ne contient pas tous les nœuds de mesure, un nœud de mesure en dehors du sous-graphe sera choisi au hasard pour éviter de générer des cycles de longueur 4. Cette procédure est répétée jusqu'à ce qu'aucun nœuds de variables n'aie des nœuds de mesure à mettre à jour.

A.1.4 Expérimentations

Dans la simulation, la matrice binaire optimale construite montre en effet une meilleure performance que les autres matrices binaires populaires. En outre, elle dépasse même les matrices aléatoires gaussiennes dans la plupart des cas.

A.2 Projection aléatoire

A.2.1 Présentation du problème

Similaire à l'acquisition parcimonieuse, la projection aléatoire est également une procédure simple de projection linéaire à travers une matrice sous-déterminée. Plutôt que pour une tâche de récupération de données, cette technique est seulement utilisée pour préserver les distances par paires de données de grande dimension dans un espace de faible dimension, de telle sorte qu'une tâche de classification puisse être menée. En fait, cette technique et ses applications connexes ont été largement étudiées dans la dernière décennie. Il est connu que certaines matrices creuses aléatoires $\{0, \pm 1\}$ peuvent préserver la distance avec une forte probabilité. Dans la pratique, ce serait mieux si l'on pouvait minimiser la densité de la matrice de projection aléatoire sans introduire une perte de performance sur la tâche de classification. Malheureusement, la performance de préservation de la distance tend à se dégrader avec une diminution de la densité de la matrice ce qui apparaît défavorable pour la classification. Toutefois, il convient de noter que la tâche de classification préfère maximiser la distance entre les différentes classes plutôt que

de préserver leurs distances. A partir de ce principe, la thèse présente ici la matrice de projection aléatoire la plus creuse, qui possède une performance en sélection de caractéristiques meilleure que les autres matrices aléatoires plus denses.

A.2.2 Fondamentaux

Pour faciliter la lecture, nous présentons d'abord quelques notations de base. Une matrice aléatoire est désignées par $\mathbf{R} \in \mathbb{R}^{k \times d}$, $k < d$. r_{ij} est utilisé pour représenter l'élément de $\mathbf{R}$ dans la i-ème ligne et j-ème colonne. $\mathbf{r} \in \mathbb{R}^d$ indique le vecteur ligne de $\mathbf{R}$. Compte tenu que cette étude se concentre principalement sur la classification binaire, nous définissons deux classes différentes d'échantillons avec $\mathbf{v} \in \mathbb{R}^d$ et $\mathbf{w} \in \mathbb{R}^d$.

Afin de préserver la distance par paires, la matrice de projection aléatoire doit satisfaire le lemme Johnson-Lindenstrauss (JL), qui est décrit comme ci-dessous.

Theorem A.2.1. *[114] Prenons une matrice aléatoire $\mathbf{R} \in \mathbb{R}^{k \times d}$, à chaque entrée r_{ij} choisie indépendemment d'une distribution symétrique par rapport au point d'origine avec $\mathbb{E}(r_{ij}^2) = 1$. Pour tous les vecteurs définis $\mathbf{v} \in \mathbb{R}^d$, faire $\mathbf{v}' = \frac{1}{\sqrt{k}}\mathbf{R}\mathbf{v}$.*

- *Supposons $B = \mathbb{E}(r_{ij}^4) < \infty$. Alors pour tout $\epsilon > 0$,*

$$\Pr(\|\mathbf{v}'\|^2 \leq (1-\epsilon)\|\mathbf{v}\|^2) \leq e^{-\frac{(\epsilon^2-\epsilon^3)k}{2(B+1)}} \tag{A.4}$$

- *Supposons $\exists L > 0$ pour tout nombre entier $m > 0$, $\mathbb{E}(r_{ij}^{2m}) \leq \frac{(2m)!}{2^m m!}L^{2m}$. Alors pour tout $\epsilon > 0$,*

$$\begin{aligned} \Pr(\|\mathbf{v}'\|^2 \geq (1+\epsilon)L^2\|\mathbf{v}\|^2) &\leq ((1+\epsilon)e^{-\epsilon})^{k/2} \\ &\leq e^{-(\epsilon^2-\epsilon^3)\frac{k}{4}} \end{aligned} \tag{A.5}$$

A partir du lemme ci-dessus, il peut être dérivé que les éléments d'une matrice de projection aléatoire devraient avoir une moyenne de $\mathbb{E}(r_{ij}) = 0$ et une variance de $\mathbb{E}(r_{ij}^2) = 1$. Cette condition sera utilisée dans la construction suivante des matrices.

A.2.3 Méthodes

Du point de vue de la sélection de caractéristiques, la projection aléatoire est prévue pour maximiser la différence entre deux échantillons arbitraires $\mathbf{v}$ et $\mathbf{w}$ de différentes classes. En ce sens, la recherche de la bonne projection aléatoire est équivalente à la recherche du vecteur ligne $\hat{\mathbf{r}}$ tel que

$$\hat{\mathbf{r}} = \underset{\mathbf{r}}{\text{argmax}}\{|\langle \mathbf{r}, \mathbf{z} \rangle|\}, \tag{A.6}$$

où $\mathbf{z} = \mathbf{v} - \mathbf{w}$. Dans cette thèse, il est démontré que le vecteur $\hat{\mathbf{r}}$ désiré sera obtenu si une seule paire d'éléments discriminatoires entre $\mathbf{v}$ et $\mathbf{w}$ est échantillonnée par $\hat{\mathbf{r}}$. Supposons que $\mathbf{z}$ suive la distribution :

$$z_i = \begin{cases} x & \text{avec une probabilité de } 1/2 \\ -x & \text{avec une probabilité de } 1/2 \end{cases} \tag{A.7}$$

où $x \in N(\mu, \sigma^2)$, μ est un nombre positif, et $\Pr(x > 0) = 1 - \epsilon$, $\epsilon = \Phi(-\frac{\mu}{\sigma})$ est un petit nombre positif. Nous pouvons ensuite dériver trois lemmes comme suit :

Theorem A.2.2. *Soit* $\mathbf{r} = [r_1, ..., r_d]$ *un vecteur aléatoire ayant* $1 \leq s \leq d$ *éléments non nuls prenant les valeurs* $\pm\sqrt{d/s}$ *de manière équiprobable, et* $\mathbf{z} = [z_1, ..., z_d]$ *avec des éléments égaux à* $\pm\mu$ *de manière équiprobable, où* μ *est une constante positive. En notant* $f(\mathbf{r}, \mathbf{z}) = |\langle \mathbf{r}, \mathbf{z} \rangle|$, *nous obtenons trois résultats concernant les valeurs attendues de* $f(r_i, z)$ *:*

1) $\mathbb{E}(f) = 2\mu\sqrt{\frac{d}{s}}\frac{1}{2^s}\lceil\frac{s}{2}\rceil C_s^{\lceil\frac{s}{2}\rceil}$;

2) $\mathbb{E}(f)|_{s=1} = \mu\sqrt{d} > \mathbb{E}(f)|_{s>1}$;

3) $\lim_{s\to\infty} \frac{1}{\sqrt{d}}\mathbb{E}(f) \to \mu\sqrt{\frac{2}{\pi}}$.

Theorem A.2.3. *Soit* $\mathbf{r} = [r_1, ..., r_d]$ *aléatoire ayant* $1 \leq s \leq d$ *éléments non nuls prenant les valeurs* $\pm\sqrt{d/s}$ *de manière équiprobable, et* $\mathbf{z} = [z_1, ..., z_d]$ *avec des éléments distribués selon la formule* (**??**). *En notant* $f(\mathbf{r}, \mathbf{z}) = |\langle \mathbf{r}, \mathbf{z} \rangle|$, *on obtient que :*

$$\mathbb{E}(f)|_{s=1} > \mathbb{E}(f)|_{s>1}$$

si $(\frac{9}{8})^{\frac{3}{2}}[\sqrt{\frac{2}{\pi}} + (1 + \frac{\sqrt{3}}{4})\frac{2}{\pi}(\frac{\mu}{\sigma})^{-1}] + 2\Phi(-\frac{\mu}{\sigma}) \leq 1.$

Theorem A.2.4. *Soit* $\mathbf{r} = [r_1, ..., r_d]$ *ayant ses éléments i.i.d suivant une* $N(0, 1)$*, et* $\mathbf{z} = [z_1, ..., z_d]$ *avec des éléments égaux à* $\pm\mu$ *de manière équiprobable, où* μ *est une constante positive. En notant* $f(\mathbf{r}, \mathbf{z}) = |\langle \mathbf{r}, \mathbf{z} \rangle|$*, on obtient* $\mathbb{E}(f) = \mu\sqrt{\frac{2d}{\pi}}$*.*

Deux conclusions importantes sont alors obtenues :

- Selon Théorèmes A.2.2 and A.2.3, le vecteur ligne $\hat{\mathbf{r}}$ de la formule (A.6) devrait échantillonner un seul élément caractéristique.
- Selon Théorème A.2.4, le vecteur désiré décrit ci-dessus sera plus efficace que des matrices aléatoires gaussiennes.

En pratique, la matrice aléatoire satisfaisant à la condition d'échantillonnage désirée ci-dessus est difficile à satisfaire, car le nombre et l'emplacement des éléments caractéristiques sont généralement inconnus. Mais nous pouvons proposer un type de matrices avec un seul élément non nul par colonne, qui satisfait à la condition ci-dessus avec une forte probabilité. La matrice aléatoire proposée présente deux avantages évidents :

- Elle est jusqu'ici la matrice de projection aléatoire la plus creuse.
- Elle surclasse les autres matrices plus denses, si la dimension de projection n'est pas beaucoup plus petite que le nombre d'éléments caractéristiques.

A.2.4 Expérimentation

L'avantage de la performance de la matrice proposée est confirmé par de nombreuses expériences de classification sur des images, des textes et des ADN.

A.3 Représentation parcimonieuse

A.3.1 Présentation du problème

La représentation parcimonieuse assure qu'un vecteur d'intérêt peut être représenté ou approché par une combinaison linéaire de quelques colonnes d'une matrice sous-déterminée (souvent appelée dictionnaire) [11] [12]. Mathématiquement, c'est un

problème d'inversion matricielle avec contrainte de parcimonie, et donc il partage les mêmes solutions algorithmiques parcimonieuses qu'avec l'acquisition parcimonieuse. Cependant à la différence de l'acquisition parcimonieuse, l'exacte solution parcimonieuse n'existe généralement pas en représentation parcimonieuse. Cela car la relation linéaire parcimonieuse entre le vecteur d'intérêt et le dictionnaire ne peut habituellement être assurée. Dans le domaine de l'apprentissage automatique et de la reconnaissance des formes, la représentation parcimonieuse est souvent appliquée pour mesurer la similarité entre le vecteur d'intérêt et le dictionnaire, ou pour identifier les éléments du dictionnaire qui sont semblables au vecteur d'intérêt. Ainsi la conception du dictionnaire n'a pas de critère uniforme, et dépend souvent de la spécifiité de l'application. Une application intéressante correspond au suivi d'objet, qui a récemment été intensivement étudié. Il faut remarquer que les travaux actuels se concentrent principalement sur l'amélioration des performances, alors que la charge de calcul introduite par la représentation parcimonieuse reste ignorée. Par contre, dans cette thèse, l'application de la représentation parcimonieuse au suivi d'objet sera explorée.

A.3.2 Fondamentaux

L'application de la représentation parcimonieuse au suivi d'objet est brièvement présentée dans la suite. Soit $\mathbf{y} \in \mathbb{R}^{m\times 1}$ le vecteur qui désigne un objet d'intérêt, et la matrice $\mathbf{D} = [D_{G_1}, D_{G_2}, \ldots, D_{G_N}] \in \mathbb{R}^{m\times n}$ qui est un dictionnaire constitué de N classes d'échantillons, où la i-ième sous-matrice $D_{G_i} = [D_{i_1}, D_{i_2}, \ldots, D_{i_{n_i}}]$ comprend n_i échantillons avec $\sum_{i=1}^{N} n_i = n$. Par la suite on suppose idéalement que $\mathbf{y}$ peut être représenté approximativement par une combinaison linéaire de plusieurs éléments de $\mathbf{D}$, c'est-à-dire

$$y = \mathbf{D}\beta + \epsilon \tag{A.8}$$

Où β est supposé détenir au plus $k \ll n$ termes positifs non nuls ; et ϵ est l'erreur tolérée. Ensuite, le vecteur caractéristique $\mathbf{y}$ est considéré comme proche du sous-

espace des éléments choisis. En d'autres termes, il peut être identifié comme la classe

$$\hat{i} = \underset{i}{\operatorname{argmax}}\{\delta_i(\beta)|1 \leq i \leq N\}, \tag{A.9}$$

Où $\delta_i(\beta)$ est la fonction qui regroup les éléments de β correspondant à D_{G_i}.

A.3.3 Méthodes

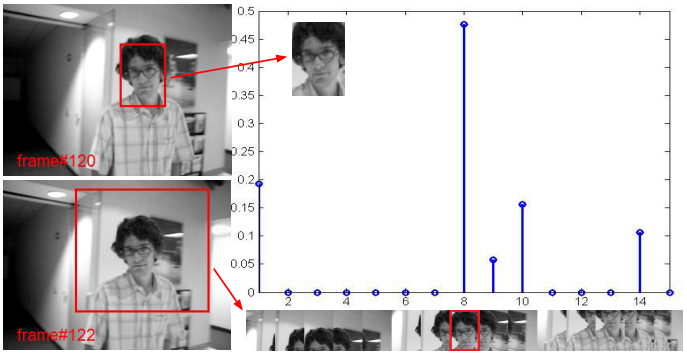


FIGURE A.2 – L'objet labélisé de l'image précédente peut être rapproché parcimonieusement par un dictionnaire consistant de parcelles locales se recouvrant dans certaines régions de l'image actuelle. La parcelle locale correspondant au plus grand coefficient est sujette à indiquer la position estimée de l'objet.

Dans cette thèse, nous proposons un schéma simple mais efficace de suivi qui nécessite de procéder à la représentation parcimonieuse seulement deux fois, si la caractéristique d'objet est suffisamment robuste. Le schéma proposé du suivi se compose de deux étapes de représentation parcimonieuse. Dans la première étape, le vecteur d'intérêt $\mathbf{y}$ est l'échantillon de l'objet connu détecté dans l'image précédente, et le dictionnaire $\mathbf{D}$ contient des candidats de l'échantillon dans l'image actuelle. Ensuite, l'échantillon candidat avec le plus large coefficient est choisi comme étant similaire à l'objet d'intérêt, qui est représenté dans Figure A.2. Dans la deuxième étape, l'échantillon sélectionné est vérifié en plus avec la représentation parcimonieuse, dans laquelle $\mathbf{y}$ représente l'échantillon sélectionné, et $\mathbf{D}$ comprend deux classes des échantillons marqués, des échantillons d'objet nommé, et des échantillons de référence. L'échantillon sélectionné est considéré comme l'objet suivi, si les coefficients parcimonieux se concentrent sur les échantillons d'objets.

A.3.4 Expérimentations

La méthode proposée est comparée avec d'autres méthodes plus complexes basées sur la représentation parcimonieuse. Le résultat montre une performance comparable ou supérieure aux autres approches.

Bibliography

[1] Christopher J. C. Burges. Dimension reduction: A guided tour. *Foundations and Trends in Machine Learning*, 2(4):275–365, 2010.

[2] Surya Ganguli and Haim Sompolinsky. Compressed sensing, sparsity, and dimensionality in neuronal information processing and data analysis. *Annual Review of Neuroscience*, 35(1):485–508, 2012.

[3] E.J. Candes and M.B. Wakin. An introduction to compressive sampling. *IEEE Signal Processing Magazine*, 25(2):21–30, March 2008.

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

[5] W. Lu, K. Kpalma, and J. Ronsin. Sparse binary matrices of LDPC codes for compressed sensing. In *Data Compression Conference (DCC)*, page 405, april 2012.

[6] Weizhi Lu, Kidiyo Kpalma, and Joseph Ronsin. Near-optimal binary compressed sensing matrix. 2013.

[7] W. B. Johnson and J. Lindenstrauss. Extensions of Lipschitz mappings into a Hilbert space. *Contemp. Math.*, 26:189–206, 1984.

[8] D. Achlioptas. Database-friendly random projections: Johnson–Lindenstrauss with binary coins. *J. Comput. Syst. Sci.*, 66(4):671–687, 2003.

[9] R. Baraniuk, M. Davenport, R. DeVore, and M. Wakin. A simple proof of the restricted isometry property for random matrices. *Constructive Approximation*, 28(3):253–263, 2008.

[10] Weizhi Lu, Weiyu Li, Kidiyo Kpalma, and Joseph Ronsin. Sparse matrix based random projection for catogarization, 2014. arXiv:1312.3522.

[11] Julien Mairall. *Sparse Coding for Machine Learning, Image Processing and Computer Vision*. Phd thesis, ENS Cachan, Paris, 2010.

[12] Michael Elad. *Sparse and Redundant Representations*. Springer, 2010.

[13] Weizhi Lu, Cong Bai, K. Kpalma, and J. Ronsin. Multi-object tracking using sparse representation. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2312–2316, May 2013.

[14] A. Cohen, W. Dahmen, and R. Devore. Compressed sensing and best k-term approximation. *Journal of The American Mathematical Society*, 22:211–231, 2009.

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

[16] E.J. Candes and T. Tao. Decoding by linear programming. *IEEE Transactions on Information Theory*, 51(12):4203 – 4215, dec. 2005.

[17] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambrige university press, March 2004.

[18] David L. Donoho. Neighborly polytopes and sparse solutions of underdetermined linear equations. Technical report, 2005-04, Stanford Univ., 2005.

[19] David L. Donoho and Jared Tanner. Neighborliness of randomly projected simplices in high dimensions. 102(27):9452–9457, 2005.

[20] David L. Donoho and Jared Tanner. Sparse nonnegative solution of underdetermined linear equations by linear programming. 102(27):9446–9451, 2005.

[21] David L. Donoho. High-dimensional centrally symmetric polytopes with neighborliness proportional to dimension. *Discrete & Computational Geometry*, 35(4):617–652, 2006.

[22] David L. Donoho and Jared Tanner. Counting faces of randomly-projected polytopes when the projection radically lowers dimension. *J. of the AMS*, pages 1–53, 2009.

[23] D.L. Donoho and J. Tanner. Precise undersampling theorems. *Proceedings of the IEEE*, 98(6):913–924, 2010.

[24] David L. Donoho and Michael Elad. Optimally sparse representation in general (nonorthogonal) dictionaries via ℓ^1 minimization. *Proceedings of the National Academy of Sciences*, 100(5):2197–2202, 2003.

[25] D.L. Donoho and X. Huo. Uncertainty principles and ideal atomic decomposition. *IEEE Transactions on Information Theory*, 47(7):2845–2862, Nov 2011.

[26] R. Gribonval and M. Nielsen. Sparse representations in unions of bases. *IEEE Transactions on Information Theory*, 49(12):3320–3325, Dec 2003.

[27] J.A. Tropp. Greed is good: algorithmic results for sparse approximation. *IEEE Transactions on Information Theory*, 50(10):2231–2242, Oct 2004.

[28] R. Horn and C. Johnson. *Matrix Analysis*. Cambrige university press, 1985.

[29] D. Donoho, M. Elad, and V. Temlyakov. Stable recovery of sparse overcomplete representations in the presence of noise. *IEEE Trans. Inf. Theory*, 52(1):6–18, 2006.

[30] E. Candes, J. Romberg, and T. Tao. Stable signal recovery from incomplete and inaccurate measurements. *Communications on Pure and Applied Mathematics*, 59:1207–1223, 2006.

[31] E.J. Candes and T. Tao. Near-optimal signal recovery from random projections: Universal encoding strategies? *IEEE Transactions on Information Theory*, 52(12):5406 –5425, Dec. 2006.

[32] CoSaMP: Iterative signal recovery from incomplete and inaccurate samples. *Applied and Computational Harmonic Analysis*, 26(3):301 – 321, 2009.

[33] Yonina C. Eldar and HaifaGitta Kutyniok, editors. *Compressive Sensing: Theory and Applications*. Campridge University Press, June 2012.

[34] E. Candes. The restricted isometry property and its implications for compressed sensing. *Comptes Rendus Mathematique*, 346(9-10):589–592, May 2008.

[35] Simon Foucart and Ming-Jun Lai. Sparsest solutions of underdetermined linear systems via l_q-minimization for $0 \leq q < 1$. *Applied and Computational Harmonic Analysis*, 26(3):395 – 407, 2009.

[36] S. Foucart. A note on guaranteed sparse recovery via ℓ_1-minimization. *Applied and Computational Harmonic Analysis*, 29(1):97 – 103, 2010.

[37] T.T. Cai, Lie Wang, and Guangwu Xu. New bounds for restricted isometry constants. *IEEE Transactions on Information Theory*, 56(9):4388–4394, 2010.

[38] A. M. Tillmann and M. E. Pfetsch. The Computational Complexity of the Restricted Isometry Property, the Nullspace Property, and Related Concepts in Compressed Sensing. *IEEE Transactions on Information Theory*, February 2014.

[39] A.S. Bandeira, E. Dobriban, D.G. Mixon, and W.F. Sawin. Certifying the restricted isometry property is hard. *IEEE Transactions on Information Theory*, 59(6):3448–3450, June 2013.

[40] Jeffrey D. Blanchard, Coralia Cartis, and Jared Tanner. Compressed sensing: How sharp is the restricted isometry property? *SIAM Rev.*, 53(1):105–125, February 2011.

[41] Eugene P. Wigner. On the distribution of the roots of certain symmetric matrices. *The Annals of Mathematics*, 67(2):325 – 327, 1958.

[42] S. Gurevich and R. Hadani. The statistical restricted isometry property and the wigner semicircle distribution of incoherent dictionaries. In *arXiv:0812.2602*, Dec. 2008.

[43] X. Zhan. Extremal eigenvalues of real symmetric matrices with entries in an interval. *SIAM Journal on Matrix Analysis and Applications*, 27(3):851–860, 2005.

[44] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B*, 58(1):267–288, 1996.

[45] R. Tyrrell Rockafellar. *Convex Analysis*. Princeton university press, 1997.

[46] Yonina C. Eldar. Generalized sure for exponential families: Applications to regularization. *IEEE Trans. Signal Processing*, 57(2):471–481, February 2009.

[47] N.P. Galatsanos and A.K. Katsaggelos. Methods for choosing the regularization parameter and estimating the noise variance in image restoration and their relation. *IEEE Transactions on Image Processing*, 1(3):322–336, Jul 1992.

[48] Gene H. Golub, Michael Heath, and Grace Wahba. Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics*, 21(2):215–223, 1979.

[49] Peng Zhao and Bin Yu. Stagewise lasso. *J. Mach. Learn. Res.*, 8:2701–2726, December 2007.

[50] M. A T Figueiredo, R.D. Nowak, and S.J. Wright. Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems. *IEEE Journal of Selected Topics in Signal Processing*, 1(4):586–597, Dec 2007.

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

[52] D.M. Malioutov, M. Cetin, and A.S. Willsky. Homotopy continuation for sparse signal representation. In *IEEE International Conference on Acoustics, Speech, and Signal Processing,*, volume 5, pages 733–736 Vol. 5, March 2005.

[53] B. Efron, T. Hastie, and R. Tibshirani. Least angle regression. *Annals of Statistics*, 32:407–499, 2004.

[54] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

[55] Stephen Becker, Jérôme Bobin, and Emmanuel J. Candès. Nesta: A fast and accurate first-order method for sparse recovery. *SIAM J. Img. Sci.*, 4(1):1–39, Jan. 2011.

[56] Stephen Becker, Emmanuel J. Candès, and Michael C. Grant. Templates for convex cone problems with applications to sparse signal recovery. *Math. Program. Comput.*, 3(3):165–218, 2011.

[57] E. van den Berg and M. P. Friedlander. Probing the pareto frontier for basis pursuit solutions. *SIAM Journal on Scientific Computing*, 31(2):890–912, 2008.

[58] Y.C. Pati, R. Rezaiifar, and P.S. Krishnaprasad. Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition. In *Conference Record of The Twenty-Seventh Asilomar Conference on Signals, Systems and Computers*, pages 40 –44 vol.1, nov 1993.

[59] J. A. Tropp and A. C. Gilbert. Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Transaction on Information Theory*, 53:4655–4666, 2007.

[60] J.A. Tropp and S.J. Wright. Computational methods for sparse solution of linear inverse problems. *Proceedings of the IEEE*, 98(6):948–958, June 2010.

[61] Allen Yang, Arvind Ganesh, Shankar Sastry, and Yi Ma. Fast ℓ_1-minimization algorithms and an application in robust face recognition: A review. Technical report, EECS Department, University of California, Berkeley, Feb 2010.

[62] Graeme Pope. *Compressive Sensing: A Summary of Reconstruction Algorithms*. Master thesis, Department of Computer Science, ETH, Zurich, 2009.

[63] Philip Breen. *Algorithms for Sparse Approximation*. Year 4 Project , School of Mathematics University of Edinburgh, 2009.

[64] Simon Foucart and Holger Rauhut. *A Mathematical Introduction to Compressive Sensing*. Springer New York, 2013.

[65] Massimo Fornasier, editor. *Theoretical Foundations and Numerical Methods for Sparse Recovery*. Berlin, Boston: De Gruyter, Web. Retrieved 5 Mar. 2014.

[66] Holger Rauhut. On the impossibility of uniform sparse reconstruction using greedy methods. *Sampl. Theory Signal Image Process*, 2008.

[67] Wei Dai and O. Milenkovic. Subspace pursuit for compressive sensing signal reconstruction. *IEEE Transactions on Information Theory*, 55(5):2230–2249, 2009.

[68] D. Needell and R. Vershynin. Signal recovery from incomplete and inaccurate measurements via regularized orthogonal matching pursuit. *IEEE Journal of Selected Topics in Signal Processing*, 4(2):310–316, April 2010.

[69] D.L. Donoho, Y. Tsaig, I. Drori, and J-L Starck. Sparse solution of underdetermined systems of linear equations by stagewise orthogonal matching pursuit. *IEEE Transactions on Information Theory*, 58(2):1094–1121, Feb 2012.

[70] T.T. Do, Lu Gan, N. Nguyen, and T.D. Tran. Sparsity adaptive matching pursuit algorithm for practical compressed sensing. In *42nd Asilomar Conference on Signals, Systems and Computers*, pages 581–587, Oct 2008.

[71] Michael B. Wakin, Jason N. Laska, Marco F. Duarte, Dror Baron, Shriram Sarvotham, Dharmpal Takhar, Kevin F. Kelly, and Richard G. Baraniuk. Compressive imaging for video representation and coding. In *Proceedings of Picture Coding Symposium (PCS)*, 2006.

[72] M.F. Duarte, M.A. Davenport, D. Takhar, J.N. Laska, Ting Sun, K.F. Kelly, and R.G. Baraniuk. Single-pixel imaging via compressive sampling. *IEEE Signal Processing Magazine*, 25(2):83–91, March 2008.

[73] A. Stern and B. Javidi. Random projections imaging with extended space-bandwidth product. *Journal of Display Technology*, 3(3):315–320, Sept 2007.

[74] Hao Fang. *Parallel Sampling and Reconstruction with Permutation in Multidimensional Compressed Sensing*. Masther Thesis, University of Alberta, Canada, Fall, 2013.

[75] M. Lustig, D.L. Donoho, J.M. Santos, and J.M. Pauly. Compressed sensing mri. *IEEE Signal Processing Magazine*, 25(2):72–82, March 2008.

[76] Shreyas S. Vasanawala, Marcus T. Alley, Brian A. Hargreaves, Richard A. Barth, John M. Pauly, and Michael Lustig. Improved pediatric mr imaging with compressed sensing. *Radiology*, 256(2):607–616, 2010.

[77] M. Mishali, Y.C. Eldar, and A.J. Elron. Xampling: Signal acquisition and processing in union of subspaces. *IEEE Transactions on Signal Processing*, 59(10):4719–4734, Oct 2011.

[78] Khanh Do Ba, Piotr Indyk, Eric Price, and David P. Woodruff. Lower bounds for sparse recovery. In *Proceedings of the Twenty-first Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '10, pages 1190–1197, Philadelphia, PA, USA, 2010. Society for Industrial and Applied Mathematics.

[79] R. Berinde and P. Indyk. Sparse recovery using sparse random matrices. *MIT-CSAIL Technical Report*, 2008.

[80] A. Amini and F. Marvasti. Deterministic construction of binary, bipolar, and ternary compressed sensing matrices. *IEEE Transactions on Information Theory*, 57(4):2360 –2370, april 2011.

[81] Nir Ailon and Edo Liberty. Fast dimension reduction using rademacher series on dual bch codes. In *Proceedings of the nineteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 1–9, 2008.

[82] M. Akcakaya and V. Tarokh. A frame construction and a universal distortion bound for sparse representations. *IEEE Transactions on Signal Processing*, 56(6):2443 –2450, june 2008.

[83] S.D. Howard, A.R. Calderbank, and S.J. Searle. A fast reconstruction algorithm for deterministic compressive sensing using second order reed-muller codes. In *42nd Annual Conference on Information Sciences and Systems (CISS 2008)*, pages 11 –15, march 2008.

[84] R. Calderbank, S. Howard, and Sina Jafarpour. Sparse reconstruction via the reed-muller sieve. In *2010 IEEE International Symposium on Information Theory Proceedings (ISIT)*, pages 1973–1977, 2010.

[85] R. Calderbank, S. Howard, and Sina Jafarpour. Construction of a large class of deterministic sensing matrices that satisfy a statistical isometry property. *IEEE Journal of Selected Topics in Signal Processing*, 4(2):358–374, 2010.

[86] H.V. Pham, Wei Dai, and O. Milenkovic. Sublinear compressive sensing reconstruction via belief propagation decoding. In *IEEE International Symposium on Information Theory*, pages 674–678, 2009.

[87] A. Barg and A. Mazumdar. Small ensembles of sampling matrices constructed from coding theory. In *IEEE International Symposium on Information Theory Proceedings (ISIT),*, pages 1963–1967, 2010.

[88] R. A. DeVore. Deterministic constructions of compressed sensing matrices. *Journal of Complexity*, 23(4-6):918 – 925, 2007.

[89] A. Khajehnejad, A.S. Tehrani, A.G. Dimakis, and B. Hassibi. Explicit matrices for sparse approximation. In *2011 IEEE International Symposium on Information Theory Proceedings (ISIT),*, pages 469–473, July 2011.

[90] W. Xu and B. Hassibi. Efficient compressive sensing with deterministic guarantees using expander graphs. In *IEEE Information Theory Workshop, ITW '07*, pages 414 –419, sept. 2007.

[91] Sina Jafarpour, Weiyu Xu, B. Hassibi, and R. Calderbank. Efficient and robust compressed sensing using optimized expander graphs. *IEEE Transactions on Information Theory*, 55(9):4299–4308, 2009.

[92] P. Gilbert, Aw; Indyk. Sparse recovery using sparse matrices. *Proceedings of the IEEE*, 98(6):937–947, 2010.

[93] A.G. Dimakis, R. Smarandache, and P.O. Vontobel. LDPC codes for compressed sensing. *IEEE Transactions on Information Theory*, 58(5):3093 –3114, may 2012.

[94] Xin-Ji Liu and Shu-Tao Xia. Reconstruction guarantee analysis of binary measurement matrices based on girth. In *IEEE International Symposium on Information Theory Proceedings (ISIT)*, pages 474–478, July 2013.

[95] X.-Y. Hu, E. Eleftheriou, and D.M. Arnold. Regular and irregular progressive edge-growth Tanner graphs. *IEEE Transactions on Information Theory*, 51(1):386 –398, Jan. 2005.

[96] Assaf Naor and Jacques Verstraëte. A note on bipartite graphs without 2k-cycles. *Comb. Probab. Comput.*, 14(5-6):845–849, November 2005.

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

[98] Wei Dai and O. Milenkovic. Subspace pursuit for compressive sensing signal reconstruction. *IEEE Transactions on Information Theory*, 55(5):2230–2249, 2009.

[99] David L. Donoho, Arian Maleki, and Andrea Montanari. Message-passing algorithms for compressed sensing. *Proceedings of the National Academy of Sciences*, 106(45):18914–18919, 2009.

[100] Zongwang Li and B.V.K.V. Kumar. A class of good quasi-cyclic low-density parity check codes based on progressive edge growth graph. In *Conference Record of the Thirty-Eighth Asilomar Conference on Signals, Systems and Computers, 2004.*, volume 2, pages 1990–1994 Vol.2, 2004.

[101] weizhi lu, kidiyo kpalma, and joseph ronsin. Semi-deterministic ternary matrix for compressed sensing. In *submitted to EUSIPCO*, 2014.

[102] Linh V. Tran, Van H. Vu, and Ke Wang. Sparse random graphs: Eigenvalues and eigenvectors. *Random Structures & Algorithms*, 42(1):110–134, 2013.

[103] Z. Füredi and J. Komlós. The eigenvalues of random symmetric matrices. *Combinatorica*, 1:233–241, 1981.

[104] Terence Tao and Van Vu. Random matrices: Universality of local eigenvalue statistics. *Acta Mathematica*, 206:127–204, 2011.

[105] Yoshiyuki Kabashima, Hisanao Takahashi, and Osamu Watanabe. Cavity approach to the first eigenvalue problem in a family of symmetric random sparse matrices. *Journal of Physics: Conference Series*, 233(1):012001.

[106] N. Goel, G. Bebis, and A. Nefian. Face recognition experiments with random projection. *in Proceedings of SPIE, Bellingham, WA*, pages 426–437, 2005.

[107] R. J. Durrant and A. Kaban. Random projections as regularizers: Learning a linear discriminant ensemble from fewer observations than data dimensions. *Proceedings of the 5th Asian Conference on Machine Learning (ACML 2013). JMLR W&CP*, 29:17–32, 2013.

[108] Robert Calderbank, Sina Jafarpour, and Robert Schapire. Compressed learning: Universal sparse dimensionality reduction and learning in the measurement domain. *Technical Report*, 2009.

[109] P. Indyk and R. Motwani. Approximate nearest neighbors: Towards removing the curse of dimensionality. *in Proceedings of the 30th Annual ACM Symposium on Theory of Computing*, pages 604–613, 1998.

[110] P. Li, T. J. Hastie, and K. W. Church. Very sparse random projections. *in Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2006.

[111] A. Dasgupta, R. Kumar, and T. Sarlos. A sparse Johnson–Lindenstrauss transform. *in Proceedings of the 42nd ACM Symposium on Theory of Computing*, 2010.

[112] S. Dasgupta and A. Gupta. An elementary proof of the Johnson–Lindenstrauss lemma. *Technical Report, UC Berkeley*, (99–006), 1999.

[113] J. Matoušek. On variants of the Johnson–Lindenstrauss lemma. *Random Struct. Algorithms*, 33(2):142–156, 2008.

[114] R. Arriaga and S. Vempala. An algorithmic theory of learning: Robust concepts and random projection. *Journal of Machine Learning*, 63(2):161–182, 2006.

[115] X. Z. Fern and C. E. Brodley. Random projection for high dimensional data clustering: A cluster ensemble approach. *in Proceedings of the 20th International Conference on Machine Learning*, 2003.

[116] A. Martinez and R. Benavente. The AR face database. *Technical Report 24, CVC*, 1998.

[117] A. Martinez. PCA versus LDA. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(2):228–233, 2001.

[118] A. Georghiades, P. Belhumeur, and D. Kriegman. From few to many: Illumination cone models for face recognition under variable lighting and pose. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(6):643–660, 2001.

[119] K. Lee, J. Ho, and D. Kriegman. Acquiring linear subspaces for face recognition under variable lighting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(5):684–698, 2005.

[120] P. J. Phillips, H.Wechsler, and P. Rauss. The FERET database and evaluation procedure for face-recognition algorithms. *Image and Vision Computing*, 16(5):295–306, 1998.

[121] A. V. Nefian and M. H. Hayes. Maximum likelihood training of the embedded HMM for face detection and recognition. *IEEE International Conference on Image Processing*, 2000.

[122] F. Samaria and A. Harter. Parameterisation of a stochastic model for human face identification. *In 2nd IEEE Workshop on Applications of Computer Vision, Sarasota, FL*, 1994.

[123] U. Alon, N. Barkai, D. A. Notterman, K. Gish, S. Ybarra, D. Mack, and A. J. Levine. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proceedings of the National Academy of Sciences*, 96(12):6745–6750, 1999.

[124] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, C. D. Bloomfield, and E. S. Lander. Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science*, 286(5439):531–537, 1999.

[125] David G. Beer, Sharon L. Kardia, Chiang-Ching C. Huang, Thomas J. Giordano, Albert M. Levin, David E. Misek, Lin Lin, Guoan Chen, Tarek G. Gharib, Dafydd G. Thomas, Michelle L. Lizyness, Rork Kuick, Satoru Hayasaka, Jeremy M. Taylor, Mark D. Iannettoni, Mark B. Orringer, and Samir Hanash. Gene-expression profiles predict survival of patients with lung adenocarcinoma. *Nature medicine*, 8(8):816–824, August 2002.

[126] Deng Cai, Xuanhui Wang, and Xiaofei He. Probabilistic dyadic data analysis with local and global consistency. In *Proceedings of the 26th Annual International Conference on Machine Learning (ICML'09)*, pages 105–112, 2009.

[127] N. G. de Bruijn. *Asymptotic methods in analysis*. Dover, March 1981.

[128] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma. Robust face recognition via sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(2):210–227, 2009.

[129] Qing Wang, Feng Chen, Wenli Xu, and Ming-Hsuan Yang. Online discriminative object tracking with local sparse representation. In *IEEE Workshop on Application of Computer Vision (WACV)*, 2012.

[130] Xue Mei, Haibin Ling, Yi Wu, E. Blasch, and Li Bai. Minimum error bounded efficient l_1 tracker with occlusion detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[131] Qing Wang, Feng Chen, Wenli Xu, and Ming-Hsuan Yang. Object tracking via partial least squares analysis. *IEEE Transactions on Image Processing*, 21(10):4454–4465, 2012.

[132] Xue Mei and Haibin Ling. Robust visual tracking using l_1 minimization. In *IEEE International Conference on Computer Vision*, 2009.

[133] Baiyang Liu, Lin Yang, Junzhou Huang, Peter Meer, Leiguang Gong, and Casimir Kulikowski. Robust and fast collaborative tracking with two stage sparse optimization. In *European Conference on Computer Vision (ECCV)*, 2010.

[134] Xu Jia, Huchuan Lu, and Ming-Hsuan Yang. Visual tracking via adaptive structural local sparse appearance model. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[135] Wei Zhong, Huchuan Lu, and Ming-Hsuan Yang. Robust object tracking via sparsity-based collaborative model. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[136] Hanxi Li, Chunhua Shen, and Qinfeng Shi. Real-time visual tracking using compressive sensing. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[137] Chenglong Bao, Yi Wu, Haibin Ling, and Hui Ji. Real time robust L_1 tracker using accelerated proximal gradient approach. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1830–1837, 2012.

[138] Huaping Liu and Fuchun Sun. Visual tracking using sparsity induced similarity. In *IEEE International Conference on Pattern Recognition (ICPR)*, pages 1702–1705, 2010.

[139] Tianzhu Zhang, B. Ghanem, Si Liu, and N. Ahuja. Robust visual tracking via multi-task sparse learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[140] Baiyang Liu, Junzhou Huang, Lin Yang, and C. Kulikowsk. Robust tracking using local sparse appearance model and k-selection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[141] D. Ross, J. Lim, and R.-S. Linand M.-H. Yang. Incremental learning for robust visual tracking. *International Jounal of Computer Vision*, 77(1):125–141, 2008.

[142] J. Kwon and K. M. Lee. Visual tracking decomposition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

[143] A. Adam, E. Rivlin, and I. Shimshoni. Robust fragments-based tracking using the integral histogram. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006.

[144] B. Babenko, Ming-Hsuan Yang, and S. Belongie. Visual tracking with online multiple instance learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[145] S. Avidan. Ensemble tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

[146] H. Grabner and H. Bischof. On-line boosting and vision. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006.

[147] Z. Kalal, J. Matas, and K. Mikolajczyk. P-n learning: Bootstrapping binary classifiers by structural constraints. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

[148] B. Efron, T. Hastie, and R. Tibshirani. Least angle regression. *Annals of Statistics*, 32:407–499, 2004.

[149] J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online learning for matrix factorization and sparse coding. *Journal of Machine Learning Research*, 11:19–60, 2010.

[150] M. Everingham, L. Van Gool, C. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Jounal of Computer Vision*, 88(2):303–338, 2010.

[151] T. Horprasert, D.Harwood, , and L. S. Davis. A statistical approach for real-time robust background subtraction and shadow detection. *in ICCV'99 Frame Rate Workshop*, 1999.

[152] H. Jiang, S. Fels, , and J. Little. Adaptive background mixture models for real-time tracking. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1999.

[153] A. Amato, M. Mozerov, F. X. Roca, and J. Gonzlez. Robust real-time background subtraction based on local neighborhood patterns. *EURASIP J. Adv. Sig. Proc.*, 2010.

[154] Baiyang Liu, Junzhou Huang, Lin Yang, and C. Kulikowsk. Robust tracking using local sparse appearance model and k-selection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1313 –1320, june 2011.

[155] M. Aharon, M. Elad, and A. Bruckestein. K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Trans. Sign. Proc.*, 50(11):4311–4322, 2006.

[156] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. V. Gool. Online multi-person tracking-by-detection from a sing and uncalibrated camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(9):1820–1833, 2011.

[157] H. B. Shitrit, J. Berclaz, F. Fleuret, and P. Fua. Tracking multiple people under global appearance constraints. *IEEE International Conference on Computer Vision*, 2011.

INSA de RENNES
Service des Formations

AVIS DU JURY SUR LA REPRODUCTION DE LA THESE SOUTENUE

Titre de la thèse:
Contribution aux techniques de réduction de dimension : application au suivi d'objets

Nom Prénom de l'auteur : LU WEIZHI

Membres du jury :
- Monsieur RONSIN JOSEPH
- Madame GUILLEMOT Christine
- Monsieur KPALMA Kidiyo
- Monsieur COQUIN Didier
- Madame GOUIFFES Michèle
- Monsieur HAMAD Denis

Président du jury : Christine GUILLEMOT

Date de la soutenance : 16 Juillet 2014

Reproduction de la these soutenue

Thèse pouvant être reproduite en l'état
~~Thèse pouvant être reproduite après corrections suggérées~~

Fait à Rennes, le 16 Juillet 2014

Le Directeur,

M'hamed DRISSI

Signature du président de jury

Résumé

Cette thèse étudie et apporte des améliorations significatives sur trois techniques répandues en réduction de dimension : l'acquisition parcimonieuse (ou l'échantillonnage parcimonieux), la projection aléatoire et la représentation parcimonieuse.

En acquisition parcimonieuse, la construction d'une matrice de réduction possédant à la fois de bonnes performances et une structure matérielle adéquate reste un défi de taille. Ici, nous proposons explicitement la matrice binaire optimale, avec éléments zéro-un, en recherchant la meilleure propriété d'isométrie restreinte (RIP). Dans la pratique, un algorithme glouton efficace est successivement développé pour construire la matrice binaire optimale avec une taille arbitraire. Par ailleurs, nous étudions également un autre problème intéressant pour l'acquisition parcimonieuse, c'est celui de la performance des matrices d'acquisition parcimonieuse avec des taux de compression élevés. Pour la première fois, la limite inférieure de la performance des matrices aléatoires de Bernoulli pour des taux de compression croissants est observée et estimée.

La projection aléatoire s'utilise principalement en classification mais la construction de la matrice de projection aléatoire s'avère également critique en termes de performance et de complexité. Cette thèse présente la matrice de projection aléatoire, de loin, la plus éparse. Celle-ci est démontrée présenter la meilleure performance en sélection de caractéristiques, comparativement à d'autres matrices aléatoires plus denses. Ce résultat théorique est confirmé par de nombreuses expériences.

Comme nouvelle technique pour la sélection de caractéristiques ou d'échantillons, la représentation parcimonieuse a récemment été largement appliquée dans le domaine du traitement d'image. Dans cette thèse, nous nous concentrons principalement sur ses applications de suivi d'objets dans une séquence d'images. Pour réduire la charge de calcul liée à la représentation parcimonieuse, un système simple mais efficace est proposé pour le suivi d'un objet unique. Par la suite, nous explorons le potentiel de cette représentation pour le suivi d'objets multiples.

Mots-clés: réduction de dimension, acquisition parcimonieuse, projection aléatoire, représentation parcimonieuse.

Abstract

This thesis studies three popular dimension reduction techniques: compressed sensing, random projection and sparse representation, and brings significant improvements on these techniques.

In compressed sensing, the construction of sensing matrix with both good performance and hardware-friendly structure has been a significant challenge. In this thesis, we explicitly propose the optimal zero-one binary matrix by searching the best Restricted Isometry Property. In practice, an efficient greedy algorithm is successively developed to construct the optimal binary matrix with arbitrary size. Moreover, we also study another interesting problem for compressed sensing, that is the performance of sensing matrices with high compression rates. For the first time, the performance floor of random Bernoulli matrices over increasing compression rates is observed and effectively estimated.

Random projection is mainly used in the task of classification, for which the construction of random projection matrix is also critical in terms of both performance and complexity. This thesis presents so far the most sparse random projection matrix, which is proved holding better feature selection performance than other more dense random matrices. The theoretical result is confirmed with extensive experiments.

As a novel technique for feature or sample selection, sparse representation has recently been widely applied in the area of image processing. In this thesis, we mainly focus our attention on its applications to visual object tracking. To reduce the computation load related to sparse representation, a simple but efficient scheme is proposed for the tracking of single object. Subsequently, the potential of sparse representation to multi-object tracking is investigated.

Keywords: dimension reduction, compressed sensing, random projection, sparse representation.

N° d'ordre : 14ISAR21 / D14-21

Institut National des Sciences Appliquées de Rennes

20, Avenue des Buttes de Coëmes • CS 70839 • F-35708 Rennes Cedex

Tel : 02 23 23 82 00 - Fax : 02 23 23 83 96