THE UNIVERSITY of ADELAIDE

DOCTORAL THESIS

Adaptive Markov Random Fields for Structured Compressive Sensing

Author:
Suwichaya SUWANWIMOLKUL

Supervisor:
Assoc. Prof. Qinfeng SHI
Dr. Damith C. RANASINGHE

A thesis submitted in fulfillment of the requirements for the degree of Doctor of Philosophy

in the

Faculty of Engineering, Computer and Mathematical Sciences
School of Computer Sciences

October 20, 2018

Declaration of Authorship

I certify that this work contains no material which has been accepted for the award of any other degree or diploma in my name, in any university or other tertiary institution and, to the best of my knowledge and belief, contains no material previously published or written by another person, except where due reference has been made in the text. In addition, I certify that no part of this work will, in the future, be used in a submission in my name, for any other degree or diploma in any university or other tertiary institution without the prior approval of the University of Adelaide and where applicable, any partner institution responsible for the joint-award of this degree.

I give permission for the digital version of my thesis to be made available on the web, via the University's digital research repository, the Library Search and also through web search engines, unless permission has been granted by the University to restrict access for a period of time.

I acknowledge the support I have received for my research through the provision of the Department of State Development under the Collaboration Pathways Program (CPP39), Government of South Australia.

Signed:

Date:

Abstract

Compressive sensing (CS) has underpinned recent developments in data compression and signal acquisition systems. The goal of CS is to recover a high dimensional sparse signal from a few measurements. Recent progress in CS has attempted to further reduce the measurements by employing signal structures. This thesis presents a novel structured sparsity model, namely, *adaptive Markov random field (MRF)* to effectively extract the signal structures. The adaptive MRF achieves two desirable properties: *flexibility*—the ability to represent a wide range of structures—and *adaptability*—being adaptive to any structures. However, most existing work can only achieve one of these two properties. Previous MRF-based methods offer high flexibility but cannot adapt to new signal structures, while the data-adaptive based methods assume limited signal structures. Therefore, the contribution of this thesis is the novel and efficient signal recovery methods for CS.

We propose to leverage the adaptability of the MRF by refining the MRF parameters based on a point estimate of the latent sparse signal, and then the sparse signal is estimated based on the resulting MRF. This method is termed *Two-step-Adaptive MRF*. To maximize the adaptability, we also propose a new sparse signal estimation method that estimates the sparse signal, support, and noise parameters jointly. The point estimation of the latent sparse signals underpins the performance of MRF parameter estimation, but it cannot depict the statistical uncertainty of the latent sparse signals, which can lead to inaccurate parameter estimations, and thus limit the ultimate signal recovery performance.

Therefore, we reformulate the parameter estimation problem to offer better generalization over the latent sparse signals. We propose to obtain the MRF parameters from given measurements by solving a maximum marginal likelihood (MML) problem. The resulting MML problem allows the MRF parameters to be estimated from measurements directly in one step; thus, we term this method *One-step-Adaptive MRF*. To solve the MML problem efficiently, we propose to approximate the MRF model with the product of two simpler distributions which enables closed-form solutions for all unknown variables with low computational cost.

Extensive experiments on three real-world datasets demonstrate the promising performance of Two-steps-Adaptive MRF. One-step-Adaptive MRF further improves over the state-of-the-art methods. Motivated by this, we apply One-step-Adaptive MRF to collaborative-representation based classifications (CRCs) to extract the underlying information that can help identify the class label of the corresponding query sample.

CRCs have offered state-of-the-art performance in wearable sensor-based human activity recognition when training samples are limited. Existing work is based on the shortest Euclidean distance to a query sample, which can be susceptible to noise and correlation in the training samples. To improve robustness, we employ the adaptive MRF to extract the underlying structure of a representation vector directly from the query sample to improve discriminative power, because the underlying structure is unique to its corresponding query sample and independent of the quality of the training samples. The adaptive MRF can be customized to further reduce to the correlation in the training samples. Extensive experiments on two real-world datasets demonstrate the promising performance of the proposed method.

Acknowledgements

I would like to express my gratitude to Assoc. Prof. Qinfeng Shi for his helpful instructions, thoughtful directions, great patience, and very insightful feedback on this thesis. I truly appreciate his expertise and interest in the research area of probabilistic graphical models. I would like to thank my co-supervisor Dr Damith Rangnasinghe for his inspiration and his always-open door that leads to many theoretical and experimental discussions in this thesis. My gratitude is also extended to Dr Lei Zhang for his thoughtful advice in compressive sensing and putting so much effort and feedback that keeps my research on the right track. Working with these excellent researchers has certainly been an enriching life experience.

I would like to also express my sincere appreciation to Asst. Prof. Chao Chen and Mr Dong Gong for the kind advice and valuable feedback on my research writing as well as Dr Zhen Zhang for his assistance and suggestions regarding probabilistic graphical model inferences. I would like to thank my Master's supervisor, Asst. Prof. Supathana Auethavekiat, who introduced me to compressive sensing and for her consistent advice.

My appreciation also extends to Dr Asangi and Dr Asanga Jayatilaka who have provided great friendship and kindness. I would also like to thanks all my peers who have made this journey interesting and delightful: Dr Álvaro Parra, Dr Roberto Luis Shinmoto Torres, Peter Mathews, Jerome Williams, Dr Huu Le, Gabriel Maicas, Ergnoor Shehu, Hayden Faulkner, Mehdi Hosseinzadeh, and Alireza Abedin.

I am deeply grateful to my beloved family. My parents are role models of hard work and dedication. This also extends to my boyfriend Derek Hung and the Hung family for the daily support and encouragement which has helped me keep things in perspective.

I would like to thank Ms Alison-Jane Hunter for proofreading my thesis.

Lastly, I would like to thank the Department of State Development under the Collaboration Pathways Program (CPP39), Government of South Australia, for funding my PhD.

Contents

List of Figures

List of Tables

List of Algorithms

Notations

Throughout this thesis, scalars are denoted by italicized letters, as in k; vectors are denoted by boldface lowercase letters, as in $\boldsymbol{x}$; and matrices are denoted by boldface uppercase, as in $\boldsymbol{A}$. The i^{th} component of a vector $\boldsymbol{x}$ is denoted x_i.

$\|\|\cdot\|\|_2, \|\|\cdot\|\|$	euclidean norm for vectors.
$\|\|\cdot\|\|_1$	l_1 norm —sums the absolute of elements in a vector.
$\|\|\cdot\|\|_0$	l_0 norm —counts the number of nonzero elements.
$\mathcal{O}(\cdot)$	Big O notation.
$abs(\cdot)$	absolute value of a variable.
k	sparsity level–the number of none zero coefficients in a signal.
$\boldsymbol{s} = \text{supp}(\boldsymbol{x})$	support vector of signal $\boldsymbol{x}$.
$\boldsymbol{x}$	the unknown sparse signal of size N which has k non-zero elements (k-sparse).
$\boldsymbol{x}_s$	vector of non-zero coefficients in the sparse signal $\boldsymbol{x}$.
$\boldsymbol{x}_{\mathbb{N}}$	vector of the sparse signal coefficients in set $\mathbb{N}$.
$\boldsymbol{y}$	vector of measurements of size M.
$\boldsymbol{n}$	vector of measurement noise of size M.
$\boldsymbol{A}$	measurement matrix of size $M \times N$.
$\boldsymbol{A}_s$	sub-matrix of $\boldsymbol{A}$ with columns corresponding to the none zero elements in $\boldsymbol{s}$.
$\hat{\boldsymbol{x}}$	reconstructed or estimated sparse signal.
$\hat{\boldsymbol{x}}_s$	reconstructed or estimated non-zero coefficients in a sparse signal.
$\hat{\boldsymbol{s}}$	reconstructed or estimated support vector.
$\boldsymbol{I}_{N\times N}$	identity matrix of size $N \times N$.
$\mathbf{1}$	vector of value 1.
$\|\boldsymbol{M}\|$	determinant of a matrix $\boldsymbol{M}$.

$Tr[\cdot]$	trace of a matrix.
$\odot$	piecewise product.
$\mathrm{diag}(\boldsymbol{M})$	extract diagonal entries from matrix $\boldsymbol{M}$.
$\mathrm{diag}(\boldsymbol{m})$	form diagonal matrix with diagonal entries from $\boldsymbol{m}$.
$\mathcal{R}^N$	N-dimensional real vector space.
$\{0,1\}^N$	N-dimensional binary vector space.
$\mathcal{U}_k$	union of k-dimensional subspaces.
Ω_k	set of support with k sparsity level.
Ω_k^C	complement of the support set.
$\mathcal{U} \setminus \Omega$	excluding Ω from set $\mathcal{U}$.
$\emptyset$	empty set.
$\|\Omega\|$	cardinality of set Ω.
$\bigcup_{i=1}^{m} a_i$	union of $a_1, ..., a_m$.
$\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$	graph defined by sets of vertices $\mathcal{V}$ and edges $\mathcal{E}$.
$\mathbb{N}_i$	set of neighboring nodes of the i^{th} node.
$\mathcal{V}$	set of vertices.
$\mathcal{E}$	set of edges.
$E_i, \mathcal{E}_i$	set of edges connecting to the i^{th} node.
$p(\cdot)$	probability density function
$p_{\mathcal{G}}(\cdot; \boldsymbol{\Theta}_{\mathcal{G}})$	probability density corresponding to a graphical model where $\boldsymbol{\Theta}_{\mathcal{G}}$ denotes associated parameters.
$KL(p\|\|q)$	Kullback-divergence between a probability p and a reference probability q.
$\mathcal{N}(\cdot; \mu, \sigma^2)$	Gaussian distribution function with mean μ and variance σ^2.
$\boldsymbol{\Sigma_x}$	Covariance matrix corresponding to random vector $\boldsymbol{x}$.
$\mathrm{Bernoulli}(\cdot)$	Bernoulli distribution function.
$\mathrm{Gamma}(\cdot)$	Gamma distribution function.
$\mathrm{Beta}(\cdot)$	Beta distribution function.

For my parents.

Chapter 1

Introduction

1.1 Compressive Sensing

Compressive sensing (CS) provides an advanced sampling strategy that acquires high-dimensional signals under a sampling rate lower than the Nyquist's bandwidth (twice of the signal's Fourier bandwidth) [1], [2]. It has been the core development of new image acquisition and signal compression, where the resulting Nyquist rate is deemed too high for storage and transmission, and when acquiring each sample becomes financially prohibitive. CS has led to many developments for the new signal acquisition and sensing systems in several fields, i.e. remote sensing [3]–[5], medical imaging [6]–[9], and wireless communication [10]–[12].

To realize the sub-Nyquist rate, CS aims at recovering a high dimensional, sparse signal $\boldsymbol{x} \in \mathcal{R}^N$ that contains a few k non-zero coefficients from a few noisy linear measurements $\boldsymbol{y} \in \mathcal{R}^M$ where $M \ll N$. Because of the limited number of measurements, the signal recovery in CS is often an ill-posed problem (i.e. the solution space is infinite); thus, it necessitates an appropriate prior knowledge about the signal representation to achieve a good reconstruction result. By using the sparsity of the signal as a prior knowledge in signal recovery, standard CS algorithms can recover a sparse signal $\boldsymbol{x}$ from $\mathcal{O}(k \log N/k)$ noisy measurements [2], which is the minimum number of measurements required. Recent research in CS focus primarily on achieving the lowest number of measurements required.

To further reduce the number of measurements required, people started to exploit the underlying structure (i.e., interdependencies or correlations) of the coefficients in a sparse signal in addition to the simple sparsity [13], [14]. However, each image and signal processing task often employs different types of sparse signal representations. For example, natural image processing employs the sparse signal representation of the image in the wavelet or DCT domain. Meanwhile, Fourier representation is often used in audio signal processing. A number of research attempted to find a structured sparsity model that is flexible enough to represent the broad range of the underlying structures in different sparse signal representations. To this end, two dominant classes of structured sparsity models have been studied [13], [14] including *deterministic* structured sparsity [15]–[28] and *probabilistic* structured sparsity models [29]–[44].

The deterministic structured sparsity models often assume prior knowledge about the geometrical structure of sparse signals. For example, block sparsity models assume the locations and sizes of the coefficient blocks in a sparse signal that they seek to recover [15]–[20]. Hierarchical sparsity models assume that the signal coefficients are organized as a tree structure [21]–[24]. For this case, the required number of measurements can achieve the information-theoretical optimum $\mathcal{O}(k)$ [13]. However, many signals do not follow the assumed block or tree structure. To circumvent this loss of flexibility, one line of work exploits graph sparsity [25]–[28]. Their flexibility, however, comes at the cost of expensive parameter tuning, e.g., the number of connected components, maximum accumulated graph weight, and sparsity level. These parameters are often unknown in practice. Moreover, most of these deterministic structured sparsity models exclude all the signals that violate their assumptions about the geometrical structure from the solution space [13], [27].

To avoid excluding signals and to achieve the small number of measurements, Cevher *et al*. [29] proposed the concept of probabilistic RIP and used Markov random fields (MRFs) to model the structure of sparse signals. The MRFs have high flexibility and expressiveness for modelling a wide variety of signal structures. This opens up a new line of work [29]–[36], where an MRF is employed to represent the underlying structure of the sparse signals. The MRF represents the underlying structure by defining a probability distribution over an undirected graph. The parameters and the underlying graph of the MRFs are learned from extensive training examples. Therefore, the performance of the MRF is constrained by the information in the training examples. These methods can fail to capture the new underlying structure of the sparse signal, which are different from the those of the training examples; thus, they lack the adaptability to model a new signal structure.

To address the lack of adaptability problem, one line of research [37]–[44] has developed a data-adaptive model without the necessity for training. The majority of this type of research resorts to clustered sparsity models [37]–[41] where a mixture model such as beta-Bernoulli or Gaussian-Gamma is employed to model signal distributions. The mixture models allow the model parameters to be adaptively updated with closed-form formulations. However, this work assumes that the sparse signals exhibit clustered structure, i.e., the non-zero coefficients of the sparse signals group into clusters. Hence, the clustered sparsity models are not as flexible as the MRFs. Among these data-adaptive models, the works [42]–[44] model the cluster structure with MRFs, but these MRFs contain only the pairwise potentials. Although the parameters of pairwise potentials can be adaptively estimated, the underlying graph of the MRF is fixed and cannot be adapted for new structures of the sparse signals.

Ultimately, the key to effectively exploiting the underlying structure of sparse signals is to develop a structured sparsity model that is not only able to represent a wide range of the underlying structure of sparse signals, but also able to adapt for new signal structures. However, most of the existing research achieves only one of these two properties. Therefore, our motivation is to develop a new data-adaptive model that has the flexibility to capture the broad range of signal structures and the adaptability to adjust for new signal structures.

1.2 Flexibility and adaptability

The structure of the sparse representation can be diverse across different applications. Moreover, the signals from the same data sources can exhibit a large variability between them. Therefore, to represent the variety of the signal structures, a desirable structured sparsity model should possess two important properties:

- **Flexibility** —the ability to represent a wide range of structures;

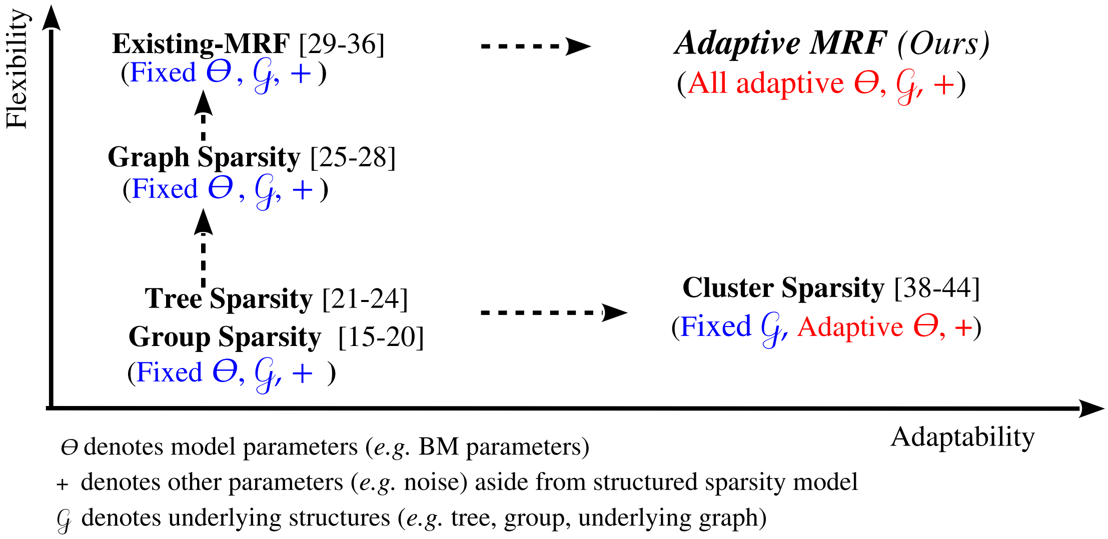

Flexibility
Existing-MRF [29-36]
(Fixed Θ, $\mathcal{G}$, +)
Graph Sparsity [25-28]
(Fixed Θ, $\mathcal{G}$, +)
Tree Sparsity [21-24]
Group Sparsity [15-20]
(Fixed Θ, $\mathcal{G}$, +)
Adaptive MRF (Ours)
(All adaptive Θ, $\mathcal{G}$, +)
Cluster Sparsity [38-44]
(Fixed $\mathcal{G}$, Adaptive Θ, +)
Adaptability
Θ denotes model parameters (e.g. BM parameters)
+ denotes other parameters (e.g. noise) aside from structured sparsity model
$\mathcal{G}$ denotes underlying structures (e.g. tree, group, underlying graph)

FIGURE 1.1: Flexibility and adaptability comparison between our Adaptive-MRF and the existing structured CS algorithms.

- **Adaptability**—the ability to adapt for any sparse signal structure, according to given measurements.

To achieve these two properties, we propose to leverage the adaptability of a Markov random field (MRF) [29]–[36]. The MRF represents the structure of signals with a graphical model. A Boltzmann machine (BM) is used as the probability distribution because of its ability to model different signal distributions. Thus, we aim to adapt the MRF parameters including *BM parameters* and the *underlying graph* of the MRF for any signal structures; thus, termed *adaptive MRF*.

The comparison of flexibility and adaptability between our approaches and the existing structured CS algorithms is shown in Figure 1.1. The proposed method inherits flexibility from the MRF and adaptability from the adaptive estimation mechanism. Hence, unlike the existing MRF approaches, our adaptive MRF can adapt its underlying graph and BM parameters to fit any signal structure. Unlike the existing data-adaptive model-based approaches such as the clustered sparsity model-based methods [37]–[44], our MRF model is more flexible and can adapt its underlying graph.

With the adaptability and flexibility properties, the proposed adaptive MRF can extract the salient information about the signal structure to provide a good prior knowledge for signal recovery. As a result, it can potentially improve the performance of sparse signals recovery when the signal recovery is performed under a very low sampling rate (high compression) and under high noise corruption (high noise tolerance).

1.3 Contributions and thesis outline

The contributions of this thesis are the novel and efficient adaptive-MRF to models structure sparsity for signal recovery in CS. We propose two new adaptive MRF-based approaches to recover sparse signals by using the adaptive MRF as the prior knowledge in signal recovery, namely *Two-steps-Adaptive MRF* and *One-step-Adaptive MRF* (presented in Chapter 3 and 4) which is then applied to develop a new Adaptive MRF-based classification method in Chapter 5. This work has led to two submitted journal articles [45], [46].

We present our contribution in more details in the following paragraphs.

1.3.1 Two-steps-Adaptive MRF

We propose to leverage the adaptability of the MRF that has been proven for its flexibility to capture different signal structures. To realize adaptability, the MRF parameters are adaptively estimated based on the point estimate of the latent sparse signals. To maximize adaptability, we also propose a new algorithm for *sparse signal estimation* that is able to jointly and iteratively estimate the support and the sparse signal, noise and signal parameters. Experiments on three real-world datasets demonstrate the effectiveness of our framework over state-of-the-art methods (see Chapter 3).

1.3.2 One-step-Adaptive MRF

The point estimation of the latent sparse signals underpins the performance of MRF parameter estimation in the Two-step-Adaptive. However, the point estimation cannot depict the statistical uncertainty of the latent signals. To capture the uncertainty, we reformulate the MRF parameter estimation into a maximum marginal likelihood (MML) problem. We propose to approximate the MRF distribution with a product of two simpler distributions to enable closed-form solutions for all the unknown variables with low computational cost. Extensive experiments on three real-world datasets demonstrate the superior performance of the proposed One-step-Adaptive MRF over state-of-the-art methods (Chapter 4).

1.3.3 Adaptive MRF-based classification

Collaborative representation-based classifications (CRCs) have enabled state-of-the-art performance in wearable sensor-based human activity recognition, when training samples are limited. Most of the existing methods are based on a shortest Euclidean distance, which can be susceptible to noise and correlation in the training samples. We propose to employ One-step-Adaptive MRF to extract the underlying structure of the representation vector to help identify the class label, which improves the discriminative power of the classifier. The adaptive MRF can be customized to further reduce the ambiguity due to the correlated training samples. With adaptive MRF, the classification performance improves over that of competitors (see Chapter 5).

1.3.4 The improved sparse signal recovery performance

Our adaptive MRF can potentially improve the compressibility and noise tolerance performance of sparse signal recovery. We evaluate the performance in four aspects: compressibility, noise tolerance, runtime, and classification robustness, as evidenced by extensive experiments:

1.3.4.1 Better compressibility.

The adaptive MRF can offer a good prior knowledge for sparse signal recovery, which results in improved signal recovery accuracy across different sampling rates. Two-step-Adaptive MRF offers promising results across different sampling rates in recovering many sparse signal representations in Figure 3.11. With the improved parameter estimation, One-step-Adaptive MRF yields a significant improvement in signal recovery accuracy across different sampling rates and achieves state-of-the-art performance, as demonstrated in Figure 4.6.

1.3.4.2 Better noise tolerance.

The adaptive MRF can offer better differentiation between the true signal information and noise. The Two-step-Adaptive MRF offers promising results across different noise levels in recovering many sparse signal representations, as shown in Figure 3.15. With the improved MRF parameter estimation, One-step-Adaptive MRF achieves a significant improvement which leads to state-of-the-art performance, as shown in Figure 4.10.

1.3.4.3 Better runtime.

Our adaptive MRF-based approaches, the Two-step-Adaptive MRF and One-step-Adaptive MRF, require less runtime than the existing MRF based-methods such as [31], [32] as shown in Figure 3.16 and 4.11 because our designed algorithms either require less algorithm complexity or fewer iterations to converge to a stable result. Two-step-Adaptive MRF requires less complexity than [32] in the worst case scenario (see Section 3.5) and fewer iterations to converge than [31] (see Section 3.10). With the improved MRF parameter estimation, One-step-Adaptive MRF has significantly less complexity and runtime than the Two-step-Adaptive MRF (see Section 4.4).

1.3.4.4 Higher classification robustness.

Our adaptive MRF-based classification improves the robustness of CRCs in wearable sensor-based human activity recognition when the number of training samples is small. The adaptive MR can extract the underlying structure of a representation vector from a query sample. The underlying structure can help identify the class label and is independent of the quality of the training data which can be noisy and correlated across different classes. Our adaptive MRF-based classification demonstrates high tolerance against ambiguity due to noise and correlation among training data over other classification methods (e.g., Figure 5.7).

Moreover, because our Two-step-Adaptive MRF and One-step-Adaptive MRF adaptively estimate the MRF parameters based on given measurements, they do not require any model training. Thus, this removes the requirement for the storage and computing process for training a model, by default. The proposed approach requires the computer memory only for a measurement matrix used in the sampling process of CS and the parameter settings which are scalar values, e.g. the cardinality for an edge set and the adaptive MRF, the maximum number of iterations for the algorithm terminating criterion, and scalar initial values for noise and signal variance. Therefore, the proposed approaches require less memory than the existing MRF-based methods and can be desirable for many real-world applications that have memory constraints.

In conclusion, we have discussed our key motivation to develop the adaptive MRF and the new sparse signal recovery approaches to flexibly capture and employ the underlying structure of sparse signals to improve the compressibility and noise tolerance performance. In Chapter 2, we will discuss structured compressive sensing, specifically, recovery with deterministic structured sparsity models and probabilistic structured sparsity models, and elaborates the associated sampling complexity for each class of the structured sparsity models. We will also provide a review about the inference and learning techniques of MRFs, as well as the collaborative representation-based classifications. The summary of this thesis and future work are provided in Chapter 6.

Chapter 2

Structured Compressive Sensing

2.1 Introduction

The goal of compressive sensing (CS) is to recover a high dimensional signal from a few measurements. Recent research in CS focuses primarily on reducing the number of measurements. To achieve this, people started to exploit the structure of the latent sparse signal, i.e. the interdependency or correlations of the coefficients in the sparse signal. Two main classes of structure sparsity models that have been studied to efficiently exploit the signal structures include *deterministic structured sparsity models* that impose prior knowledge about the geometrical structure of sparse signals, e.g. group sparsity models, hierarchical sparsity models, and graph sparsity models [15]–[28]; and *probabilistic structured sparsity models* that employ flexible graphical models to capture the signal structure, e.g. Markov random fields (MRFs). Some of the probabilistic structured sparsity models can adapt for new signal signals, i.e. clustered structured sparsity models [29]–[42]. Therefore, one of the foci of this chapter is to give insight into the underlying assumptions and respective limitations of these two classes of structure sparsity models that lead to limited flexibility to capture different signal structures. Our proposed adaptive MRF leverages the flexibility and adaptability of these probabilistic models. Then, we provide background regarding the inference and learning of the MRFs and the application of sparse signal recovery to classification.

We highlight the following discussions covered by this chapter:

Signal recovery using deterministic structure sparsity models is presented to explain why the deterministic structured sparsity models lack the flexibility to model signal structures. Three different examples of signal recovery algorithms with group sparsity models, hierarchical sparsity models, and graph sparsity models are also provided. The discussion is in Section 2.2.1.

Then, signal recovery using probabilistic structure sparsity models is presented. How the probabilistic structured sparsity models improve the flexibility in modelling signal structures is explained. Three examples of signal recovery algorithms with graphical structured sparsity and clustered structured sparsity models are provided in Section 2.2.2.

We turn to the relationship between sample complexity and the structured sparsity model. The discussion on how the deterministic structured sparsity models achieve the low sample complexity by reducing the feasible set is in Section 2.3.1. The relationship can be shown by the connection between the restricted isometry property (RIP) and the sample complexity.

Unlike the deterministic structured sparsity models, the probabilistic structured sparsity models can reduce the sample complexity without restricting the feasible set. However, the sample complexity can be reduced only if the underlying information in the training samples is representative of the testing samples. The sample complexity of the probabilistic models is analyzed through the notion of the probabilistic RIP. These discussions are provided in Section 2.3.3.

We provide background on the Markov random fields (MRFs) that have been used in our proposed method in Chapters 3 and 4 for their flexibility to capture different signal structures. We also review the graphical model inference and learning on MRFs and a graphical model learning algorithm that has been used for achieving adaptive MRF. The review is in Section 2.4.

Then, the application of sparse signal recovery to classification is discussed. We discuss three classification methods, i.e. sparse representation based classification (SRC), collaborative representation based classification (CRC), and probabilistic collaborative representation based classification (ProCRC) in Section 2.5. Finally, a summary of this chapter is presented in Section 2.6.

The following sections are organized as follows: we discuss the structured sparse signal recovery in Section 2.2. Then, the discussion on the corresponding sample complexity is provided in Section 2.3. Our revision on Markov random fields is in Section 2.4. Finally, we review the classification methods based on sparse signal recovery, i.e., SRCs, CRCs, and ProCRC in Section 2.5.

2.2 Structured sparse signal recovery problem

The goal of compressive sensing is to recover a sparse signal $\boldsymbol{x} \in \mathcal{R}^N$ from noisy linear measurements $\boldsymbol{y} \in \mathcal{R}^M$ where $M \ll N$, i.e.

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{n}, \tag{2.1}$$

where $\boldsymbol{A} \in \mathcal{R}^{M \times N}$ represents a random measurement matrix, and $\boldsymbol{n}$ represents a small perturbation where $M \ll N$. Here, the sparse signal $\boldsymbol{x}$ is defined as a signal that possesses a few k non-zero coefficients, lying in ambient dimensionality. The problem of recovering sparse signal $\boldsymbol{x}$ is ill-posted (i.e. the solution space is infinite); thus, it necessitates an appropriate prior on sparse signals to effectively reduce the solution space. *Sparsity* of the signal is a commonly used prior in sparse signal recovery. The model for sparsity is defined as

$$\boldsymbol{x} \in \mathcal{U}_k = \{\boldsymbol{x} : ||\boldsymbol{x}||_0 \leq k\}. \tag{2.2}$$

Given the prior of the signal sparsity, the signal recovery can be formulated as the following optimization problem:

$$\hat{\boldsymbol{x}} = \min_{\boldsymbol{x} \in \mathbb{R}^N} ||\boldsymbol{x}||_0 \qquad \text{subject to} \qquad ||\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}||_2 \leq \epsilon \tag{2.3}$$

where the regularization term $||\boldsymbol{x}||_0$ depicts the sparsity of $\boldsymbol{x}$, and ϵ is a small value that bounds the deviation of measurements due to noise corruption. $\epsilon = 0$ in the noiseless case. With the sparsity assumption, the cardinality of the solution space is reduced to the number of subspaces in $\mathcal{U}_k$, i.e. $\binom{N}{k}$. However, solving the l_0-problem is NP-hard [47]. A common approach is to approximate l_0-norm with l_1-norm to depict the sparsity of the solution signal.

The structure of sparse signals can be employed as a prior in addition to simple sparsity to restrict the solution space. Most of the existing studies [15]–[42], [48] employ the signal structure as a criterion to select the candidate signals in the solution space [13], [25]–[28], [48]. Alternatively, the regularization term with a special function is used to enforce the structure of the solution signals according to the prior knowledge about the signal structure, [15]–[17], [28]–[42]. With the smaller solution space, the minimum measurements, defined as sample complexity, for successful recovery can be further reduced [13], as will be discussed in Section 2.3.

The two classes of structured sparsity models are explored in the following.

2.2.1 Signal recovery with deterministic structured sparsity models

The deterministic structured sparsity models assume prior knowledge about the geometrical structures of sparse signals in addition to their sparsity. Formally, the models are represented as a union of k-dimensionality subspaces [13], [14], [49]: let x_Ω represent the coefficients of $\boldsymbol{x}$ chosen according to the set $\Omega \subseteq \{1, ..., N\}$ and Ω^C denote the complement of Ω.

Definition 2.2.1. A structured sparsity model $\mathcal{M}_k$ is defined as the union of m_k canonical k-dimensional subspaces:

$$\boldsymbol{x} \in \mathcal{M}_k = \bigcup_{m=1}^{m_k} \mathcal{S}_m \quad \text{where} \quad \mathcal{S}_m = \{\boldsymbol{x} : \boldsymbol{x}_{\Omega_m} \in \mathcal{R}^k, \boldsymbol{x}_{\Omega_m^C} = 0\}. \tag{2.4}$$

where $\{\Omega_1, ..., \Omega_{m_k}\}$ is the set containing all allowed supports, with $|\Omega_m| = k$ for each $m = 1, ..., m_k$, and each subspace $\mathcal{S}_m$ contains all signals $\boldsymbol{x}$ with $\mathrm{supp}(\boldsymbol{x}) \subseteq \Omega_m$ [13].

It can be seen that a structured sparsity model $\mathcal{M}_k$ contains m_k subspaces only. Each subspace is the set of sparse signals whose support exhibits a certain pattern defined by Ω_m. As a result, the structured sparsity model $\mathcal{M}_k$ restricts the number of subspaces from $\binom{N}{k}$ to m_k, and thus, it limits the number of candidate solutions. Different configurations of $\mathcal{M}_k$ can vary the number of subspaces. The deterministic structured sparsity model is often imposed as a constraint in the optimization problem to solve for sparse signals:

$$\hat{\boldsymbol{x}} = \min_{\boldsymbol{x} \in \mathcal{R}^N} ||\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}||_2 \quad \text{subject to} \quad \boldsymbol{x} \in \mathcal{M}_k, \tag{2.5}$$

where the structured sparsity model is defined through $\mathcal{M}_k$. If the configuration of the structured sparsity model is simple, the constraint of the structured sparsity model can be replaced with a regularization term to induce a structured sparsity in the candidate sparse signal solution. However, if the configuration of the structured sparsity model is complicated, solving the constrained optimization problem Eq. (2.5) directly can be difficult. A group of research [13], [25], [27], [48] resorted to recovering the signal with greedy approaches such as CoSAMP [50], IHT [51], and obtained candidate sparse signals from the best k-term structured sparse approximation [13], [25], [27], [48]. The best k-term structured sparse approximation aims to search for the k-sparse signal candidate in $\mathcal{M}_k$ that minimizes Euclidean distance to an intermediate estimate of the latent sparse signals [13], [25] (see Section 2.2.1.2).

However, the sparsity-induced regularization as well as the best k-term structured sparse approximation does not necessarily offer good candidate sparse signals, especially when the underlying structure of the sparse signals to be reconstructed is different from the assumed geometrical structure. Also, the best k-term structured sparse approximation can be inaccurate, if the assumed geometrical structure imposed on $\mathcal{M}_k$ is too restrictive. Thus, the deterministic structured sparsity model could perform poorly in such cases. Many deterministic structured sparsity models have been developed to accommodate with different sparse signal structures. Such structured sparsity models can be grouped into three broad classes:

- **Group/Block sparsity models** [15]–[20] assume that signal coefficients in one group/block have to be either all zero or all non-zero. This property has been enforced by l_1/l_2 norms in early research, and has been extended to overlapping group-sparsity. Recently, the research [38]–[42] proposes cluster sparsity models which are improved from the group sparsity model by enabling adaptive parameter estimation, given the measurements.

- **Hierarchical sparsity models** [21]–[24] represent signal coefficients as trees.

For example, the wavelet transformation of a piecewise smooth signal often exhibits the tree structure, where a zero parent node implies zero offspring nodes [21]–[23]. Another example is the k-sparse rooted sub-tree model [24], where only non-zero element nodes form a sub-tree.

- **Graph sparsity models** [25]–[28] organize signal coefficients in a general graph, thus are able to represent various types of sparsity patterns, including the above group along with hierarchical sparsity models. Initially, graph sparsity models are employed as sparsity-induced regularization to capture the overlapping-group sparsity pattern [25], [26]. Recently, a weighted graph sparsity model [27], [28] has been employed where the candidate structured sparse signal is obtained from the best k-term structured sparse approximation [13].

However, group/block and hierarchical sparsity models only fit signals with assumed structures, thus they are considered as lacking the flexibility to cope with different signal structures. Graph sparsity models have better flexibility, however, its flexibility could come with the cost of expensive parameter tuning [13]. Moreover, these models cannot adapt for different signal structures, once the models have been tuned. The following algorithms are provided as examples of signal recovery using three different deterministic structured sparsity models: (i) the group-lasso [52] is used as an example for signal recovery with the group sparsity models; (ii) MBCS [13], [53] is the example for signal recovery with hierarchical sparsity model; and (iii) GraphCoSaMP [27], [48] is the example of signal recovery with the graph sparsity model.

2.2.1.1 Group-lasso

In group-lasso [52], [54], [55], it is assumed that the sparse signals exhibit a group structure where all coefficients within the same group become zero/nonzero simultaneously. The group structure in sparse signal $\boldsymbol{x}$ can be modelled through the mix $l_{2,1}$-regularization function. Let m_k denote the total number of groups. $\boldsymbol{x}_m$ is a sub-vector associated with the m^{th} group containing sparse coefficients that are entirely zero/non-zero. $\boldsymbol{A}_m$ is the sub-matrix whose columns are chosen from $\boldsymbol{A}$ according to the coefficients in the sub-vector $\boldsymbol{x}_m$. The sparse signal recovery can be formulated as the following convex optimization problem:

$$\hat{\boldsymbol{x}} = \min_{\boldsymbol{x} \in \mathbb{R}^N} \frac{1}{2} ||\boldsymbol{y} - \sum_{m=1}^{m_k} \boldsymbol{A}_m \boldsymbol{x}_m||_2 + \gamma \sum_{m=1}^{m_k} ||\boldsymbol{x}_m||_2. \tag{2.6}$$

where γ is the constant controlling the sparsity level of sparse signals. This convex optimization problem can be solved efficiently using a block coordinate descend. With this technique, the sparse signal is calculated by performing the following updates in each iteration:

$$\hat{\boldsymbol{x}}_m = \begin{cases} 0, & \text{if } ||\boldsymbol{A}_m \boldsymbol{r}_m||_2 \leq \gamma \\ \left(\boldsymbol{A}_m^T \boldsymbol{A}_m + \frac{\gamma}{||\hat{\boldsymbol{x}}_m||} \boldsymbol{I} \right)^{-1} \boldsymbol{A}_m^T \boldsymbol{r}_m, & \text{otherwise.} \end{cases} \tag{2.7}$$

where $\boldsymbol{r}_m = \boldsymbol{y} - \sum_{j \neq m} \boldsymbol{A}_j \boldsymbol{x}_j$.

Lasso is an efficient approach to recover a sparse signal with a group structure [52]. However, it should be noted that not every signal exhibits the group structure. Next, we will explore examples of the hierarchical and the graph structured sparsity models that can model more flexible signal structures.

2.2.1.2 MBCS with tree-sparsity model

MBCS [13], [53] offers a general signal recovery framework that allows any structured sparsity model to be integrated into a fast CS recovery algorithm such as CoSaMP [13], [25]. In this example, the tree-sparsity model is used to represent a sparse signal structure. The tree-sparsity model assumes that the coefficients of a k-sparse signal can be modelled with a *binary tree* where only k non-zero coefficients can form rooted subtrees. Each variation of k-rooted subtrees represents a subspace. Thus, the tree-sparsity model is defined as [13], [53]:

$$\begin{aligned} \mathcal{T}_k = \{\boldsymbol{x} : \ \boldsymbol{x}_\Omega \subset \mathcal{R}^k, \boldsymbol{x}_{\Omega^C} = 0 \ \text{ where } \ |\Omega| = k, \\ \Omega \ \text{ forms a connected subtree}\}. \end{aligned} \tag{2.8}$$

The tree sparsity model is employed to define the solution space. MBCS obtains the candidate signal with tree structure from the best k-term structured sparse approximation [13]. The best k-term structured sparse approximation is obtained by

Algorithm 2.1 MBCS with tree-sparsity model.

```
Input: Measurements $y$, a measurement matrix $A$, the expected sparsity level $k$, and
the algorithm for tree-structured sparse approximation $\mathbb{M}$.
Initialization : $\hat{x} = 0, d = y$.
  while a stopping criterion is not satisfied do
      1. Form signal residual estimate:
         $e \leftarrow A^T d$;
      2. Prune residual estimate according to tree structure:
         $\Omega \leftarrow \mathrm{supp}(\mathbb{M}(e, k))$;
      3. Merge supports:
         $T \leftarrow \Omega \cup \mathrm{supp}(\hat{x})$;
      4.Form signal estimate:
         $b_T \leftarrow A_T^{\dagger} y, b_{T^C} = \mathbf{0}$;
      5. Prune signal estimate according to tree structure
         $x \leftarrow \mathbb{M}(b, k)$;
      6. Update measurement residual:
         $d \leftarrow y - Ax$;
  end while
Output: Recovered $\hat{x}$.
```

solving the following optimization problem:

$$x_k^{\mathcal{T}} = \min_{\bar{x} \in \mathcal{T}_k} ||x - \bar{x}||_2 \tag{2.9}$$

that aims to search for $\bar{x} \in \mathcal{T}_k$, which has the shortest Euclidean distance to an intermediate estimation of x. The optimization problem in Eq. (2.9) can be solved efficiently with the condensing sort and select algorithms [13], [56]. CoSaMP [50] is employed to recover sparse signals where the candidate structured sparse signals are obtained from the best k-term structured sparse approximation. The whole process of the model based CoSAMP is summarized in Algorithm 2.1 where $\mathbb{M}(\cdot, \cdot)$ is the algorithm that can solve the optimization problem in Eq. (2.9).

MBCS has a flexible framework to employ any sparsity model; however, a new sparsity model has to be redesigned every time when the assumptions of the geometrical structure of the sparse signal are changed. Moreover, the performance of the best k-term structured sparse approximation crucially depends on the configuration of the sparsity model $\mathcal{T}_k$. As a result, the best k-term structured sparse approximation Eq. 2.9 does not necessarily provide a good approximation, especially, either when the structure of the intermediate estimate x is different from the assumed geometrical structure in the sparsity model $\mathcal{T}_k$, or when the sparsity model $\mathcal{T}_k$ is restrictive.

2.2.1.3 Graph-CoSaMP

Graph-CoSaMP [25], [27], [48] employs a graph sparsity model that can flexibly model many signal structures aside from the assumed block or tree structure. Graph-CoSaMP uses a weighted graph model whose configuration is achieved by adjusting a set of parameters; thus, unlike MBCS, the weight graph does not need to be totally redesigned for a new signal structure every time. Let s denote signal support, and $S \subseteq [N]$ is the corresponding index set. $G = (V, E)$ denotes the underlying graph. The desired configuration of the weighted graph model is achieved through adjusting the following parameters.

- k the total sparsity of S
- g is the maximum number of connected components formed by the forest F corresponding to S.
- B is the bound of the total weight $w(F)$ of edges in the forest F corresponding to S.

Let $\gamma(H)$ be the number of the connected components in a graph H. The weighted graph model $\mathcal{W}_{G,k,g,B}$ is defined as

$$\begin{aligned} \mathcal{W}_{G,k,g,B} = \{S : \; S \subseteq [N], |S| = k \; \text{ and there is a } F \subseteq G \\ \text{with } \; V_F = S, \gamma(F) = g, \text{and } \; w(F) \leq B\}. \end{aligned} \tag{2.10}$$

Since the graph sparsity model cannot be directly mapped into a regularization function, these approaches [25], [27], [48] resort to searching for the best k-term structured sparse approximation in the structured sparsity model by solving the optimization problem Eq. (2.9) where the solution space is defined by the weighted graph model Eq. (2.10). However, solving this optimization problem exactly is NP-hard for the weighted graph model. To circumvent this problem, an approximation-tolerant framework is used instead of Eq. (2.9). The approximation-tolerant framework requires two algorithms with the following complementary approximation guarantees:

Tail approximation: Find an $S \in \mathcal{W}_{G,k,g,B}$ such that

$$||\boldsymbol{x} - \mathrm{M}(\boldsymbol{x}, S)||_2 \leq c_T \min_{S' \in \mathcal{W}_{G,k,g,B}} ||\boldsymbol{x} - \mathrm{M}(\boldsymbol{x}, S')||_2 \tag{2.11}$$

Algorithm 2.2 Graph-CoSAMP (GCoSaMP).

```
Input: Measurements $\boldsymbol{y}$, a measurement matrix $\boldsymbol{A}$, and parameters to configure the weight graph model $\mathcal{W}_{G,k,g,B}$ —$G$,$B$,$k$, and $g$— and number of iteration $t$.
Initialization : $\hat{\boldsymbol{x}} = 0$.
  for $i \leftarrow 1, ..., t$ do
      1.Form signal residual estimate:
          $\boldsymbol{r} \leftarrow \boldsymbol{A}^T(\boldsymbol{y} - \boldsymbol{A}\hat{\boldsymbol{x}})$;
      2.Merge the supports with head approximation:
          $\Omega \leftarrow \text{supp}(\hat{\boldsymbol{x}}) \cup \text{HEADAPPROX}'(\boldsymbol{r}, G, k, g, B)$;
      3. Form signal estimate
          $\boldsymbol{b}_\Omega \leftarrow \boldsymbol{A}_\Omega^\dagger \boldsymbol{y}, \boldsymbol{b}_{\Omega^C} = \boldsymbol{0}$;
      4. Obtain the supports with tail approximation:
          $S \leftarrow \text{TAILAPPROX}'(\boldsymbol{b}, G, k, g, B)$;
      5. Form signal estimate with the new support:
          $\hat{\boldsymbol{x}}_S \leftarrow \boldsymbol{b}_S$;
  end for
Output: Recovered $\hat{\boldsymbol{x}}$.
```

Head approximation: Find an $S \in \mathcal{W}_{G,k,g,B}$ such that

$$||\text{M}(\boldsymbol{x}, S)||_2 \leq c_H \min_{S' \in \mathcal{W}_{G,k,g,B}} ||\text{M}(\boldsymbol{x}, S')||_2 \tag{2.12}$$

where $\text{M}(\boldsymbol{x}, S)$ is the function that sets all coefficients in $\boldsymbol{x}$ that are not specified in the index set S to zero.

It is shown in [27] that these two approximations Eq. (2.11) and Eq. (2.12) can be solved based on connection to the prize-collecting Steiner tree problem (PCST). Both approximations Eq. (2.11) and Eq. (2.12) are to be modified to a formalization of the PCST problem that can be solved with extant algorithms. For more details on how the two approximations are solved, we refer the reader to the full papers [27], [48]. The whole process of Graph-CoSaMP is summarized in Algorithm 2.2.

Graph-CoSaMP [25], [27], [48] is so far a novel and effective signal recovery approach that employs the flexible weighted graph model to capture the signal structure. However, it should be noted that tuning the required parameters in both the weighted graph model and Graph-CoSaMP is not easy. This includes the underlying graph G, the bound of total graph weight B, signal sparsity k, and the maximum number of connected components g, which are often unknown. More importantly, Graph-CoSaMP employs the similar best k-term structured sparse approximation as

MBCS. Thus, Graph-CoSaMP has the same problem as MBCS. That is, it can falsely exclude a good candidate signal, either when the testing signal $\boldsymbol{x}$ is different from the assumed geometrical structure, or when the assumed geometrical structures specified by the weighted graph model are too limited.

2.2.2 Signal recovery with probabilistic structured sparsity models

To flexibly represent various signal structures and avoid false exclusion of candidate signals, a group of research [29]–[42] resorts to employing the probabilistic structured sparsity models and a Bayesian approach for recovering sparse signals. In this setting, the non-zero coefficients in $\boldsymbol{x_S}$ are assumed to be a realization of an iid multivariate Gaussian with zero mean and covariance matrix $\boldsymbol{\Sigma_{x_S}}$ where $\boldsymbol{s} \in \{0,1\}^N$ is the binary support of $\boldsymbol{x}$, such that $s_i = 1$ when $x_i \neq 0$ and $s_i = 0$ when $x_i = 0$. Based on the observation model Eq. (2.1), the measurements are assumed to be corrupted by an iid Gaussian noise with zero mean and variance σ_n. The observation likelihood model given the measurements $\boldsymbol{y}$ can be formulated as

$$p(\boldsymbol{y}|\boldsymbol{x}_s, \boldsymbol{s}; \sigma_n) = \mathcal{N}(\boldsymbol{A_S x_S}, \sigma_n \boldsymbol{I}). \tag{2.13}$$

The structure of the sparse signal is modelled through the signal support. Hence, the probability of the support $p(\boldsymbol{s})$ is a prior. Our objective is to recover the sparse signals from solving a maximum a posteriori (MAP) problem:

$$\{\hat{x}, \hat{s}\} = \max_{\boldsymbol{x} \in \mathbb{R}^N, \boldsymbol{s} \in \{0,1\}^N} p(\boldsymbol{x_S}, \boldsymbol{s}|\boldsymbol{y}) \propto p(\boldsymbol{y}|\boldsymbol{x_S}, \boldsymbol{s}) p(\boldsymbol{x_S}|\boldsymbol{s}) p(\boldsymbol{s}). \tag{2.14}$$

Given the probabilistic model for the support $p(\boldsymbol{s})$ in Eq. (2.14), most of the existing approaches [29]–[33], [35], [36] solve for the support $\boldsymbol{s}$ and the sparse signal $\boldsymbol{x}$, separately. This is done by performing the following non-recursive two-step estimation shown in Algorithm 2.3.

The support estimation problem Eq. (2.15) is analogous to the best k-term structured sparsity approximation Eq. (2.9) of the deterministic approaches. The estimated support identifies the subspace of the sparse signal solution. Unlike the best k-term structured sparsity approximation, this support estimation only searches for the

Algorithm 2.3 Non-recursive two-steps estimation for solving Eq. (2.14)

```
Input: Measurements $\boldsymbol{y}$, random matrix $\boldsymbol{A}$, and the involved model parameters for $p(\boldsymbol{s})$ and $p(\boldsymbol{y}|\boldsymbol{s})$.
  1. The support is estimated from solving MAP problem:
       $\hat{\boldsymbol{s}} = \max_{\boldsymbol{s}\in\{0,1\}^N} p(\boldsymbol{s}|\boldsymbol{y}) \propto p(\boldsymbol{y}|\boldsymbol{s})p(\boldsymbol{s}),$    (2.15)
  2. Given the support, the sparse signal is obtained from solving the MAP problem Eq. (2.14).
Output: Recovered $\hat{\boldsymbol{x}}$.
```

candidate that provides the highest posterior density without imposing a constraint on the geometrical structure of the solution sparse signal. Therefore, this probabilistic approach can avoid false exclusion problems, which is a major advantage over the deterministic approaches. With recent advances in computer visions, many efficient probabilistic models with the high flexibility and tremendous representation power have been developed. Most of these models capture the underlying structure of training examples.

We briefly discuss three broad classes of these probabilistic models in the following paragraphs:

- **Markov random fields (MRFs)** have been exploited for their flexibility to model various types of signal structures [29]–[36]. Most of the MRFs employed in these works consists of pairwise and unary potentials which are powerful enough to represent a variety of sparse signal structures. The parameters and the underlying graph of MRFs are learned from training data. Many researches have been developed to efficiently learn a fully connected MRF models (see Section 2.4.2). While the representation power of MRF is high in general, the performance of trained MRFs are limited to the representativeness of training data.

- **Deep learning networks** (DNN) can be used either (i) to model the underlying structure of the latent sparse signal or (ii) to decode the information of the sparse signals from a few measurements [57]–[62]. For the former case (i), Restricted Boltzmann machine [57]–[59] is often used to model signal structures. Its modelling performance can be leveraged by adding hidden units. For the latter cases (ii), the encoders of autoencoders [60]–[62] are employed to compress a sparse signal. Given a few measurements, the sparse signals can be reconstructed from the decoder of the autoencoders. Compared to the MRFs, DNNs require much more amount of training data.

- **Clustered-sparsity models** are extended from the deterministic group/block structure sparsity models. Clustered-sparsity models assume that the non-zero coefficients group in clusters [37]–[44]. Mixture models, such as Gaussian-Bernoulli [38], [39] or Gaussian-inverse Gamma [40], [41], are used to model the sparse solution. Among these works, the approaches in [42]–[44] employ MRFs, but these MRFs contains only pairwise potential. The model parameters of these mixture models can be estimated directly from the measurements using EM algorithms [63]. Compared to the MRFs and DNNs, the clustered-sparsity models are more adaptive to the test signals as they *do not rely on any training data*. However, due to the limited signal structure assumption, the clustered sparsity models lack the flexibility to represent different signal structures aside from assumed clustered structures.

These probabilistic structured sparsity models have different advantages and disadvantages. Although MRFs and DNNs can flexibly represent different sparse signals, the quality of the learned MRFs and DNNs rely on the amount of training examples. Although MRFs do not require as many training examples as DNNs, the trained MRFs are effective only when those of training data can well represent the structure of testing data. The clustered-sparsity model does not require any training as its model parameters can be estimated from measurements directly. Nonetheless, the clustered sparsity models have two important limitations due to the limited signal structure assumption—it is not as flexible as the MRF or DNNs, and its underlying graph is fixed and cannot adapt to a new structure. However, the structure of the sparse representation can be diverse across different applications. Moreover, the signals from the same data sources can exhibit a large variability between them. This become the motivation to our proposed adaptive MRF with high flexibility and adaptability to capture different signal structures without relying on any training data, as presented in Chapter 3 and Chapter 4.

The theoretical and experimental discussions in this chapter and subsequent chapters will focus on the MRF and the clustered-sparsity models, which are directly related to the improvement of our proposed adaptive MRF. It is worth mentioning that our objective and respective model are significantly departed from the DNNs; the objective of our approach is to leverage the flexibility and adaptability of an probabilistic model. Thus, our approach does not require any training examples to estimate the adaptive MRF, but these DNNs require extensive training examples to learn the DNN. Without training phase, our proposed model can be flexibly applied to any pairs of measurement matrices and the sparse signal transformation, but the DNNs require specific training settings suitable for different measurement matrices and the sparse signal transformation [60].

If the probabilistic structured sparsity model is ineffective, the support estimation Eq. (2.15) suggests that the performance could be similar to when none of the model is used. This is because the probabilistic structured sparsity models does not put any restriction on the solution spaces. Thus, using the probabilistic structured sparsity models are still much less restrictive than the deterministic structured sparsity models.

Despite the advantage over the deterministic structured sparsity models, solving the MAP problem of the support estimation Eq. (2.15) exactly is, nonetheless, computationally expensive, i.e., it requires exhaustive calculation to compute the value of $p(\boldsymbol{s}|\boldsymbol{y}) \propto p(\boldsymbol{y}|\boldsymbol{s})p(\boldsymbol{s})$ for every possible support. The existing studies proposed different methods to estimate the supports efficiently. In the following, we explore signal recovery with probabilistic structured sparsity models. We will focus on two main types of probabilistic structured sparsity models [29], [64] that have been used as the prior $p(\boldsymbol{s})$: the clustered sparsity models [37]–[42] and the graphical sparsity models [29]–[33], [35], [36]. Gibbs [31] and MAP-OMP [32] are examples of signal recovery with a graphical sparsity model. Bernoulli [39] is an example of signal recovery with a clustered sparsity model.

2.2.2.1 Gibbs

The work in [31] is an early work that employs a Markov random field (MRF) as the graphical sparsity model for its flexibility in capturing a wide variety of signal structures. This work follows the non-recursive two-step estimation (Algorithm 2.3): (i) first, it obtains the support from solving Eq. (2.15), and (ii) given the support, the sparse signal is obtained from solving the MAP problem Eq. (2.14). Since a Gibbs sampling approach is employed to solve Eq. (2.15), this work has been referred to as Gibbs in [32] and in this literature as well.

This approach employs an MRF to capture the support distribution as the prior. The MRF captures the support distribution by defining a probability over an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with a node set $\mathcal{V}$ and set of undirected edges $\mathcal{E}$. In the MRF, a Boltzmann machine (BM) is employed to model the support distribution $p(\boldsymbol{s})$, defined as:

$$p(\boldsymbol{s}) = \frac{1}{Z} \exp\left(\sum_{i \in \mathcal{V}} b_i s_i + \sum_{(i,j) \in \mathcal{E}} w_{i,j} s_i s_j \right), \tag{2.16}$$

where Z is the normalizing constant. Here, each b_i defines the bias toward zero for each support coefficient s_i; meanwhile, $w_{i,j}$ characterizes the interaction between each pair of support coefficients s_i, s_j whose connection is defined by the edge set $\mathcal{E}$. The parameters and structures of the MRF are learned from abundant training examples in the training phase.

This work proposes to solve this MAP problem with a Gibbs sampling technique where the simulated annealing [65] is employed to search for the support that maximizes the posterior distribution, while the value of the posterior distribution for each immediate support estimate is calculated based on a Gibbs sampling approach [66]. The process of solving MAP problems with Gibbs sampling and simulated annealing is summarized in Algorithm 2.4. In the Gibbs sampling procedure, the transition probability for the i^{th} node that changes its value from s_i to s_i^+ at temperature T is given by

$$p(s_i \rightarrow s_i^+ | \boldsymbol{s}_{-i}, \boldsymbol{y}) = \left(1 + \exp\left(-\frac{\Delta E_{\boldsymbol{y}}}{T} \right) \right)^{-1} \tag{2.17}$$

where $\boldsymbol{s}_{-i} = [s_i]_{j \in [N] \setminus i}$ denotes the vector that contains all the support coefficients except s_i, and $\Delta E_{\boldsymbol{y}} = E_{\boldsymbol{y}}(s_i, \boldsymbol{s}_{-i}) - E_{\boldsymbol{y}}(s_i^+, \boldsymbol{s}_{-i})$ is the difference in energy in the next and the current iteration in the transition probability. The Sherman-Morrison formula is employed to facilitate the computation for the transition probability:

$$\Delta E_{\boldsymbol{y}} = \frac{1}{2}\frac{\alpha(\boldsymbol{y}^T \boldsymbol{C}_{\boldsymbol{y}}^{-1} \boldsymbol{a}_i)^2}{1+\alpha \boldsymbol{a}_i^T \boldsymbol{C}_{\boldsymbol{y}}^{-1} \boldsymbol{a}_i} - \frac{1}{2}\log\left(1+\alpha \boldsymbol{a}_i^T \boldsymbol{C}_{\boldsymbol{y}}^{-1} \boldsymbol{a}_i\right) + (s_i^+ - s_i)\left(\sum_{j\neq i} w_{ij} + b_i\right), \quad (2.18)$$

where $\alpha = \frac{1}{2}(s_i^+ - s_i)[\boldsymbol{\Sigma}_{\boldsymbol{x_S}}]_{i,i}$ and $[\boldsymbol{\Sigma}_{\boldsymbol{x_S}}]_{i,i}$ is the variance of the i^{th} sparse signal coefficient, and $\boldsymbol{C}_{\boldsymbol{y}}$ is the covariance matrix of a Gaussian density function for $p(\boldsymbol{s}|\boldsymbol{y})$. If a new state is accepted, the covariance matrix is updated as follows:

$$\boldsymbol{C}_{\boldsymbol{y}}^{+^{-1}} = \boldsymbol{C}_{\boldsymbol{y}}^{-1} - \alpha \boldsymbol{C}_{\boldsymbol{y}}^{-1} \boldsymbol{a}_i (1+\alpha \boldsymbol{a}_i^T \boldsymbol{C}_{\boldsymbol{y}}^{-1} \boldsymbol{a}_i)^{-1} \boldsymbol{a}_i^T \boldsymbol{C}_{\boldsymbol{y}}^{-1}, \quad (2.19)$$

where $\boldsymbol{C}_{\boldsymbol{y}}^0 = \sigma_n \boldsymbol{I} + \sum_{i=1}^N [\boldsymbol{\Sigma}_{\boldsymbol{x_S}}]_{i,i} \boldsymbol{a}_i \boldsymbol{a}_i^T$.

Although the Gibbs sampling [66] is known to be an efficient approach to solving the MAP problem [31]; it can suffer severely from slow convergence and stick at a local minima [32], [67]. As a consequence, this work requires high runtime in general. Because the two step estimation is non-recursive (see Algorithm 2.3), the error due to the local minima problem in support estimation can be accumulated into the sparse signal estimation step. The error accumulation and slow convergence problems become obvious when using Gibbs [31] to estimate a sparse signal in the Two-step-Adaptive MRF, as shown in the convergence of the recovery accuracy and runtime, Figures 3.9 and 3.10, where Gibbs requires the highest number of iterations and provides low accuracy.

2.2.2.2 MAP-OMP

Similar to Gibbs, MAP-OMP [32] solves for sparse signals using the non-recursive two-step estimation (Algorithm 2.3) where the probability distribution of the signal support is modeled with the graphical sparsity model Eq. (2.16). Nevertheless, MAP-OMP [32] proposed heuristic approaches to solve the MAP problem Eq. (2.15), where the parameters and the underlying graph of the MRF are learned from abundant training examples in the training phase.

The proposed heuristic approach attempts to solve the MAP problem Eq. (2.15) by searching for the group of support coefficients that maximize the objective function of

Algorithm 2.4 Support estimation using Gibbs sampling.

```
Input: Measurements $\boldsymbol{y}$, a measurement matrix $\boldsymbol{A}$, and BM parameters $\{W_{i,j}, b_i\}_{(i,j)\in\mathcal{E}, i\in\mathcal{V}}$ and underlying graph of the MRF $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, $\sigma_n$, and $\boldsymbol{\Sigma}_{\boldsymbol{x}_{\boldsymbol{S}}}$.
Initialization : $\boldsymbol{s} = \boldsymbol{s}_0$ and $t = 1$.
  for $t = 1, ..., t_{max}$ do
      1. Assign a temperature value:
         $T \leftarrow$ temperature $(t/t_{max})$;
      2. Pick the next support
      for $i = 1, ..., N$ do
         Pick a binary value for each $s_i^+$ at random;
      end for
      3. Computing transition probability and update support
      for $i = 1, ..., N$ do
         Computing $p(s_i \to s_i^+ | \boldsymbol{s}_{-i}, \boldsymbol{y}; T)$ Eq. (2.17)
         if $p(s_i \to s_i^+ | \boldsymbol{s}_{-i}, \boldsymbol{y}; T) \geq$ random$(0, 1)$ then
             Accept the new state $s_i = s_i^+$ and update $\boldsymbol{C}_{\boldsymbol{y}}^+$ Eq. (2.19).
         end if
      end for
  end for
Output: Recovered support $\boldsymbol{s}$.
```

Eq. (2.15). However, instead of calculating the objective function in Eq. (2.15) exactly, MAP-OMP searches for the support that maximizes a pseudo function approximating the conditional distribution of a chosen support coefficient given other supports and measurements $p(s_i|\boldsymbol{s}^k, \boldsymbol{y})$:

$$q(i, \boldsymbol{s}^k) = \frac{1}{2\sigma_n}\boldsymbol{y}^T\boldsymbol{A}_{\boldsymbol{s}^k}\boldsymbol{Q}_{\boldsymbol{s}^k}^{-1}\boldsymbol{A}_{\boldsymbol{s}^k}^T\boldsymbol{y} - \frac{1}{2}\ln\left(\det\left(\boldsymbol{Q}_{\boldsymbol{s}^k}\right)\right) + 2b_i + 2\sum_j w_{i,j}s_j^k - \frac{1}{2}\ln\left([\boldsymbol{\Sigma}_{\boldsymbol{x}_{\boldsymbol{s}^k}}]_{i,i}\right)$$
$$\propto \ln\left(p(s_i|\boldsymbol{s}^k, \boldsymbol{y})\right) \tag{2.20}$$

where $\boldsymbol{Q}_{\boldsymbol{s}^k} = \boldsymbol{A}_{\boldsymbol{s}^k}^T\boldsymbol{A}_{\boldsymbol{s}^k} + \sigma_n\boldsymbol{\Sigma}_{\boldsymbol{x}_{\boldsymbol{s}^k}}^{-1}$. The entire procedure is illustrated in Algorithm 2.5. The procedure performs until the value of $p(\hat{s}^k|\boldsymbol{y})$ in the current iteration decreases.

Notice that unlike Gibbs [31], MAP-OMP employs the sparsity of signals as the prior to update the support and is thus more efficient than Gibbs [32]. MAP-OMP is a very efficient MRF-based approach that has consistently yielded state-of-the-art results in recovery accuracy to date. Nevertheless, similar to Gibbs [31], the non-recursive two-step estimation (Algorithm 2.3) is prone to two problems: (i) it can be time-consuming in performing support estimation, and then, sparse signal estimation, and (ii) the error in the support estimation step can propagate to the sparse signal estimation, where the error cannot be fixed later. Note that the MAP-OMP

Algorithm 2.5 MAP-OMP.

```
Input: Measurements $\boldsymbol{y}$, a measurement matrix $\boldsymbol{A}$, and the BM parameters $\{W_{i,j}, b_i\}_{(i,j)\in\mathcal{E}, i\in\mathcal{V}}$ and the underlying graph of the MRF $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, $\sigma_n$, and $\boldsymbol{\Sigma}_{\boldsymbol{x}_s}$.
Initialization : $\hat{\mathbf{s}} = -\mathbf{1}^{N\times 1}$, $\hat{S} = \emptyset$, and $t = 1$.
  repeat
     1. Pick the next support vector
     for $i \notin S^{k-1}$ do
        $S^k \leftarrow \hat{S}^{k-1} \cup \hat{\imath}$
        $s_j^k = \begin{cases} \hat{s}_j^{k-1}, & j \neq \hat{\imath} \\ 1, & j = \hat{\imath} \end{cases}$ .
        Evaluate $q(i, \boldsymbol{s}^k)$ using Eq. (2.20)
     end for
     2. Search for the coefficient that maximizes the distribution
     $\hat{\imath} \leftarrow \max_{i\in N}\{q(i, \boldsymbol{s}^k)\}$
     3. Merge the new non-zero index to the existing index set
     $\hat{S}^k \leftarrow \hat{S}^{k-1} \cup \hat{\imath}, \quad \hat{s}_j^k = \begin{cases} \hat{s}_j^{k-1}, & j \neq \hat{\imath} \\ 1, & j = \hat{\imath} \end{cases}$ .
     4. Increase iteration
     $t = t + 1$.
  until $p(\hat{\mathbf{s}}^k|\boldsymbol{y}) \leq p(\hat{\mathbf{s}}^{k-1}|\boldsymbol{y})$.
Output: Recovered support $\hat{\mathbf{s}}$.
```

has to calculate Eq. 2.20 up to N times in each iteration to pick up the next support vector and search for the coefficient that maximizes the distribution, which can cause high computation when k becomes large. The runtime problem becomes obvious when using MAP-OMP to estimate the sparse signal in the Two-step-Adaptive MRF framework, as shown in our experimental results in Figures 3.9 and 3.10.

More importantly, in both of these MRF-based approaches, the MRF is obtained from training and cannot adapt for new signal structures. Thus, their performance can deteriorate obviously when the structure of the testing signals is different from those of the training signals. Next, we explore an example of signal recovery using the clustered sparsity model that can be adaptive to testing signals.

2.2.2.3 Bernoulli

The work in [39] exploits the structure of the sparse signals through the clustered sparsity model. A beta-Bernoulli model is used to model the distribution of the signal coefficients that are assumed to cluster in groups. Thus, we refer to the clustered sparsity model and the method in [39] as *Bernoulli* in our study. The model is defined as follows:

$$p(s_i|b_i) = \text{Bernoulli}(s_i|b_i), \ \text{ where } \ b_i = b_k, \ \text{ with } \ b_k \sim \text{Beta}(\alpha, \beta) \ \ \forall i \in \mathbb{N}_k. \tag{2.21}$$

where α and β are constants with appropriate settings. Each b_i enforces bias toward zero to each coefficient in each overlapping neighborhood $\mathbb{N}_k$ to exhibit clustering structure. Unlike the previous MRF, the beta-Bernoulli model is only suitable for modelling the sparse signals whose coefficients are grouped into clusters.

Although this model is not as flexible as the MRF, the conjugate property of the beta-Bernoulli model allows an efficient expectation maximization (EM) technique [63] to estimate the support $\boldsymbol{s}$ as well as the model parameters b_i. Let $\boldsymbol{\lambda} = \{\boldsymbol{s}, \boldsymbol{t}\}$ denote the set of the unknown variables, and $\boldsymbol{\Theta} = \{b_1, ..., b_N\}$ denote the set of the unknown parameters. To estimate the parameters, the following MAP problem is considered [39]:

$$\max_{\boldsymbol{\theta}} p(\boldsymbol{y}|\boldsymbol{\Theta}) \ = \int p(\boldsymbol{y}, \boldsymbol{\lambda}|\boldsymbol{\Theta}) \mathrm{d}\boldsymbol{\lambda} \tag{2.22}$$

which can be solved efficiently with a variational expectation maximization (EM) method [63] by introducing $q_{\boldsymbol{\lambda}}(\boldsymbol{\lambda})$ and $q_{\boldsymbol{\Theta}}(\boldsymbol{\Theta})$. Eq. (2.22) can be rewritten in log-likelihood as

$$\ln p(\boldsymbol{y}; \boldsymbol{\Theta}) \propto F(q_{\boldsymbol{\lambda}}(\boldsymbol{\lambda}), q_{\boldsymbol{\Theta}}(\boldsymbol{\Theta})) + KL(q_{\boldsymbol{\lambda}}(\boldsymbol{\lambda}), q_{\boldsymbol{\Theta}}(\boldsymbol{\Theta}) || p(\boldsymbol{\lambda}, \boldsymbol{\Theta}|\boldsymbol{y})), \tag{2.23}$$

where

$$F(q_{\boldsymbol{\lambda}}(\boldsymbol{\lambda}), q_{\boldsymbol{\Theta}}(\boldsymbol{\Theta})) = \int q_{\boldsymbol{\lambda}}(\boldsymbol{\lambda}), q_{\boldsymbol{\Theta}}(\boldsymbol{\Theta}) \ln \frac{p(\boldsymbol{\lambda}, \boldsymbol{\Theta}|\boldsymbol{y})}{q_{\boldsymbol{\lambda}}(\boldsymbol{\lambda}), q_{\boldsymbol{\Theta}}(\boldsymbol{\Theta})} \mathrm{d}\boldsymbol{\lambda} \tag{2.24}$$

and the Kullback-Leibler divergence $KL(\cdot||\cdot)$ is always greater than or equal to zero.

Since the left hand-side of Eq. 2.23 is independent of $\boldsymbol{\lambda}$ and $\boldsymbol{\Theta}$, maximizing $F(q_{\boldsymbol{\lambda}}(\boldsymbol{\lambda}), q_{\boldsymbol{\Theta}}(\boldsymbol{\Theta}))$ with respect to $q_{\boldsymbol{\lambda}}$ and $q_{\boldsymbol{\Theta}}$ is equivalent to maximizing KL. Therefore, $q_{\boldsymbol{\lambda}}$ and $q_{\boldsymbol{\Theta}}$ represent approximations to the posterior distribution $p(\boldsymbol{\lambda}, \boldsymbol{\Theta}|\boldsymbol{y})$. $q_{\boldsymbol{\lambda}}$ and $q_{\boldsymbol{\Theta}}$ are estimated by performing the following expectation-maximization steps [63]:

- **Expectation step** updates the random variables $\boldsymbol{\lambda}$ with the following rules.

$$q^{t+1}_{\boldsymbol{\lambda}_i}(\boldsymbol{\lambda}_i) \propto \exp\left\langle \ln p(\boldsymbol{\lambda}_i, \boldsymbol{y}|\boldsymbol{\Theta})\right\rangle_{q^t_{\boldsymbol{\Theta}}(\boldsymbol{\Theta}), \prod_{j\neq i} q_{\lambda_j}(\lambda_j)} \tag{2.25}$$

and $q_{\boldsymbol{\lambda}}(\boldsymbol{\lambda}) = \prod_{i=1}^{N} q_{\boldsymbol{\lambda}_i}(\boldsymbol{\lambda}_i)$. The sparse signal and the support can be updated in the expectation step. The update for support is:

$$q^{t+1}(s_i) \propto \exp\left\langle \ln p(\boldsymbol{s}|\boldsymbol{b})p(\boldsymbol{y}|\boldsymbol{s},\boldsymbol{x})\right\rangle_{q^t_{\boldsymbol{b}}(\boldsymbol{b})q^t_{\boldsymbol{x}}(\boldsymbol{x})q^t_{\boldsymbol{s}\backslash s_i}(\boldsymbol{s}\backslash s_i)} \tag{2.26}$$

Then,

$$\hat{s}_i = \frac{q(s_i = 1)}{q(s_i = 1) + q(s_i = 0)} \tag{2.27}$$

where $q(s_i = 0) = \exp\left\langle \ln(1 - b_i)\right\rangle_{q_{b_i}(b_i)}$ and $q(s_i = 1) = \exp(-\sigma_n(\boldsymbol{y}^T\boldsymbol{y} + \langle x_i^2\rangle \boldsymbol{a}_i^T\boldsymbol{a}_i - 2\hat{x}_i\boldsymbol{a}_i^T(\boldsymbol{y} - \sum_{i>j}\boldsymbol{a}_j\hat{x}_j\hat{s}_j))) \exp\left\langle \ln(b_i)\right\rangle_{q_{b_i}(b_i)}$.

The sparse signal is updated as follows:

$$q^{t+1}(\boldsymbol{x}) \propto \exp\left\langle \ln p(\boldsymbol{x})p(\boldsymbol{y}|\boldsymbol{s},\boldsymbol{x})\right\rangle_{q^t_{\boldsymbol{s}}(\boldsymbol{s})} \tag{2.28}$$

Then, $p(\boldsymbol{x}|\boldsymbol{y})$ is a Gaussian distribution $\mathcal{N}(\mu^t_p, \boldsymbol{C}^{t\,-1}_p)$

$$\hat{\boldsymbol{x}} = \mu^t_p. \tag{2.29}$$

where $\boldsymbol{u}^t_p = \hat{\sigma}_n\boldsymbol{C}_t^{-1}\hat{\boldsymbol{S}}\boldsymbol{A}^T\boldsymbol{y}$ and $\boldsymbol{C}^t_p = \hat{\boldsymbol{\Sigma}}_t + \hat{\sigma}_n\langle\boldsymbol{S}\boldsymbol{A}^T\boldsymbol{A}\boldsymbol{S}\rangle_{q_{\boldsymbol{S}}(\boldsymbol{s})}$ where $\hat{\sigma}_n$ and $\hat{\boldsymbol{\Sigma}}_t$ are predefined noise and signal variance.

- **Maximization step** updates the parameters $\boldsymbol{\Theta}$ with the following update rules:

$$q^{t+1}_{\Theta_i}(\Theta_i) \propto p(\boldsymbol{\Theta})\exp\left(\left\langle \ln p(\boldsymbol{\lambda}, \boldsymbol{y}|\boldsymbol{\Theta})\right\rangle_{q^t_{\boldsymbol{\lambda}}(\boldsymbol{\lambda})}\right. \tag{2.30}$$

The model parameters are updated in maximization steps. They are calculated by performing the following updates.

$$q^{t+1}(\boldsymbol{b}) \propto \exp\left\langle \ln p(\boldsymbol{s}|\boldsymbol{b})p(\boldsymbol{b})\right\rangle_{q^t_{\boldsymbol{s}}(\boldsymbol{s})} = Beta(\boldsymbol{b}|\hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\beta}}) \tag{2.31}$$

where $\hat{\alpha} = \alpha + \sum_{i\in\mathbb{N}_k}\hat{s}_i$ and $\hat{\beta} = \beta + |\mathbb{N}_k| - \sum_{i\in\mathbb{N}_k}\hat{s}_i$.

From this example, the model parameters can be adapted to testing signals and can effectively estimate the support with closed-form update formulations. However, beta-Bernoulli are only suitable for these signals with a clustering structure. This becomes the motivation of our research: to bring adaptability to the MRF, which has a higher flexibility to capture the broad range of the structure of sparse signals.

2.3 Sample complexity

Sample complexity represents the minimum number of measurements that is required to achieve successful signal recovery in CS. This section discusses how the sample complexity can be reduced with deterministic and probabilistic structured sparsity models. First, we discuss the relationship between the sample complexity and the restricted isometry property (RIP) in Section 2.3.1. Here, we are only interested in RIP for the sub-Gaussian matrix. Then, we provide the example results of sample complexity with deterministic structured sparsity models in Section 2.3.2. The relationship between the sample complexity and probabilistic RIP is discussed in Section 2.3.3. The sample complexity with using probabilistic structured sparsity models is then discussed in Section 2.3.4.

2.3.1 Sample complexity and restricted isometry property (RIP)

To guarantee that a sparse signal can be successfully recovered in CS, two important properties are held by the linear compression: (i) there is a unique solution— $\boldsymbol{A}\boldsymbol{x}_1 \neq \boldsymbol{A}\boldsymbol{x}_2$ for all sparse signal pairs $\boldsymbol{x}_1, \boldsymbol{x}_2$—,and (ii) that the linear compression can stably embed under bounded noise—the Euclidean distance between each pair of k-sparse signals are preserved by linear projections. To guarantee this, the measurement matrix $\boldsymbol{A}$ involved in the linear compression must satisfy the restricted isometry property (RIP) that ensures the two properties. The definition of the RIP is as follows:

Definition 2.3.1. Let $\boldsymbol{A}$ be an $M \times N$ matrix. $\boldsymbol{A}$ satisfies a restricted isometry property (RIP) of order k, if there exists a bounded restricted isometry constant $\delta_k \in (0, 1)$ such that

$$(1 - \delta_k)||\boldsymbol{x}||_2^2 \leq ||\boldsymbol{A}\boldsymbol{x}||_2^2 \leq (1 + \delta_k)||\boldsymbol{x}||_2^2, \tag{2.32}$$

for all sparse signals $\boldsymbol{x} \in \mathcal{U}_k$.

There are certain types of matrices that are known to satisfy RIP. Sub-Gaussian matrices are a group of random matrices that satisfy RIP with overwhelming probability [68], if the number of measurements M meets a requirement of sample complexity. The relationship between the sample complexity and the RIP of sub-Gaussian matrices is as follows:

Theorem 2.3.1. *([68] , Theorem 5.2). For any fixed sparse signal $\boldsymbol{x}$, if the sub-Gaussian random matrix $\boldsymbol{A}$ satisfies the following concentration inequality:*

$$P\left(|||\boldsymbol{Ax}||_2^2 - ||\boldsymbol{x}|||_2^2| < \varepsilon||\boldsymbol{x}||_2^2|\right) \geq 1 - 2e^{-c\varepsilon M/2}, \tag{2.33}$$

then the matrix $\boldsymbol{A}$ satisfies RIP with probability of at least $1 - 2\left(\frac{12}{\delta_k}\right)^k e^{\frac{-c\delta_k M}{2}}$.

Then, a measurement matrix $\boldsymbol{A}$ will satisfy the RIP for all sparse signals with a probability higher than $1 - 2L\left(\frac{12}{\delta_k}\right)^k e^{\frac{-c\delta_k M}{2}}$ [49], where L is the cardinality of the sparse signal space, if M complies with the following requirement for sample complexity, i.e. [49]

$$M \geq \frac{2}{c\delta_k}\left(\log(2L) + k\log\left(\frac{12}{\delta_k}\right) - t\right), \tag{2.34}$$

for any $t > 0$.

The sample complexity can be generalized as $\mathcal{O}(k + \log L)$ [49]. The cardinality of the simple sparsity model Eq.(2.2) is $L = \binom{N}{k}$. From the Stirling formulation [49], $L = \binom{N}{k} \leq \left(\frac{Ne}{k}\right)^k$. Then, $\mathcal{O}(k + \log L) = \mathcal{O}(k + k\log(\frac{N}{k})) \approx \mathcal{O}(k\log(\frac{N}{k}))$. This opens up a direction to reduce the sample complexity by restricting the cardinality of signal space L.

This direction is employed to reduce the sample complexity for deterministic structured sparsity models.

2.3.2 Reducing sample complexity with deterministic structured sparsity models

Deterministic structured sparsity models impose additional geometrical assumptions to restrict the cardinality of signal space L. In the following, we provide examples of sample complexity and the reduced cardinality with each deterministic structured sparsity model that has been discussed in Section 2.2.1:

- **Block sparse/group sparsity model** [13] reduces the sparse signal space by assuming that the sparse signal $\boldsymbol{x} \in \mathcal{R}^N$ can be reshaped into a matrix $\boldsymbol{X}$ of size $n \times m_k$ that has p entirely non-zero columns. The sparsity of $\boldsymbol{X}$ is k $(= np)$. A set of m_k-block sparse signals is defined as follows [13]:

$$\mathcal{B}_s = \{\boldsymbol{X} = [\boldsymbol{x}_1, ...\boldsymbol{x}_{m_k}] \in \mathbb{R}^{n\times m_k} : \ \boldsymbol{x}_i \in \mathcal{R}^n \ \text{ for } i \in \Omega, \text{and } \ \boldsymbol{x}_j = \mathbf{0}, \text{ otherwise}, \\ \Omega \subset [m_k], \text{ and } |\Omega| = k\} \tag{2.35}$$

From definition, the cardinality of the k-block sparse signal set is reduced to $|\mathcal{B}_s| = \binom{m_k}{p} = \binom{N/n}{k/n} < \left(\frac{eN/n}{k/n}\right)^{k/n}$. With the cardinality, the sample complexity is $\mathcal{O}(\frac{k}{n}\log(\frac{N}{k}))$ [13].

- **Hierarchical structured sparsity model** [13], [49] reduces the sparse signal space by assuming that non-zero coefficients in $\boldsymbol{x}$ can be constrained to form a tree-structure. The model is defined in Eq. (2.9) where all elements in a sparse signal $\boldsymbol{x}$ form binary tree, and only k non-zero coefficients can form a rooted subtree. Each subtree with k nodes defines a subspace. The total number of subspaces is bounded by the total number of different trees with k nodes, which is the Catalan number $C_k = \frac{1}{k+1}\binom{2k}{k} \leq \frac{(2e)^k}{k+1}$. With the reduced cardinality, the sample complexity is $M = \mathcal{O}(k + k\log(2e) - \log(k+1)) \approx \mathcal{O}(2k)$ [13], [49].

- **Graph sparsity model** [27], [48] reduces the sparse signal space by assuming that the sparsity of $\boldsymbol{x}$ can be constrained with the weighted graph model defined by Eq. (2.10). The cardinality of the set of the weighted graph model is bounded by the total sparsity of signal k, the maximum number of connected components g formed by the forest in the graph G, and the bound of the total weight $w(F)$ corresponding to the edges in the forest, which are less than $\binom{N}{g}\binom{B+k-g-1}{g}\rho(G)^{k-g}\binom{2k-g}{s-1}$ [27]. Hence, the sample complexity $\mathcal{O}(k + k(\log\rho(G) + \log\frac{B}{k}) + g\log\frac{N}{g})$. The full derivation of the cardinality of the model is referred to [27].

It can be seen that the sample complexity has been greatly reduced with the deterministic structured sparsity models. However, as we have discussed, for the examples of signal recovery using deterministic models in Section 2.2.1, imposing geometrical assumptions can cause some good candidate signals to be excluded from solution spaces, especially when the structure of the testing signals is different from the assumed geometrical structure. Therefore, many studies [29]–[42] resort to probabilistic structured sparsity models that focuses on the likelihood of the signal structure rather than the exact geometrical structure of the signal.

In the following, we discuss how the sample complexity can be reduced by the probabilistic structured sparsity models.

2.3.3 Sample complexity and probabilistic RIP (PRIP)

The prior probability distribution of sparse signals helps identify where the important information lies in the solution space in signal recovery. Cevher *et al.* [29] introduced the following lemma that establishes the relationship between the signal probability and the sample complexity:

Lemma 2.3.2. *([29] , Lemma 1.). Suppose that $\delta_k \in [0, 1]$ are given, and the signal $\boldsymbol{x}$ is generated by a known probabilistic model $\mathcal{P}$. Let $\Omega_{k,\varepsilon} \subseteq \mathcal{U}_k$ denote the smallest set of support for which the probability that a k-sparse signal $\boldsymbol{x}$ has supp$(\boldsymbol{x}) \notin \Omega_{k,\varepsilon}$ is less than ε, and denote the cardinality of the support set as $D = |\Omega_{k,\varepsilon}|$. A sub-Gaussian random matrix $\boldsymbol{A} \in \mathcal{R}^{M\times N}$ satisfies the (k, ε)-PRIP with a probability of at least $1 - e^{-c_2 M}$, if $M \geq c_1(k + \log(D))$, where $c_1, c_2 > 0$ depends only on the PRIP constant δ_k,*

where the definition of the probabilistic RIP (PRIP) is as follows.

Definition 2.3.2. [29] A matrix $\boldsymbol{A}$ satisfies (k, ε)-PRIP, if there exists a constant $\delta_k > 0$, such that for a k-sparse signal $\boldsymbol{x}$ generated by a specified probabilistic signal model $\mathcal{P}$, the random matrix $\boldsymbol{A}$ satisfies RIP with a probability of at least $1 - \varepsilon$ over the signal probability space.

The extra condition of the support set $\Omega_{k,\varepsilon}$ in Lemma 2.3.3 is crucial to reducing the sample complexity. It implies that the support set $\Omega_{k,\varepsilon}$ does not need to contain every support from the probabilistic model $\mathcal{P}$; however, only those that have high likelihood [29]. If this condition is satisfied, then RIP holds with the similar requirement of sample complexity, i.e., $\mathcal{O}(k + \log(D))$ where D is the cardinality of the support set $\Omega_{k,\varepsilon}$.

2.3.4 Reducing sample complexity with probabilistic structured sparsity models

In practice, the probabilistic model $\mathcal{P}$ of sparse signals is often unknown. Previous studies [29]–[42] either assume clustered sparsity models suitable for many signal applications [37]–[42], or they employ the flexible graphical sparsity model from training data [29]–[36]. In the following, we provide examples of how the sample complexity is reduced with a clustered sparsity model and discuss the sample complexity by using graphical sparsity models.

- **Clustered sparsity model** [29]. Suppose that non-zero coefficients of k-sparse signals follow a homogeneous Poisson process with rate $\lambda = -\log(\frac{\varepsilon}{k})N^{-\alpha}$. N^{α} is the duration where a non-zero coefficient occurs, and $\alpha \ll 1$, since non-zero coefficients cluster together. The probability of the first non-zero coefficient occurs within the distance N^{α} is $1 - \frac{\varepsilon}{k}$, and then, the probability for the next $k-1$ non-zero coefficients to occur within the next $N^{\alpha(k-1)}$ is $(1 - \frac{\varepsilon}{k})^{k-1}$. Thus, this forms a set of supports $\Omega_{k,\varepsilon}$ with cardinality $N^{\alpha k}$, with the probability $(1 - \frac{\varepsilon}{k})^{k}$, which is higher than $(1 - \varepsilon)$. With such cardinality, the sample complexity becomes $\mathcal{O}(k + \alpha k \log(N)) \approx \mathcal{O}(k)$ as $\alpha \ll 1$.

- **Graphical sparsity model** [29]–[36]. These research papers [29]–[36] employ Markov random fields (MRFs) as the structured sparsity model. Unlike the clustered sparsity models, they do not have any geometrical assumptions about the clustering of signal coefficients. Since the geometrical structure is not strictly defined, quantifying the cardinality of the support set $\Omega_{k,\varepsilon}$ is difficult. To our knowledge, the analytical results for the sample complexity reduced by this model is still very limited. Hence, in the following we provide a discussion of the sample complexity when using MRF based on Lemma 2.3.3.

Let $p(\cdot;\hat{\mathbf{\Theta}}_{\mathcal{G}})$ denote the probability distribution of the MRF with parameters $\hat{\mathbf{\Theta}}_{\mathcal{G}}$ and underlying graph $\mathcal{G}$. According to [29]–[36], the parameters of the MRF are learned from solving the following maximum likelihood estimation problem:

$$\hat{\mathbf{\Theta}}_{\mathcal{G}} = \underset{\mathbf{\Theta}_{\mathcal{G}}}{max} p(\mathcal{D}|\mathbf{\Theta}_{\mathcal{G}}). \tag{2.36}$$

$\mathcal{D}$ is the set of training data. It is assumed that this MRF can well represent the true probability model of support signals of the testing signals, i.e. $\boldsymbol{s} \sim p(\cdot|\mathbf{\Theta}_{\mathcal{G}})$. Following the support estimation process Eq. (2.15) in [29]–[36], the candidate supports can be sampled from the posterior distribution. The support set $\Omega_{k,\epsilon}$ is defined as:

$$\Omega_{k,\epsilon} = \{\bar{\mathbf{s}} \in \{0,1\}^N : \ \bar{\mathbf{s}} \sim p(\boldsymbol{s}|\boldsymbol{y}) \propto p(\boldsymbol{y}|\boldsymbol{s})p(\boldsymbol{s}|\hat{\mathbf{\Theta}}_{\mathcal{G}})\}. \tag{2.37}$$

Calculating the number of candidates is, however, difficult, since the analytical form of the posterior distribution $p(\boldsymbol{s}|\boldsymbol{y})$ cannot be obtained; thus, the credible interval cannot be quantified. One can resort to finding the confidence interval [69]; however, finding the confidence interval requires the true parameters of $p(\boldsymbol{s}|\boldsymbol{y})$ to be known. Meanwhile, in practice, these candidates are obtained from solving Eq. (2.15) with MAP estimators that often provide a few solutions [29]–[36]. Thus, the support set can be further restricted according to the estimators employed for solving the supports, i.e.

$$\tilde{\Omega}_{k,\epsilon} = \{\bar{\mathbf{s}} \in \{0,1\}^N : \ \bar{\mathbf{s}} = \text{Estimator}(p(\boldsymbol{y}|\boldsymbol{s})p(\boldsymbol{s}|\hat{\mathbf{\Theta}}_{\mathcal{G}}))\}, \tag{2.38}$$

Thus, if the estimator yields a small, constant number of solutions, the sample complexity can be reduced to the optimal theoretical complexity $\mathcal{O}(k)$, as $|\tilde{\Omega}_{k,\epsilon}|$ is a small constant.

Notice that this resulting sample complexity relies on the assumption that the training data $\mathcal{D}$ can well represent the testing signals, and the learned MRF model is thus representative of the true signal probability distribution. However, if the testing signals are different from those training signals, the learned MRF model does not necessarily represent the true signal probability distribution. As a result, the support set $\tilde{\Omega}_{k,\epsilon}$ fails to capture the support of the testing signal, which violates the extra condition on the support set $\tilde{\Omega}_{k,\epsilon}$ in Lemma 2.3.3. As the structured prior is no longer informative, the sparse signal recovery only relies on the simple sparsity as the prior knowledge. This will result in the requirement on the sample complexity to be as high as for the non-structured cases, i.e., $\mathcal{O}(k \log \frac{N}{k})$.

Data-adaptive models, such as our adaptive MRF (see Chapters 3 and 4), can be used to address this problem, as our adaptive MRF is employed to capture the structure of testing signal. In Chapter 3, we will discuss the essence of data-adaptive prior and the connection between the data-adaptive prior and the sample complexity.

2.4 Markov random fields

So far, we have discussed an application of Markov random fields (MRFs) for signal recovery in CS. The MRFs have been used to model the distribution of the support. Graphical model inferences and learning are two important mechanisms to infer/encode the information in the MRFs. The graphical model inference can be used to solve the support estimation Eq. (2.15). Graphical model learning is employed to learn the MRF parameters, as Eq. (2.36). The choice of graphical model inference and learning are important to construct an efficient signal recovery, especially for the proposed signal recovery method in Chapter 3. In the following, we provide background on the MRFs, including the graphical model inferences and learning.

MRFs [70] provide a principled framework to represent the *interdependency* or *correlation* among subsets of random variables [71], [72]. They have played a crucial role in extensive image processing and computer vision tasks, such as image denoising [73], segmentation [74] , super-resolution [75] , inpainting [76], etc. The MRFs are an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ that consists of sets of nodes $\mathcal{V}$ and undirected edges $\mathcal{E}$, and holds two important properties: (1) the positive joint probability, i.e., $p(\boldsymbol{S}) > 0$, and (2) local Markov property, i.e., $S_i \perp S_{\mathcal{V} \setminus i} | S_{\mathcal{N}_i}$ where $\mathcal{N}_i$ denotes the set of neighbors of node i in the graph $\mathcal{G}$. Let $\boldsymbol{S}$ be a set of random variables where each $\boldsymbol{S}_{C_i}$ is a random variables vector in clique[1] $C_i \in \mathcal{G}$. According the Hammersley-Clifford theorem [77], Gibbs distributions, that is the family of distributions satisfying the local Markov property, can be factorized into the following forms:

$$p(\boldsymbol{S}) = \frac{1}{Z} \prod_{C_i \in \mathcal{G}} f(\boldsymbol{S}_{C_i}; \theta_{C_i}), \tag{2.39}$$

where Z is a normalizing constant. $Z = \sum_{\boldsymbol{s}} \prod_{C_i \in \mathcal{G}} f(\boldsymbol{s}_{C_i}; \boldsymbol{\theta}_{C_i})$. $f(\boldsymbol{S}_{C_i}; \theta_{C_i})$ denotes the factor associated with random variables in each clique C_i where θ_{C_i} is the parameter of the MRFs in each clique.

2.4.1 Inference

Inference in graphical models is a task to infer the information in hidden variables $\boldsymbol{S}_H$, given observed variables $\boldsymbol{S}_O$. Two common types of inference task are:

- (1) to compute $\hat{\boldsymbol{s}}$ that maximizes the posterior probability $p(\boldsymbol{S}_H | \boldsymbol{S}_O)$, i.e.,

  $$\hat{\boldsymbol{s}} = \underset{\boldsymbol{s}}{max} p(\boldsymbol{s}_H | \boldsymbol{s}_O), \tag{2.40}$$

  which is called maximum a posteriori (MAP) estimation; and

- (2) to compute the marginal distribution over a single hidden node i or sets of hidden nodes, i.e., let D denote the index of the nodes of interest

  $$p(\boldsymbol{s}_D | \boldsymbol{s}_O) = \sum_{i \in \mathcal{V} \backslash D} p(s_1, ...s_{|\mathcal{V}|} | \boldsymbol{s}_O). \tag{2.41}$$

It should be noted that the computational cost of Eq.(2.40) and Eq.(2.41) grow exponentially with the number of nodes. If the vector $\boldsymbol{s} \in \{0, 1\}^N$ is a binary vector, then the computational complexity is proportional to 2^N. Efficient algorithms have been developed to improve the computational complexity in graphical model inference.

Generally, graphical model inference can be categorized into: (1) *exact* inference and (2) *approximate* inference. In exact inference, the marginal distribution over each group of node variables is analytically computed. Belief propagation is a standard

[1] A clique is defined as a fully connected subset of nodes in a graph.

example of algorithms for solving exact inference. It is efficient for a certain class of graphical models such as tree-structured models, except when there are cycles formed within the underlying graph [78], since the computational complexity grows with the tree width of the graph. There are polynomial time algorithms for certain classes of MRFs [79], [80]. Still, performing exact inference may not be feasible in practice.

For this reason, approximate inference approaches are preferred. Two common types of *approximate* inference algorithms are (1) sampling methods and (2) variational methods. The idea of a sampling method is to approximate the marginal distribution with samples. Since the distribution of the graphical model is often flexible and cannot be sampled directly, the sampling usually relies on methods such as Markov chain Monte Carlo (MCMC) to generate samples from a simpler distribution that approximates the more complicate one. Gibbs sampling is an alternative to MCMC; however, it can suffer from slow convergence [67]. Simulated annealing is an interesting alternative as it has global convergence properties [81]. However, these sampling-based methods can be computationally expensive under certain conditions [67]. Variational algorithms [82] improve the computational complexity by employing a simpler distribution, where inference can be performed easily, than the underlying distribution of graphical model. The surrogate distribution, however, is restricted to the family of distributions that closely resembles the underlying distribution.

In Chapter 3, our approach requires us to perform inference on an MRF to estimate a signal support from an MAP problem Eq. (3.20). This MAP problem can be solved using any inference technique mentioned previously. Here, we employ an approximate inference technique [83] for its low computational complexity, especially when the underlying graph of the MRF contains cycles.

2.4.2 Learning

Given the underlying graph, the parameters of an MRF are learned from a given set of training data. Let $\mathcal{D} = \{\boldsymbol{s}[1], ..., \boldsymbol{s}[D]\}$ be a set of training data (T is the number of training data); $\boldsymbol{\theta}_{\mathcal{G}} = \{\theta_{C_i}\}_{C_i \in \mathcal{G}}$ is the set of parameters corresponding to the graph $\mathcal{G}$. One of the most popular criteria for parameter learning is to solve the maximum a posteriori problem:

$$\boldsymbol{\theta}_{\mathcal{G}} = \max_{\boldsymbol{\theta}_{\mathcal{G}}} p(\boldsymbol{\theta}_{\mathcal{G}}|\mathcal{D}) \propto p(\mathcal{D}|\boldsymbol{\theta}_{\mathcal{G}})p(\boldsymbol{\theta}_{\mathcal{G}}) = \prod_{d=1}^{D} p(\mathbf{s}^d|\boldsymbol{\theta}_{\mathcal{G}})p(\boldsymbol{\theta}_{\mathcal{G}}), \tag{2.42}$$

where $\mathbf{s}[1], ..., \mathbf{s}[D]$ are assumed to be drawn from an iid distribution. If the prior $p(\boldsymbol{\theta}_{\mathcal{G}})$ is assumed to be a uniform distribution, the MAP problem is reduced to a maximum likelihood (ML) problem. Given the probability distribution of the MRF Eq. (2.39), the ML problem can be written in logarithmic form as follows:

$$\boldsymbol{\theta}_{\mathcal{G}} = \max_{\boldsymbol{\theta}_{\mathcal{G}}} \sum_{d=1}^{D} \log p(\mathbf{s}^d|\boldsymbol{\theta}_{\mathcal{G}}) = \sum_{d=1}^{D} \log \frac{f(\mathbf{s}^d|\theta_{C_1}, ..., \theta_{C_g})}{\sum_{\mathbf{c}} f(\mathbf{c}|\theta_{C_1}, ..., \theta_{C_g})}. \tag{2.43}$$

The estimation of the model parameters are obtained by performing gradient descends. The gradient of the objective function is:

$$\frac{\partial \log p(\mathbf{s}|\theta_{C_1}, ..., \theta_{C_g})}{\partial \theta_{C_i}} = \frac{\partial \log f(\mathbf{s}|\theta_{C_1}, ..., \theta_{C_g})}{\partial \theta_{C_i}} - \sum_{\mathbf{c}} p(\mathbf{c}|\theta_{C_1}, ..., \theta_{C_g}) \frac{\partial \log f(\mathbf{c}|\theta_{C_1}, ..., \theta_{C_g})}{\partial \theta_{C_i}}. \tag{2.44}$$

The second term in the right-hand side in Eq. (2.44) is the expected derivative of the logarithm of the probability. Calculating the expected derivative requires performing the difficult inference to compute the marginal probability, which is often computationally expensive. A number of alternative learning criteria have been developed to address this problem, including the maximum pseudo-likelihood [84], contrastive divergence [85], discriminative training of energy-based methods [86], and score matching [87]. These methods proposed different approaches to address the problem of estimating the normalizing term in the MRFs. The pseudo-likelihood [84] employs the conditional independence property of MRFs to factorize $p(\mathbf{s}|\boldsymbol{\theta}_{\mathcal{G}})$ into the product of marginal distribution over a small group of nodes in each clique $p(\{s_i\}_{i \in C_i}|\theta_{C_i})$, which involves only a small number of the parameters. If the normalizing term can be calculated, this approach is efficient. However, if undirected graphical models can have high tree width, calculating the normalizing term is intractable, except in the Gaussian case. Contrastive divergence [85] is proposed to employ the MCMC approximation to the estimation of the non-normalized models, which may be quite poor [88]. Discriminative training of energy-based methods [86] enable the development of an alternative to learn the parameters from the non-normalized graphical models with a wide family of loss functions, and provide a sufficient condition that the loss function must satisfy so that its minimization will make the system approach the desirable behavior. Meanwhile, score matching [87] is proposed to replace the ML problem Eq. (2.43) by minimizing the expected squared distance between the gradient of the log-density given by the model and the gradient of the log-density of the observed data.

Pseudo-likelihood [84] is used in our work (Chapter 3) since this approach can learn the MRF parameters efficiently, especially when the underlying graph of the MRF is sparse. Pseudo-likelihood [84] assumes that s_i and s_j are conditionally independent given the neighborhood of s_i, and resort to maximize $\prod_d \prod_i p(s_i^d | \mathbf{s}_{\mathbb{N}_i}^d, \boldsymbol{\theta}_{\mathcal{G}})$, where $\mathbf{s}_{\mathbb{N}_i}^d$ are the neighbors of the node s_i, i.e.

$$\boldsymbol{\theta}_{\mathcal{G}} = \max_{\boldsymbol{\theta}_{\mathcal{G}}} \prod_{d=1}^{D} \prod_{i=1}^{N} p(s_i^d | \mathbf{s}_{\mathbb{N}_i}^d, \boldsymbol{\theta}_{\mathcal{G}}). \tag{2.45}$$

According to the Boltzmann machine Eq. (2.43), $p(s_i^d | \mathbf{s}_{\mathbb{N}_i}^d, \boldsymbol{\theta}_{\mathcal{G}})$ is written as

$$p(s_i^d | \mathbf{s}_{\mathbb{N}_i}^d, \{w_{i,j}, b_i\}_{ij \in \mathcal{E}, i \in \mathcal{V}}) = \frac{1}{Z(\mathbf{s}_{\mathbb{N}_i}^d; \{w_{i,j}, b_i\})} \exp\,(s_i^d b_i + \sum_{j \in \mathbb{N}_i} s_i^d w_{i,j} s_j^d). \tag{2.46}$$

where $\{w_{i,j}\}$ and $\{b_i\}$ are the pairwise and unary parameters corresponding to the Boltzmann machine. Then, the corresponding negative logarithmic function is defined as:

$$l_{PL}(\{w_{i,j}, b_i\}_{ij \in \mathcal{E}, i \in \mathcal{V}}) = \frac{1}{D} \sum_{d}^{D} \sum_{i=1}^{N} (s_i^d b_i + \sum_{j \in \mathbb{N}_i} s_i^d w_{i,j} s_j^d) - \frac{1}{D} \sum_{d}^{D} \sum_{i=1}^{N} \log Z(\mathbf{s}_{\mathbb{N}_i}^d; \{w_{i,j}, b_i\}). \tag{2.47}$$

Gradient descent is obtained to minimize this log-likelihood function. The gradient learning equation for the unary and pairwise parameters are:

$$\begin{aligned}
b_i(t+1) &= b_i(t) + \rho \left(\frac{1}{D} \sum_d^D s_i^d + 1 - \frac{1}{D} \sum_d^D \left(\frac{2}{1 + \exp -2(b_i(t) + \sum_{j \in \mathbb{N}_i} w_{i,j}(t) s_j^d)} \right) \right); \\
w_{i,j}(t+1) &= w_{i,j}(t) + \rho \left(\frac{1}{D} \sum_d^D 2 s_i^d s_j^d + \frac{1}{D} \sum_d^D s_i^d + \frac{1}{D} \sum_d^D s_j^d - T_{i,j}^d \right);
\end{aligned} \tag{2.48}$$

where,

$$\begin{aligned}
T_{i,j} = \frac{1}{D} \sum_d^D & \left(\frac{2 s_j^n}{1 + \exp -2(b_i(t) + \sum_{j \in \mathbb{N}_i} w_{i,j}(t) s_j^d)} \right) + \\
& \frac{1}{D} \sum_d^D \left(\frac{2 s_i^n}{1 + \exp -2(b_j(t) + \sum_{i \in \mathbb{N}_j} w_{i,j}(t) s_i^d)} \right).
\end{aligned} \tag{2.49}$$

It can be seen that the computation at every gradient step only involves summation over scalar value; and thus is tractable. In our work, the MRF is adaptively estimated based on an intermediate estimate of the sparse signals. Thus, the underlying graph of MRF is often sparse. Therefore, pseudo-likelihood is a suitable choice for our work.

2.5 Collaborative representation-based classification

The discriminative nature of signal representation can be employed to perform classification. A group of researchers [89] proposed a classification method that is based on collaborating multiple training samples from all classes; thus, this type of signal recovery-based classification approach is commonly referred to as collaborative representation-based classification (CRC). The main application of CRC is on face recognition. Nevertheless, it can be suitable for other applications that *(i)* have a small number of training samples and *(ii)* cannot learn the model or perform inference over the learned model [89]–[92]. In this section, we provide background on CRC and relevant techniques that are developed based on the CRC framework.

CRC aims at predicting the class label of a query sample $\boldsymbol{y}$ in a collaborative representation setting, i.e. it is assumed that the query $\boldsymbol{y}$ can be approximately represented by the linear combination of all training samples in the sample matrix $\boldsymbol{A}$, i.e. $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{n}$ where $\boldsymbol{x}$ is an underlying representation vector and $\boldsymbol{n}$ is a small perturbation. The training samples in the sample matrix $\boldsymbol{A}$ are sorted according to class labels. Specifically, the sample matrix $\boldsymbol{A}$ is defined as.

$$\boldsymbol{A} = [\boldsymbol{A}_1, ..., \boldsymbol{A}_C], \tag{2.50}$$

where $\boldsymbol{A}_1, ..., \boldsymbol{A}_C$ are sorted according to their labels. Each $\boldsymbol{A}_c \in \mathcal{R}^{m \times n_c}$ is a sub-matrix containing n_c training samples associated with the c^{th} class, and C is the total number of classes. The goal is to recover a representation vector $\boldsymbol{x}$ given the noisy query sample. CRC recovers the shortest Euclidean distance to the query sample.

A similar line of research was first introduced in [90], which assumes that the representation vector $\boldsymbol{x}$ is sparse, termed sparse representation-based classification (SRC). Although Zhang *et al.* [89] argued that the important key to achieving good result is due to the use of collaborative representations rather than the sparsity assumption, the role of sparsity is important to increase the robustness in classification, especially when the query samples contain outliers [89], or when the number of training samples is significantly high [89], [90], [92]. Many modifications of CRC/SRC have been proposed [93]–[98]. For example, the work in [93], [94] proposed employing multiple regularization terms both SRC and CRC, as a function to control sparsity in the representation vector. However, these works are mainly developed for vision applications and often assume the good condition of training samples with high visual quality and clear visible distinctions across different classes.

In wearable based-human activity recognition, the training data are not guaranteed to be noise-free. Despite this, the CRC/SRC offers state-of-the-art performance [91], [99]–[101]. Zhang *et al.* [91], [99] employs SRC to recognize daily human activities using a wearable sensing device attached to the waist of fourteen participants. However, the location of the sensors could impact the recognition; thus, in [99], they proposed to co-recognize the sensor locations and human activities with using a Bayesian SRC. In [100], both CRC and SRC have been used to test the effectiveness of a decision-level fusion approach, where both CRC and SRC offer similar results, but SRC requires higher computational cost. Although CRCs and SRCs have shown promising classification performance in many applications, their intrinsic classification mechanism remains unclear. Recently, ProCRC [98] offers a probabilistic interpretation of CRCs and proposes to maximize the likelihood that a test sample belongs to each of the multiple classes. This significantly improves the classification performance of CRCs in vision applications.

Therefore, in the following, we provide examples of three famous, state-of-the-art algorithms, i.e. the SRC, CRC, and ProCRC methods.

2.5.1 Sparse representation-based classification (l_1-based method)

In SRC [90], the representation vector is reconstructed by solving the following l_1-minimization problem :

$$\hat{\boldsymbol{x}} = \min_{\boldsymbol{x} \in \mathcal{R}^N} \frac{1}{\sigma_n} ||\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}||_2 + ||\boldsymbol{x}||_1; \tag{2.51}$$

With the resulting representation vector, the classification is performed by choosing the class label that minimizes the residual error.

$$l^*(\boldsymbol{y}) = \min_{c \in [C]} ||\boldsymbol{y} - \boldsymbol{A}_c \hat{\boldsymbol{x}}_c||_2. \tag{2.52}$$

2.5.2 Collaborative representation-based classification (l_2-based method)

Collaborative representation-based classification (CRC) [89] aims to find the class label that minimizes the least square error:

$$\hat{\boldsymbol{x}} = \min_{\boldsymbol{x} \in \mathcal{R}^N} ||\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}||_2 + \lambda ||\boldsymbol{x}||_2; \tag{2.53}$$

where λ is the regularization parameter. The l_2-regularization has two roles: (i) it makes the least square solution stable, and (ii) it introduces certain amount of sparsity to the solution $\hat{\boldsymbol{x}}$ that is much weaker than employing l_1-norm. The solution to Eq. (2.53) can be analytically derived as

$$\hat{\boldsymbol{x}} = (\boldsymbol{A}^T \boldsymbol{A} + \lambda \boldsymbol{I})^{-1} \boldsymbol{A}^T \boldsymbol{y}, \tag{2.54}$$

Let $\boldsymbol{P} = (\boldsymbol{A}^T \boldsymbol{A} + \lambda \boldsymbol{I})^{-1} \boldsymbol{A}^T$ that is independent of $\boldsymbol{y}$. Clearly, $\boldsymbol{P}$ can be pre-calculated as a projection matrix. This makes the signal recovery very fast. Then, the classification is performed by searching for the class that has the minimum residual as follows:

$$l^*(\boldsymbol{y}) = \min_i \{r_i\}, \tag{2.55}$$

where $r_i = \frac{||\boldsymbol{y}-\boldsymbol{A}_i\boldsymbol{x}_i||_2}{||\boldsymbol{x}_i||_2}$.

CRC [89] is a very efficient method for face recognition, i.e. it can offer comparable recovery accuracy to SRC, with much faster computing time. It is shown in [89] that using only the l_2- regularization, the minimum error can be achieved. Let $e = ||\boldsymbol{y} - \boldsymbol{A}\hat{\boldsymbol{x}}||_2^2$ denotes the error corresponding to the least square regularization, and $\hat{\boldsymbol{x}}$ is the least square solution,

$$e = ||\boldsymbol{y} - \boldsymbol{A}_c\boldsymbol{x}_c||_2^2 + ||\boldsymbol{A}_c\boldsymbol{x}_c - \boldsymbol{A}\boldsymbol{x}||_2^2. \tag{2.56}$$

The error in the first term $e_1 = ||\boldsymbol{y} - \boldsymbol{A}_c\boldsymbol{x}_c||_2$ cannot be reduced since it is proportional to noise in the measurements. Meanwhile, the error in the second term $e_2 = ||\boldsymbol{A}_c\boldsymbol{x}_c - \boldsymbol{A}\boldsymbol{x}||_2$ can be further improved. The minimum error is e_1 which occurs when $e_2 = 0$, or when $\sum_{j\neq c} \boldsymbol{A}_j\boldsymbol{x}_j = 0$, which is bounded by the performance of the least square Eq. 2.53. Thus, only the l_2 can lead to the minimum error.

Next we will explore the ProCRC that elaborates the probabilistic interpretation of CRC and brings an important regularization term into the least square regularization, i.e., $||\boldsymbol{A}_c\boldsymbol{x}_c - \boldsymbol{A}\boldsymbol{x}||_2$ which is shown to significantly improve the classification performance in [98].

2.5.3 Probabilistic collaborative representation-based classification

ProCRC [98] aims to solve for the signal $\boldsymbol{x}$ that maximizes the probability of label of $\boldsymbol{y}$ equal to the c^{th} class which is the following MAP problem:

$$\max_{\boldsymbol{x}\in\mathcal{R}^N} p(l(\boldsymbol{y}) = c) \tag{2.57}$$

where the probability of the label of $\boldsymbol{y}$ is equal to a class c is defined as

$$\begin{aligned} p(l(\boldsymbol{y}) = c) =& p(l(\boldsymbol{y}) \in L_{\boldsymbol{A}})p(l(\boldsymbol{A}\boldsymbol{x}) = c|l(\boldsymbol{A}\boldsymbol{x}) \in L_{\boldsymbol{A}}) \\ &\propto \exp -(||\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}||^2 + \lambda||\boldsymbol{x}||^2 + \gamma||\boldsymbol{A}\boldsymbol{x} - \boldsymbol{A}_c\boldsymbol{x}_c)||^2, \end{aligned} \tag{2.58}$$

where γ is a regularization constant. However, the classification by the maximal $p(l(\boldsymbol{y}) = c)$ can become unstable and less discriminative [98]. ProCRC resorts to maximizing the joint probability $p(l(\boldsymbol{y}) = 1, ..., l(\boldsymbol{y}) = C)$. Applying the logarithmic operator

$$\max_{\boldsymbol{x} \in \mathcal{R}^N} \{||\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}||^2 + \lambda||\boldsymbol{x}||^2 + \tfrac{\gamma}{C} \textstyle\sum_{c=1}^{C} ||\boldsymbol{A}\boldsymbol{x} - \boldsymbol{A}_c\boldsymbol{x}_c)||^2\}. \tag{2.59}$$

This has the following closed form solution $\hat{\boldsymbol{x}} = \boldsymbol{T}\boldsymbol{y}$, where

$$\boldsymbol{T} = \left(\boldsymbol{A}^T\boldsymbol{A} + \frac{\gamma}{C} \sum_{c=1}^{C} \bar{\boldsymbol{A}}_c^T \bar{\boldsymbol{A}}_c + \lambda \boldsymbol{I} \right)^{-1} \boldsymbol{A}^T. \tag{2.60}$$

Given the solution $\hat{\boldsymbol{x}}$, the probability $p(l(\boldsymbol{y}) = c)$ is employed to perform classification:

$$p(l(\boldsymbol{y}) = c) \propto \exp -(||\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}||^2 + \lambda||\boldsymbol{x}||^2 + \frac{\gamma}{C}||\boldsymbol{A}\boldsymbol{x} - \boldsymbol{A}_c\boldsymbol{x}_c)||^2), \tag{2.61}$$

where $(||\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}||^2 + \lambda||\boldsymbol{x}||^2)$ is the same for all classes; thus, this term can be omitted in computing $p(l(\boldsymbol{y}) = c)$, i.e.

$$p_c = \exp -(||\boldsymbol{A}\boldsymbol{x} - \boldsymbol{A}_c\boldsymbol{x}_c)||^2). \tag{2.62}$$

The classification rule can be formulated as

$$l(\boldsymbol{y}) = \max_c \{p_c\}. \tag{2.63}$$

It can be seen that the two main improvements in ProCRC over CRC are (i) the ProCRC contains the additional term $(\boldsymbol{A}\boldsymbol{x} - \boldsymbol{A}_c\boldsymbol{x}_c)$ in Eq. (2.59) to jointly maximize the likelihood that a test sample belongs to each of the multiple classes, and (ii) The classifier uses Eq. (2.62) which is the same as the additional term $(\boldsymbol{A}\boldsymbol{x} - \boldsymbol{A}_c\boldsymbol{x}_c)$; thus, it can directly search for the class label from the optimal representation vector. It has been shown that these two differences in ProCRC lead to better performance than CRC in multiple recognition tasks [98].

Despite a number of variations of CRCs and SRCs, e.g. [89], [90], [92]–[95], [98], these methods often rely on finding the representation vectors that minimize

$||\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}||_2$ or $||\boldsymbol{A}\boldsymbol{x} - \boldsymbol{A}_c\boldsymbol{x}_c||_2$. However, if two sets of training samples $\boldsymbol{A}_i$ and $\boldsymbol{A}_j$ are similar or correlated, the predicted label can be misclassified as the other class. Nevertheless, correlation between training samples is a common problem in wearable-sensor-based human activity recognition. This is because each activity is a combination of body motions. To address this problem, we are motivated to extract the underlying information in the query sample as a prior in signal recovery. The underlying structure can offer information that is related to the class of the query sample. Here, we provide more details on the correlated training samples problem and how we approach and address the problem in Chapter 5.

2.6 Summary

This chapter presents a review of many algorithms for structured sparse signal recovery in compressive sensing. Two classes of structured sparsity models have been studied, namely, the *deterministic* structured sparsity models, and the *probabilistic* structured sparsity models. The deterministic model represents the geometrical structure of the signals in a union of sparse subspaces (see Section 2.2.1); however, the assumption about signal geometrical structure is limited to some groups of sparse signals. To address this problem, the probabilistic structured sparsity models represent the signal structure through the probability distribution of signals (see Section 2.2.2).

The theoretical guarantee of the sample complexity for the deterministic model can be directly derived based on the restricted isometry properties. The examples of how to reduce the sample complexity with deterministic models have been presented in Section 2.3.1. The sample complexity can achieve the theoretical optimum $\mathcal{O}(k)$ with the assumed block and tree structure. Meanwhile, the sample complexity of the probabilistic structured sparsity model is analyzed based on the probabilistic restricted isometric properties. The examples of how to reduce the sample complexity with this class of structured sparsity model are also explored (see Section 2.3.4). The existing results demonstrate the potential to reduce the sample complexity to a theoretical optimum $\mathcal{O}(k)$ without imposing any geometrical structure assumptions. However, this is under the condition that the candidate supports encoded from a trained probabilistic model can well represent the true signal. This condition could be violated, if the training signals are not representative enough. To address this problem, Chapter 3 and Chapter 4 will present a solution to employ our proposed adaptive MRF that has sufficiently high flexibility to capture new signal structures.

This chapter also provides a background review on the inference and learning in Markov random fields (Section 2.4) that have been used in our proposed method. Also, we review the sparse representation-based classification and briefly discuss the problems with the existing approaches that can be addressed by our adaptive MRF approach in Section 2.5. The application of our adaptive MRF to classification will be presented in Chapter 5.

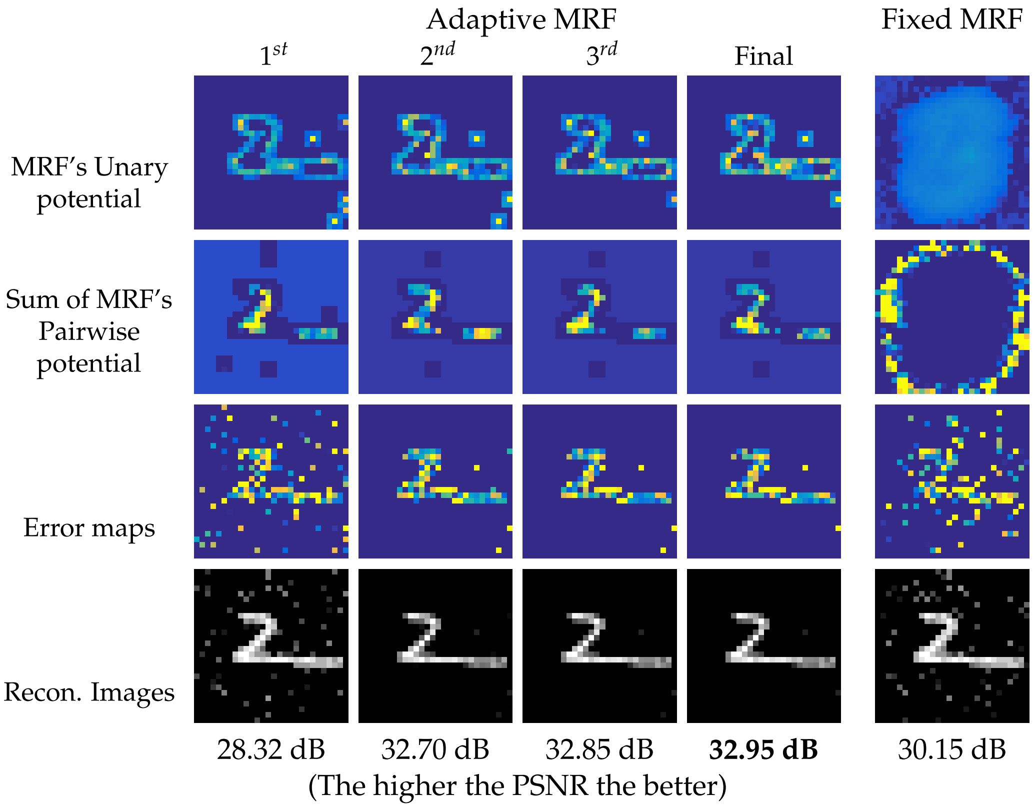
Chapter 3

Adaptive Markov Random Fields for Structured CS

3.1 Introduction

The exploitation of intrinsic structures of sparse signals underpins the recent progress in compressive sensing (CS). The key to exploiting signal structures is to employ structured sparsity models that have the two desirable properties: *flexibility*—the ability to fit a wide range of signals with diverse structures, and *adaptability* —being adaptive to actual signal structures. In the previous chapter, we have reviewed the two main classes of structure sparsity models: the *deterministic* and the *probabilistic structured sparsity models*. Deterministic structured sparsity models often assume prior knowledge about the geometrical structure of the sparse signals such as group or tree structure. Thus, these models lack the flexibility to capture different types of signal structures. The probabilistic structure sparsity models, e.g. Markov random fields (MRFs), have the flexibility to model a wide variety of signal structures, but the MRF parameters are obtained from training and cannot adapt for new signal structures. Thus, their performance is constrained by the information in the training data. Meanwhile, the clustered sparsity models can adapt the model parameters for a new sparse signal, but they assume there are limited signal structures such as a cluster structure. As a result, the clustered sparsity models are not as flexible as the MRFs. Therefore, these existing structure sparsity models can only achieve one of the two desirable properties; either flexibility or adaptability.

To achieve the two desirable properties, we propose to leverage the adaptability of



(A) Intermediate and final results.

(B) Final result.

FIGURE 3.1: Comparison of the effect of an Adaptive MRF prior and a Fixed MRF prior on a sample MNIST data: The first (top) and second rows include the unary potential at each image pixel and the sum of the pairwise potentials of the adjacent pixels. The third and fourth rows include reconstructed images and error maps. Our adaptive MRF is much more attuned to the image structure (digit 2). However, the fixed MRF only focuses on the region (the disk shape) where all the digits appear.

an MRF. MRFs represent the structure of signals by defining a probability distribution over an undirected graph. A Boltzmann machine (BM) is used as the probability distribution of the MRF because of its flexibility to model different signal distributions. To realize the adaptability, we enable the parameter estimation for the MRF where both the *BM parameters* and the *underlying graph* of the MRF are estimated, based on an intermediate estimate of the latent sparse signal. Thus, the estimated MRF parameters are adapted to represent the underlying structure of the latent signal.

Figure 3.1A demonstrates the improved performance by employing the adaptive MRF versus the trained MRF as a prior in recovering a sample MNIST image (no. 2). The evolution of the intermediate estimates of the adaptive MRF and the reconstructed images are provided in the 1^{st}-3^{rd} columns, and their final results are provided in the 4^{th} column. Meanwhile, the trained MRF is fixed throughout the signal recovery process. The trained MRF and the reconstructed images are provided in the last column, denoted *fixed MRF*. The first (top) and the second rows show the resulting unary potential and the pairwise potentials. The third and fourth rows show the error maps with respect to the ground truth image no. 2 and the intermediate estimates of images. It is clear that the adaptive MRF improves the quality of the estimated image in each iteration, as the MRF parameters are refined: the unary and pairwise potentials of the adaptive MRF are adjusted to fit the digit number 2, as opposed to recovering the sparse signal with a fixed MRF which cannot adapt throughout the signal recovery process. The fixed MRF captures the universal pattern of all the training, which appears as a disk shape, where all the digits appear. As a result, the adaptive MRF provides higher reconstruction quality, both numerically and visually, than the fixed MRF.

To exploit an MRF as a prior in signal recovery, most existing MRF methods such as [30]–[32], [34] are based on the non-recursive two-step approach (see Algorithm 2.3), that is, it estimates the support first, and then, estimates the sparse signal. However, this can cause high computational time, and any error in the first step can propagate to the second step and can not be minimized later. Moreover, these methods employ homogeneous noise and signal parameters from the training data, which do not necessarily represent the actual parameters of the testing signals well.

To address these problems, we propose to estimate the sparse signal, support, noise parameters, and sparse signal parameters jointly and iteratively, based on the adapted MRF. However, by doing this, the whole signal estimation becomes a non-convex optimization problem over discrete and continuous variables (see Eq.(3.8))—support, sparse signals, noise and signal parameters —which is very difficult to solve in general. To tackle this non-convex problem, we propose to apply a latent Bayes model [102], [103] to provide a new formulation (see Eq.(3.10)), which can be reduced into several subproblems by using alternative minimization optimization scheme. With the structured sparsity prior being considered, we derive several new formulations to solve for the sparse signal, support, and signal covariance with maximum a posteriori (MAP) estimation. To estimate the support efficiently, we propose to approximate the non-linear, pairwise potentials in the resulting MAP problem into linear, unary potentials. This brings in the closed-form solutions for estimating sparse signal, noise and signal parameters. Meanwhile, the support estimation problem can be solved efficiently with any off-the-shelf MAP inference tools.

Therefore, we propose to leverage the adaptability of the MRF and develop a new sparse signal estimation to obtain the sparse signals with the adapted MRF. We highlight the contribution of this chapter as follows:

1. *Two-step-Adaptive MRF* framework to adaptively estimate an MRF to fit any signal structure. To realize adaptability, both the BM parameters and the underlying graph are updated based on an estimated sparse signal. Then, our sparse signal estimation exploits the adapted MRF as a prior to improve the estimation accuracy. The Two-step-Adaptive MRF framework is discussed in Section 3.3. The superior performance of the proposed adaptive MRF is provided in Section 3.6.5.

2. *New sparse signal estimation algorithm* to jointly and iteratively estimate the support and the sparse signal, noise and signal parameters. We achieve this by employing a latent Bayes model [102], [103] to provide a new formulation (see Eq.(3.10)) that can be solved efficiently with alternative minimization optimization. With the structure of sparse signals being considered, we derive several new formulations to solve for the sparse signal, support, and signal covariance with maximum a posteriori (MAP) estimation. We compare the proposed sparse signal estimation that solves the new formulation Eq.(3.10) against the existing schemes [31], [32] that solve Eq.(3.8) in the Two-step-Adaptive MRF framework. Our approach offers better reconstruction accuracy and runtime (see Section 3.6.6).

3. *Theoretical result* to demonstrate the essence of adaptive support prior and the connection between an adaptive support prior and probabilistic RIP in Section 3.4. It shows that if the adapted support prior converges to the distribution of the test signal, it guarantees that the feasible set contain the test signal. Then, the sample complexity of $\mathcal{O}(k)$ can be achieved.

4. We evaluate the performance of the proposed algorithm with three benchmark datasets: i) MNIST, ii) CMU-IDB, and iii) CIFAR-10. To observe the performance in exploiting different signal structures, we study the reconstruction of sparse signals in the spatial domain and standard bases—wavelet, discrete cosine transform (DCT), and principal component analysis (PCA) bases. The results demonstrate promising performance in terms of accuracy in recovering the sparse signal, with a moderate runtime (see Section 3.6.7).

The following sections are organized as follows: Section 3.2 presents the signal model for graphical compressive sensing. Section 3.3 addresses the proposed Two-step-Adaptive MRF. The new sparse signal estimation and the corresponding optimization process are provided in Section 3.3.3. Then, the computational complexity is discussed in Section 3.5. To this end, extensive experiments and analysis on three benchmark datasets are provided in Section 3.6.

3.2 Graphical Compressive Sensing

In this study, we capture the structure of sparse signal $\boldsymbol{x}$ by modeling its support explicitly. Let $\boldsymbol{s} \in \{-1, 1\}^N$ indicate the support of $\boldsymbol{x}$ such that $s_i = 1$ when $x_i \neq 0$ and $s_i = -1$ when $x_i = 0$. Let $\boldsymbol{x}_s \in \mathbb{R}^k$ denote the non-zero coefficients of the k-sparse $\boldsymbol{x}$. Our goal is to estimate $\boldsymbol{s}$ and $\boldsymbol{x}_s$ from the linear measurements $\boldsymbol{y}$ corrupted by additive noise $\boldsymbol{n}$ as follows,

$$\boldsymbol{y} = \boldsymbol{A}_s \boldsymbol{x}_s + \boldsymbol{n}. \tag{3.1}$$

Here $\boldsymbol{A}_s$ is the matrix with k columns selected from the matrix $\boldsymbol{A}$ according to non-zero coefficients specified by $\boldsymbol{s}$, and $\boldsymbol{n}$ is the Gaussian white noise, i.e., $\boldsymbol{n} \sim \mathcal{N}(0, \sigma_n \boldsymbol{I})$ where σ_n is the noise variance and $\boldsymbol{I}$ denotes an identity matrix with a proper size. The corresponding likelihood over $\boldsymbol{y}$ can thus be formulated as

$$p(\boldsymbol{y}|\boldsymbol{x}_s; \sigma_n) = \mathcal{N}(\boldsymbol{A}_s \boldsymbol{x}_s, \sigma_n \boldsymbol{I}). \tag{3.2}$$

Each observed measurement y_i can be seen a noisy linear combination of non-zero sparse signal coefficients that are projected on the matrix atoms. The interdependencies among coefficients can be modelled through the prior of support $\boldsymbol{s}$.

Specifically, we impose a graphical sparsity prior on $\boldsymbol{x}_s$ and $\boldsymbol{s}$ (Section 3.2.1). Subsequently, we show how to recovery the sparse signal $\boldsymbol{x}$ from the measurements $\boldsymbol{y}$ by our new adaptive MRF inference (Section 3.3).

3.2.1 Graphical sparsity prior

The sparse signals often exhibit an arbitrary and complex statistical dependency between the sparse signal coefficients [32], [34]. The MRFs are flexible and expressive enough to capture complex dependency by defining the probabilistic distribution over an undirected graph [72], [104]. Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ represents the undirected graph where $\mathcal{V}$ is a set of nodes (each representing a variable) and $\mathcal{E}$ is a set of undirected edges. Let $\boldsymbol{\Theta}_{\mathcal{G}} = \left\{ \mathcal{W}_i(\cdot; \xi^i), \mathcal{W}_{i,j}(\cdot; \xi^{(i,j)} \right\}_{i \in \mathcal{V}, (i,j) \in \mathcal{E}}$ denote the set of potential parameters associated with the probability distribution defined on an MRF to represent local dependency among the nodes in the undirected graph $\mathcal{G}$. Therefore, by configuring the edges in the edge set $\mathcal{E}$ as well as the corresponding potentials $\boldsymbol{\Theta}_{\mathcal{G}}$, the MRF is capable of representing a wide range of signals with diverse structures [105], which include most of the geometrical structures, e.g. block and tree structures. To model the structure of sparse signals, we impose a graphical sparsity prior on $\boldsymbol{x}_s$ and $\boldsymbol{s}$ as follows.

First, we define the prior of support $\boldsymbol{s}$ based on MRFs. Each coefficient s_i of the support $\boldsymbol{s}$ is mapped onto each node $i \in \mathcal{V}$. Given the graph $\mathcal{G}$, the probability of the support $p(\boldsymbol{s}; \boldsymbol{\Theta}_{\mathcal{G}})$ can be represented as follows with a normalization constant Z,

$$\frac{1}{Z} \exp \left(\sum_{i \in \mathcal{V}} \mathcal{W}_i(s_i; \xi^i) + \sum_{(i,j) \in \mathcal{E}} \mathcal{W}_{i,j}(s_i, s_j; \xi^{(i,j)}) \right), \tag{3.3}$$

where $\mathcal{W}_{(\cdot)}(\cdot; \xi^{(\cdot)})$ is commonly assumed to be a linear function with respect to $\xi^{(\cdot)}$, e.g. $\mathcal{W}_i(s_i; \xi^i) = \xi^i s_i$ and $\mathcal{W}_{i,j}(s_i, s_j; \xi^{(i,j)}) = \xi^{(i,j)} s_i s_j$. With the linear potentials, the probability distribution Eq.(3.3) is often called a *Boltzmann machine (BM)*. Hence, the parameter set contains the *BM parameters* $\boldsymbol{\Theta}_{\mathcal{G}} = \left\{ \xi^{(i)}, \xi^{(i,j)} \right\}_{i \in \mathcal{V}, (i,j) \in \mathcal{E}}$. The first group of BM parameters defines the bias (e.g., confidence) potential to each s_i; while the second group characterizes the pairwise interaction between two variable nodes, e.g. $\xi^{(i,j)}$ weights dependency between s_i, s_j.

In addition, we assume $\boldsymbol{x}$ comes from a Gaussian distribution $\mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_x)$ where $\boldsymbol{\Sigma}_x$ denote sparse signal covariance which is a diagonal matrix. Given $\boldsymbol{s}$, the probability of non-zero coefficients is defined as

$$p(\boldsymbol{x}_s|\boldsymbol{s}) = \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{x,s}) \tag{3.4}$$

where $\boldsymbol{\Sigma}_{x,s}$ denotes the covariance of non-zero coefficients according to $\boldsymbol{s}$. Then, $p(\boldsymbol{x}_s|\boldsymbol{s})p(\boldsymbol{s}; \boldsymbol{\Theta}_{\mathcal{G}})$ forms the graphical sparsity prior in this study. To reduce the computation in estimating the sparse signal variance, $\boldsymbol{\Sigma}_{x,s}$ is assumed to be a diagonal matrix whose diagonal entry is chosen from $\boldsymbol{\Sigma}_x$.

3.3 Two-step-Adaptive MRF

Provided that the optimum parameters $\hat{\sigma}_n$, $\hat{\boldsymbol{\Sigma}}_{x,s}$, $\hat{\boldsymbol{\Theta}}$, and $\hat{\mathcal{G}}$ are given beforehand, the latent $\boldsymbol{x}_s$ and $\boldsymbol{s}$ can be estimated by solving a maximum a posteriori (MAP) problem:

$$\max_{\boldsymbol{x}_s, \boldsymbol{s}} p(\boldsymbol{x}_s, \boldsymbol{s}|\boldsymbol{y}) \propto p(\boldsymbol{y}|\boldsymbol{x}_s; \hat{\sigma}_n) p(\boldsymbol{x}_s|\boldsymbol{s}; \hat{\boldsymbol{\Sigma}}_{x,s}) p(\boldsymbol{s}; \hat{\boldsymbol{\Theta}}_{\mathcal{G}}). \tag{3.5}$$

However, these parameters are often unknown in real applications. Some previous work obtains both of the BM parameters and the underlying graph of the MRF from the training data [31]–[33]. While these parameters can well represent the common characteristics among training data, they fail to adapt to testing data as shown in the preliminary results in Figure 3.1B. On the contrary, with adaptive MRF, the quality of the reconstructed image is obviously improved because the adapted MRF keeps reducing the reconstruction error in each iteration.

Motivated by this result, we propose a Two-step-Adaptive MRF to adaptively estimate all the parameters—the BM parameters $\boldsymbol{\Theta}_{\mathcal{G}}$ and the underlying graph $\mathcal{G}$ of the MRF, and noise and signal parameters σ_n and $\boldsymbol{\Sigma}_{x,s}$— according to the measurements. Our objective is, therefore, to estimate these unknowns—$\boldsymbol{s}, \sigma_n$, $\boldsymbol{\Sigma}_{x,s}$, and $\boldsymbol{\Theta}_{\mathcal{G}}$— from the measurements by solving

$$\begin{aligned}\max_{\boldsymbol{s},\sigma_n,\boldsymbol{\Sigma}_{x,s},\boldsymbol{\Theta}_{\mathcal{G}}} \; & p(\boldsymbol{s},\sigma_n,\boldsymbol{\Sigma}_{x,s},\boldsymbol{\Theta}_{\mathcal{G}}|\boldsymbol{y}) \;\propto \\ & \int p(\boldsymbol{y}|\boldsymbol{x}_s,\sigma_n)p(\boldsymbol{x}_s|\boldsymbol{s},\boldsymbol{\Sigma}_{x,s})p(\boldsymbol{s}|\boldsymbol{\Theta}_{\hat{\mathcal{G}}})\mathrm{d}\boldsymbol{x}_s,\end{aligned} \tag{3.6}$$

which intrinsically maximizes the likelihood of measurements over all the model parameters, as well as the support. Solving Eq. (3.6) directly is intractable. To circumvent this problem, we reduce Eq. (3.6) into two subproblems as follows.

3.3.1 Sparse signal estimation

Given the MRF parameters $\hat{\boldsymbol{\Theta}}_{\mathcal{G}}$ and $\hat{\mathcal{G}}$, we first infer other parameters from the measurements by solving

$$\begin{aligned}\max_{\boldsymbol{s},\sigma_n,\boldsymbol{\Sigma}_{x,s}} \; & p(\boldsymbol{s},\sigma_n,\boldsymbol{\Sigma}_{x,s}|\boldsymbol{y},\hat{\boldsymbol{\Theta}}_{\hat{\mathcal{G}}}) \;\propto \\ & \int p(\boldsymbol{y}|\boldsymbol{x}_s,\sigma_n)p(\boldsymbol{x}_s|\boldsymbol{s},\boldsymbol{\Sigma}_{x,s})p(\boldsymbol{s}|\hat{\boldsymbol{\Theta}}_{\hat{\mathcal{G}}})\mathrm{d}\boldsymbol{x}_s.\end{aligned} \tag{3.7}$$

The optimization problem in Eq. (3.7) can be equally reformulated as [32], [106] :

$$\begin{aligned}&\min_{\boldsymbol{s},\sigma_n,\boldsymbol{\Sigma}_{x,s}} -\log \int p(\boldsymbol{y}|\boldsymbol{x}_s,\sigma_n)p(\boldsymbol{x}_s|\boldsymbol{s},\boldsymbol{\Sigma}_{x,s})p(\boldsymbol{s},\hat{\boldsymbol{\Theta}}_{\hat{\mathcal{G}}})\mathrm{d}\boldsymbol{x}_s \equiv \\ &\frac{1}{2}\boldsymbol{y}^T(\sigma_n + \boldsymbol{A}_s\boldsymbol{\Sigma}_{x,s}\boldsymbol{A}_s^T)^{-1}\boldsymbol{y} + \frac{1}{2}\log|\sigma_n\boldsymbol{I} + \boldsymbol{A}_s\boldsymbol{\Sigma}_{x,s}\boldsymbol{A}_s^T| \\ &-\log p(\boldsymbol{s}|\hat{\boldsymbol{\Theta}}_{\hat{\mathcal{G}}}).\end{aligned} \tag{3.8}$$

The existing work [30]–[32] employed the two step-non-recursive approach [30], [107]: first, they attempt to solve Eq. (3.8) for the support. Given the resulting support, they still have to estimate $\boldsymbol{x}$ from Eq. (3.5). However, this can cause error accumulation problem since the error in the first step cannot be minimized in the second step. Moreover, the support estimation problem in Eq. (3.8) is non-convex over discrete and continuous variables—support, noise and signal parameters—which is difficult to solve in general. Even after fixing $\boldsymbol{s}$, the remaining problem of Eq. (3.8) is still non-convex, and there are no closed-form solutions for σ_n and $\boldsymbol{\Sigma}_{x,s}$. Therefore, these works [30]–[32] resorts to employ the noise and signal parameters from training data.

To tackle this non-convex problem, we propose to use a strict upper bound of Eq. (3.8) based on a latent Bayes model [102], [103]:

$$\begin{aligned} &\boldsymbol{y}^T(\sigma_n + \boldsymbol{A}_s\boldsymbol{\Sigma}_{x,s}\boldsymbol{A}_s^T)^{-1}\boldsymbol{y} \\ &= \inf_{\boldsymbol{x}_s} \frac{1}{\sigma_n}(\boldsymbol{y} - \boldsymbol{A}_s\boldsymbol{x}_s)^T(\boldsymbol{y} - \boldsymbol{A}_s\boldsymbol{x}_s) + \boldsymbol{x}_s^T\boldsymbol{\Sigma}_{x,s}^{-1}\boldsymbol{x}_s. \end{aligned} \tag{3.9}$$

With this bound, the cost function Eq. (3.8) can be transformed into a new cost function as

$$\begin{aligned} L(\boldsymbol{x}_s, \boldsymbol{s}, \sigma_n, \boldsymbol{\Sigma}_{x,s}) =& \frac{1}{2\sigma_n}(\boldsymbol{y} - \boldsymbol{A}_s\boldsymbol{x}_s)^T(\boldsymbol{y} - \boldsymbol{A}_s\boldsymbol{x}_s) + \frac{1}{2}\boldsymbol{x}_s^T\boldsymbol{\Sigma}_{x,s}^{-1}\boldsymbol{x}_s + \frac{1}{2}\log|\sigma_n\boldsymbol{I} + \boldsymbol{A}_s\boldsymbol{\Sigma}_{x,s}\boldsymbol{A}_s^T| \\ & - \log p(\boldsymbol{s}; \hat{\boldsymbol{\Theta}}_{\hat{\mathcal{G}}}). \end{aligned} \tag{3.10}$$

It can be proved that the resulting support, noise variance, and non-zero coefficients' covariance—$\boldsymbol{s}$, σ_n, $\boldsymbol{\Sigma}_{x,s}$—from Eq. (3.10) are equivalent to that from solving Eq. (3.8) [102], [103]. This enables the closed-form solutions for $\boldsymbol{s}$, σ_n and $\boldsymbol{\Sigma}_{x,s}$. Moreover, $\boldsymbol{x}$, $\boldsymbol{s}$, σ_n, and $\boldsymbol{\Sigma}_{x,s}$ are jointly estimated in a single framework.

Note that the structured sparsity prior is not considered in the previous latent Bayes model [102], [103]. Since optimizing Eq. (3.10) involves several unknown variables, we apply an alternative minimization scheme to reduce the optimization problem into several subproblems. With structured sparsity prior being considered, we derive several new formulations for the sub-optimization problems to efficiently estimate sparse signal, support, and signal covariance. The details on the optimization are provided in Section 3.3.3.1.

3.3.2 MRF parameter estimation

Given the estimates of sparse signal, support, noise variance, and non-zero coefficients' covariance, i.e. $\hat{\boldsymbol{x}}_s, \hat{\boldsymbol{s}}, \hat{\sigma}_n, \hat{\boldsymbol{\Sigma}}_{x,s}$, we estimate the MRFs parameters given the

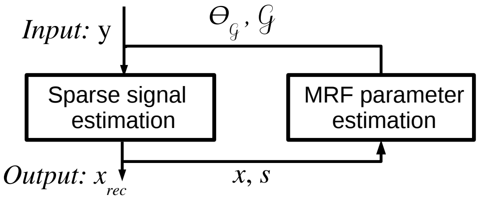


FIGURE 3.2: Visualization of the two-step framework

measurements by maximizing the likelihood function:

$$\begin{aligned} p(\boldsymbol{\Theta}_\mathcal{G}|\boldsymbol{y}) &\propto \int p(\boldsymbol{y}|\boldsymbol{x}_s,\sigma_n)p(\boldsymbol{x}_s|\boldsymbol{s},\boldsymbol{\Sigma}_{x,s})p_\mathcal{G}(\boldsymbol{s}|\boldsymbol{\Theta}_\mathcal{G})\mathrm{d}\boldsymbol{x}_s \\ &= p(\boldsymbol{y}|\hat{\boldsymbol{s}},\hat{\sigma}_n,\hat{\boldsymbol{\Sigma}}_{x,s})p(\hat{\boldsymbol{s}}|\boldsymbol{\Theta}_\mathcal{G}) \propto p(\hat{\boldsymbol{s}}|\boldsymbol{\Theta}_\mathcal{G}), \end{aligned} \tag{3.11}$$

where the likelihood is approximated by the point-wise maximum. If $\hat{\mathcal{G}}$ is given, the BM parameters are obtained from solving the following maximum likelihood (ML) problem Eq. (3.12):

$$\hat{\boldsymbol{\Theta}}_{\hat{\mathcal{G}}} = \max_{\boldsymbol{\Theta}_{\hat{\mathcal{G}}}} p(\hat{\boldsymbol{s}}|\boldsymbol{\Theta}_{\hat{\mathcal{G}}}) \tag{3.12}$$

which encourages $\boldsymbol{\Theta}_{\hat{\mathcal{G}}}$ (i.e. the graphical sparsity prior) to be adaptive to the distribution of the latent support signal. This ML can be solved by many graphical model learning approaches such as the maximum pseudo-likelihood [84], contrastive divergence [85], and discriminative training of energy-based methods [86]. The graph $\mathcal{G}$ can be estimated from structured learning approaches such as score-based learning [72]. However, performing the structure learning every iteration could result in extremely high computation. Thus, we update the graph with a graph update procedure. The details on solving ML for the parameters and graph update procedure are provided in Section 3.3.3.2.A.

The estimation problems Eq. (3.10) in Section 3.3.1 and Eq. (3.12) in Section 3.3.2 are then alternatively optimized until convergence. Figure 3.2 illustrates the proposed Two-step-Adaptive MRF. The estimated sparse signal is able to refine the MRF parameters, while the refined MRF parameters result in more accurate sparse signal recovery. After these two processes iterate until they converge, we obtain the final result.

3.3.3 Optimization

In this section, we will first focus on solving the sparse signal estimation problem in Eq. (3.10), given the MRF. Then, we will focus on the MRF parameter estimation based on the estimated sparse signal (3.12).

3.3.3.1 Sparse signal estimation

Here, we mainly focus on optimizing Eq. (3.10) to obtain all involved unknown variables, given the parameters $\hat{\mathbf{\Theta}}_{\hat{\mathcal{G}}}$ and the underlying graph $\hat{\mathcal{G}}$, as follows:

$$\{\hat{\mathbf{x}}, \hat{\mathbf{s}}, \hat{\sigma}_n, \hat{\mathbf{\Sigma}}_{x,s}\} = \min_{\mathbf{x}_s, \mathbf{s}, \sigma_n, \mathbf{\Sigma}_{x,s}} L(\mathbf{x}, \mathbf{s}, \sigma_n, \mathbf{\Sigma}_{x,s}). \tag{3.13}$$

Since the optimization problem Eq. (3.13) involves several unknown variables, we apply an alternative minimization scheme to reduce the problem Eq. (3.13) into several subproblems, each of which involves only one variable and often can be solved directly. With the structured sparsity prior being considered, we present several formulations for the estimation of sparse signal, noise and signal parameters which gain the closed-form solutions. To estimate the support efficiently, we propose to further approximate non-linear, pairwise potentials in the resulting subproblem into linear, unary potentials. These subproblems are then optimized until convergence using alternating optimization scheme.

3.3.3.1.A Optimization over support $\mathbf{s}$ Given the estimates of sparse signal, support, and non-zero coefficients' covariance—$\mathbf{x}$, σ_n, and $\mathbf{\Sigma}_{x,s}$, the subproblem over the support $\mathbf{s}$ can be given as

$$\begin{aligned} \min_{\mathbf{s} \in \{-1,1\}^N} \quad & \frac{1}{2\sigma_n} \mathbf{x}_s^T \mathbf{A}_s^T \mathbf{A}_s \mathbf{x}_s - \frac{1}{\sigma_n} \mathbf{y}^T \mathbf{A}_s \mathbf{x}_s + \frac{1}{2} \mathbf{x}_s^T \mathbf{\Sigma}_{x,s}^{-1} \mathbf{x}_s \\ & + \frac{1}{2} \log |\sigma_n \mathbf{I} + \mathbf{A}_s \mathbf{\Sigma}_{x,s} \mathbf{A}_s^T| - \log p(\mathbf{s}; \hat{\mathbf{\Theta}}_{\hat{\mathcal{G}}}). \end{aligned} \tag{3.14}$$

The minimization problem in Eq. (3.14) can be viewed as an MAP problem over a graphical model. Solving Eq. (3.14) is computationally extensive because the logarithmic and the pairwise terms require an exhaustive search over all possible support patterns. In particular, when the coefficients of the estimated sparse signals $\boldsymbol{x}$ are all non-zero, so the first term $\boldsymbol{x}_s^T \boldsymbol{A}_s^T \boldsymbol{A}_s \boldsymbol{x}_s$ becomes a fully connected graph. To address these problems, we derive a new support estimation formulation Eq.(3.20) where the logarithmic and quadratic terms are approximated into linear functions (unary potentials) with respect to the support.

To approximate the logarithmic term, we use the upper bound of the determinant of a positive definite matrix, which is the determinant of the diagonal entries of $(\sigma_n \boldsymbol{I} + \boldsymbol{A}_s \boldsymbol{\Sigma}_{x,s} \boldsymbol{A}_s^T)$, i.e.

$$\log |\sigma_n \boldsymbol{I} + \boldsymbol{A}_s \boldsymbol{\Sigma}_{x,s} \boldsymbol{A}_s^T| \leq \sum_{i \in \mathcal{V}} \log[\boldsymbol{\Sigma}_x]_{i,i} + \log[(\sigma_n \boldsymbol{\Sigma}_x^{-1} + \boldsymbol{A}^T \boldsymbol{A})]_{i,i}, \tag{3.15}$$

where $\mathcal{V}$ is the index set of non-zero sparse coefficients. The notation $[\boldsymbol{M}]_{i,i}$ refers to the $i-$th diagonal entry of the matrix $\boldsymbol{M}$. Then, we employ the Hadarmard product to explicitly represent the support. Let $\boldsymbol{v} \in \{0, 1\}^N$ be a binary variable vector that is the result from mapping each coefficient of $\boldsymbol{s}$ to binary values 0 and 1, i.e., if $s_i > 0$, then $v_i = 1$; otherwise, $v_i = 0$. We exploit Hadamard product properties to extract $\boldsymbol{v}$ by transforming the following terms:

$$\begin{aligned}
\boldsymbol{x}_s^T \boldsymbol{A}_s^T \boldsymbol{A}_s \boldsymbol{x}_s &= (\boldsymbol{x} \odot \boldsymbol{v})^T \boldsymbol{A}^T \boldsymbol{A} (\boldsymbol{x} \odot \boldsymbol{v}) &&= \boldsymbol{v}^T \boldsymbol{X}^T \boldsymbol{A}^T \boldsymbol{A} \boldsymbol{X} \boldsymbol{v} \\
\boldsymbol{x}_s^T \boldsymbol{\Sigma}_{x,s}^{-1} \boldsymbol{x}_s &= (\boldsymbol{x} \odot \boldsymbol{v})^T \boldsymbol{\Sigma}_x^{-1} (\boldsymbol{x} \odot \boldsymbol{v}) &&= \boldsymbol{v}^T \boldsymbol{X}^T \boldsymbol{\Sigma}_x^{-1} \boldsymbol{X} \boldsymbol{v}. \\
\frac{1}{\sigma_n} \boldsymbol{y}^T \boldsymbol{A}_s \boldsymbol{x}_s &= \frac{1}{\sigma_n} \boldsymbol{y}^T \boldsymbol{A} (\boldsymbol{x} \odot \boldsymbol{v}) &&= \frac{1}{\sigma_n} \boldsymbol{y}^T \boldsymbol{A} \boldsymbol{X} \boldsymbol{v}.
\end{aligned} \tag{3.16}$$

Then, the optimization problem in Eq. (3.14) can be equivalently formulated as

$$\begin{aligned}
&\min_{\boldsymbol{v} \in \{0,1\}^N} \frac{1}{2\sigma_n} \boldsymbol{v}^T (\boldsymbol{X}^T \boldsymbol{A}^T \boldsymbol{A} \boldsymbol{X} + \sigma_n \boldsymbol{X}^T \boldsymbol{\Sigma}_x^{-1} \boldsymbol{X}) \boldsymbol{v} \\
&+ (-\frac{1}{\sigma_n} \boldsymbol{y}^T \boldsymbol{A} \boldsymbol{X} + \boldsymbol{p}^T + \boldsymbol{q}^T) \boldsymbol{v} - \log p(2\boldsymbol{v} - 1; \hat{\boldsymbol{\Theta}}_{\hat{\mathcal{G}}}),
\end{aligned} \tag{3.17}$$

where $\boldsymbol{v} = \frac{1}{2}(\boldsymbol{s} + 1)$, $\boldsymbol{p} = \frac{1}{2} \log(\mathbf{diag}\{\boldsymbol{\Sigma}_x\})$; $\boldsymbol{q} = \frac{1}{2} \log\left(\mathbf{diag}\{\sigma_n \boldsymbol{\Sigma}_x^{-1} + \boldsymbol{Q}\}\right)$; $\boldsymbol{Q}$ is a diagonal matrix whose diagonal entries are the diagonal entries of $\boldsymbol{A}^T \boldsymbol{A}$; and $\boldsymbol{X}$ is a diagonal matrix with diagonal coefficients from $\boldsymbol{x}$. The cost function of Eq. (3.17) is the upper bound of Eq. (3.14).

To avoid causing a fully connected graph when the coefficients in $\boldsymbol{x}$ are all non-zero, we exploit the fact that the measurement matrix $\boldsymbol{A}$ satisfies the restricted isometric property and is thus nearly orthogonal [108]:

$$||\boldsymbol{A}_s^* \boldsymbol{A}_s - \boldsymbol{I}||_{2\to 2} \leq \delta_s, \tag{3.18}$$

where $\boldsymbol{I}$ is an identity matrix with an appropriate size, $||\cdot||_{2\to 2}$ is the operator norm, δ_s is a small value corresponding restricted isometric constant, and $\boldsymbol{A}^*$ is the Hermitian transpose of $\boldsymbol{A}$. Hence, the first term in Eq. (3.17) can be approximated as follows:

$$\boldsymbol{v}^T(\boldsymbol{X}^T\boldsymbol{A}^T\boldsymbol{A}\boldsymbol{X} + \sigma_n \boldsymbol{X}^T \boldsymbol{\Sigma}_x^{-1}\boldsymbol{X})\boldsymbol{v} = \boldsymbol{v}^T\boldsymbol{X}^T(\boldsymbol{I} + \sigma_n \boldsymbol{\Sigma}_x^{-1})\boldsymbol{X}\boldsymbol{v}, \tag{3.19}$$

Thus, the signal support $\boldsymbol{s}$ is estimated by solving the following optimization problem:

$$\begin{aligned}\min_{\boldsymbol{v}\in\{0,1\}^N} & \left(\frac{1}{2\sigma_n}\boldsymbol{r}^T - \frac{1}{\sigma_n}\boldsymbol{y}^T\boldsymbol{A}\boldsymbol{X} + \boldsymbol{p}^T + \boldsymbol{q}^T\right)\boldsymbol{v} \\ & - \log p(2\boldsymbol{v}-1; \hat{\boldsymbol{\Theta}}_{\hat{\mathcal{G}}}),\end{aligned} \tag{3.20}$$

where $\boldsymbol{r}$ is a vector containing the diagonal entry of the matrix $(\boldsymbol{X}^T(\boldsymbol{I} + \sigma_n \boldsymbol{\Sigma}_x^{-1})\boldsymbol{X})$. As the pairwise terms in Eq.(3.18) reduces to a unary term, the Eq. (3.20) is much faster to evaluate. The terms $(\frac{1}{2\sigma_n}\boldsymbol{r}^T - \frac{1}{\sigma_n}\boldsymbol{y}^T\boldsymbol{A}\boldsymbol{X} + \boldsymbol{p}^T + \boldsymbol{q}^T)\boldsymbol{v}$ in Eq. (3.20) can be viewed as the unary terms ; meanwhile, $p_{\hat{\mathcal{G}}}(2\boldsymbol{v}-1; \hat{\boldsymbol{\Theta}}_{\hat{\mathcal{G}}})$ is a typical MRF (see Section 3.2.1). Therefore, Eq. (3.20) can be effectively solved by any off-the-shelf inference tools, e.g., dual decomposition [109], TWRS [110], ADLP [111]. The computational complexity for solving Eq.(3.20) depends only on the tree width of the updated MRF defined by $\hat{\boldsymbol{\Theta}}_{\hat{\mathcal{G}}}$ (see Section 3.5). Therefore, the optimization problem Eq. (3.20) is much faster to evaluate than Eq. (3.14).

3.3.3.1.B Optimization over non-zero signal coefficient variance $\boldsymbol{\Sigma}_{x,s}$ We start by calculating the covariance of sparse signal $\boldsymbol{\Sigma}_x$, then the covariance of non-zero coefficients $\boldsymbol{\Sigma}_{x,s}$ is found by choosing the diagonal member of $\boldsymbol{\Sigma}_x$ according to $\boldsymbol{s}$. Let $\nu \in \mathbb{R}_+^N$ be a vector whose members are the diagonal entry of $\boldsymbol{\Sigma}_x$. Given $\boldsymbol{x}$, $\boldsymbol{s}$, and σ_n, we have the sub-problem over $\mathbf{\Sigma}_x$ as

$$\min_{\boldsymbol{v}} \frac{1}{2}\boldsymbol{x}^T\mathbf{\Sigma}_x^{-1}\boldsymbol{x} + \frac{1}{2}\log|\sigma_n \boldsymbol{I} + \boldsymbol{A}\boldsymbol{V}\mathbf{\Sigma}_x\boldsymbol{V}^T\boldsymbol{A}^T|, \tag{3.21}$$

where $\boldsymbol{V}$ is a diagonal matrix with diagonal coefficients from $v = \frac{1}{2}(\boldsymbol{s}+1)$. From the sub-optimization over $\mathbf{\Sigma}_x$ Eq. (3.21), we let $\boldsymbol{v}$ be a vector of the diagonal entry in $\mathbf{\Sigma}_x$. Given $\boldsymbol{x}, \boldsymbol{s}$, and σ_n, we have the following optimization problem over $\mathbf{\Sigma}_x$

$$\min_{\boldsymbol{v}} \frac{1}{2}\boldsymbol{x}^T\mathbf{\Sigma}_x^{-1}\boldsymbol{x} + \frac{1}{2}\log|\sigma_n \boldsymbol{I} + \boldsymbol{A'}\mathbf{\Sigma}_x\boldsymbol{A'}^T|, \tag{3.22}$$

where $\boldsymbol{A'} = \boldsymbol{A}\boldsymbol{V}$ is the product between $\boldsymbol{A}$ and $\boldsymbol{V}$ to suppress the columns associated with zero elements in $\boldsymbol{x}$. The first term in (3.22) is convex over $\boldsymbol{v}$, while the second term is concave over $\boldsymbol{v}$. We will transform the second term into a convex function by first decomposing the logarithm term as follows:

$$\log|\sigma_n \boldsymbol{I} + \boldsymbol{A'}\mathbf{\Sigma}_x\boldsymbol{A'}^T| = \log|\mathbf{\Sigma}_{\mathbf{x}}^{-1} + \frac{1}{\sigma_n}\mathbf{A}'^T\mathbf{A}'| + \log|\sigma_n\boldsymbol{I}| + \log|\mathbf{\Sigma}_x|. \tag{3.23}$$

Let $\boldsymbol{\beta}$ be a point-wise inverse of the vector $\boldsymbol{v}$, i.e., $\boldsymbol{\beta} = \boldsymbol{v}^{\odot -1}$. We use a conjugate function to find a strict upper bound of the concave function $g(\boldsymbol{\beta}) = \log|\mathbf{\Sigma}_x^{-1} + \frac{1}{\sigma_n}\boldsymbol{A'}^T\boldsymbol{A'}|$, as follows, $\forall \alpha_i \geq 0$,

$$g(\boldsymbol{\beta}) \leq \boldsymbol{\alpha}^T\boldsymbol{\beta} - g^*(\boldsymbol{\beta}), \tag{3.24}$$

where $g^*(\boldsymbol{\beta})$ is the concave conjugate function of $g(\boldsymbol{\beta})$ and $\boldsymbol{\alpha} = [\alpha_1, ..., \alpha_K]^T$. The equation (3.24) holds when

$$\boldsymbol{\alpha}_k = \nabla_{\boldsymbol{\beta}_k} \log|\mathbf{\Sigma_x}^{-1} + \frac{1}{\sigma_n}\boldsymbol{A'}^T\boldsymbol{A'}| = \mathrm{Tr}\left[\boldsymbol{e}_k^T(\mathbf{\Sigma}_x^{-1} + \frac{1}{\sigma_n}\boldsymbol{A'}^T\boldsymbol{A'})^{-1}\boldsymbol{e}_k\right]. \tag{3.25}$$

Thus, $\boldsymbol{\alpha} = \mathbf{diag}\{(\mathbf{\Sigma}_x^{-1} + \frac{1}{\sigma_n}\boldsymbol{A'}^T\boldsymbol{A'})^{-1}\}$. Substituting Eq.(3.24) into Eq.(3.22) and using Eq.(3.23), we have the subproblem as follows:

$$\min_{\boldsymbol{v}} \boldsymbol{x}^T\mathbf{\Sigma}_x^{-1}\boldsymbol{x} + \boldsymbol{\alpha}^T\boldsymbol{\beta} + \log|\mathbf{\Sigma}_x| = \sum_{i=1}^{N}\left(\left(x_i^2 + \alpha_i\right)v_i^{-1} + \log v_i\right). \tag{3.26}$$

Because $\nu_i > 0$, the update of ν_i is

$$\nu_i^{new} = x_i^2 + \alpha_i. \tag{3.27}$$

α_i is the i-th entry of vector $\boldsymbol{\alpha} = \mathbf{diag}\{(\boldsymbol{\Sigma'}_x^{-1} + \frac{1}{\sigma_n}\boldsymbol{V}^T\boldsymbol{A}^T\boldsymbol{A}\boldsymbol{V})^{-1}\}$, and $\boldsymbol{\Sigma'}_x$ is the resulted $\boldsymbol{\Sigma}_x$ in the previous iteration. Then, $\boldsymbol{\Sigma}_{x,s}$ is a diagonal matrix where each diagonal coefficient ν_i^{new} is chosen according to $\boldsymbol{s}$.

3.3.3.1.C Optimization over noise variance σ_n Given the estimates of sparse signal, support, and non-zero coefficients' covariance—$\boldsymbol{x}_s$, $\boldsymbol{s}$ and $\boldsymbol{\Sigma}_{x,s}$, we have the sub-problem over σ_n

$$\min_{\sigma_n} \quad \frac{1}{2\sigma_n}(\boldsymbol{y} - \boldsymbol{A}_s\boldsymbol{x}_s)^T(\mathbf{y} - \boldsymbol{A}_s\boldsymbol{x}_s) + \frac{1}{2}\log|\sigma_n\boldsymbol{I} + \boldsymbol{A}_s\boldsymbol{\Sigma}_{x,s}\boldsymbol{A}_s^T|. \tag{3.28}$$

From the sub-optimization over σ_n Eq.(3.28). Let $\boldsymbol{\lambda} = \sigma_n\mathbf{1}$ be a vector where each element is noise variance σ_n. Given $\boldsymbol{\Sigma}_{x,s}$, $\boldsymbol{x}$, and $\boldsymbol{s}$, the optimization Eq.(3.28) is reformulated as

$$\min_{\boldsymbol{\lambda}}\frac{1}{2\sigma_n}||\boldsymbol{y} - \boldsymbol{A}_s\boldsymbol{x}_s||^2 + \frac{1}{2}\log|\mathbf{diag}\{\boldsymbol{\lambda}\} + \boldsymbol{A}_s\boldsymbol{\Sigma}_{x,s}\boldsymbol{A}_s^T|. \tag{3.29}$$

The concave function $h(\boldsymbol{\lambda}) = \log|\mathbf{diag}\{\boldsymbol{\lambda}\} + \boldsymbol{A}_s\boldsymbol{\Sigma}_{x,s}\boldsymbol{A}_s^T|$ is transformed into a convex function which is its upper bound, using a conjugate function. Let $h^*(\boldsymbol{\lambda})$ be the concave conjugate function of $h(\boldsymbol{\lambda})$ as follows:

$$h(\boldsymbol{\lambda}) = \log|\mathbf{diag}\{\boldsymbol{\lambda}\} + \boldsymbol{A}_s\boldsymbol{\Sigma}_{x,s}\boldsymbol{A}_s^T| \leq \boldsymbol{\eta}^T\boldsymbol{\lambda} - h^*(\boldsymbol{\lambda}), \forall \mathbf{J} \geq 0. \tag{3.30}$$

The equation (3.30) holds when

$$\eta_k = \nabla_{\lambda_k}\log|\mathbf{diag}\{\boldsymbol{\lambda}\} + \boldsymbol{A}_s\boldsymbol{\Sigma}_{x,s}\boldsymbol{A}_s^T| = \mathrm{Tr}\left[\mathbf{e}_k^T(\mathbf{diag}\{\boldsymbol{\lambda}\} + \boldsymbol{A}_s\boldsymbol{\Sigma}_{x,s}\boldsymbol{A}_s^T)^{-1}\mathbf{e}_k\right]. \tag{3.31}$$

Thus, we have $\boldsymbol{\eta} = \mathbf{diag}\{(\mathbf{diag}\{\boldsymbol{\lambda}\} + \boldsymbol{A}_s\boldsymbol{\Sigma}_{x,s}\boldsymbol{A}_s^T)^{-1}\}$.

Algorithm 3.6 Sparse signal estimation.

```
Input: measurements $\boldsymbol{y}$, a measurement matrix $\boldsymbol{A}$, and the BM parameters $\boldsymbol{\Theta}_{\mathcal{G}}$ and
the underlying graph of the MRF $\mathcal{G}$.
Initialization : $\boldsymbol{\Sigma}_x = \boldsymbol{I}_{N\times N}$, $\sigma_n = 1$, and $\boldsymbol{x} = \boldsymbol{0}$
  while A stopping criterion is not satisfied do
      1. Update the support $\boldsymbol{s}$ by solving Eq. (3.20)
      2. Update the covariance matrix $\boldsymbol{\Sigma}_{x,s}$ as Eq. (3.27)
      3. Update the noise variance $\sigma_n$ as Eq. (3.33)
      4. Update the sparse signal $\boldsymbol{x}_s$ as Eq. (3.35)
  end while
Output: $\boldsymbol{x}$ whose non-zero coefficients are from $\boldsymbol{x}_s$.
```

Substituting (3.30) into (3.29), we obtain the following reformulated sub-problem over λ:

$$\min_{\boldsymbol{\lambda}} \frac{1}{\sigma_n}(\boldsymbol{y} - \boldsymbol{A}_s\boldsymbol{x}_s)^T(\boldsymbol{y} - \boldsymbol{A}_s\boldsymbol{x}_s) + \boldsymbol{\eta}^T\boldsymbol{\lambda} = \sum_{i=1}^{M}\left(\frac{b_i^2}{\lambda_i} + \lambda_i\eta_i\right), \tag{3.32}$$

where b_i denotes the $i-$th entry of $\boldsymbol{b} = \boldsymbol{y} - \boldsymbol{A}_s\boldsymbol{x}_s$. Because $\boldsymbol{\lambda} > 0$, we obtain $\lambda_i^{new} = \sqrt{\frac{b_i^2}{\eta_i}}$. Thus, this gives rise to a closed-form solution for σ_n as

$$\sigma_n^{new} = \frac{1}{M}\sum_{i=1}^{M}\sqrt{\frac{b_i^2}{\eta_i}} \tag{3.33}$$

where η_i is the i-th entry of vector $\boldsymbol{\eta} = \mathbf{diag}\{(\sigma_n\boldsymbol{I} + \boldsymbol{A}_s\boldsymbol{\Sigma}_{x,s}\boldsymbol{A}_s^T)^{-1}\}$, and b_i is the i-th entry of $\boldsymbol{b} = \boldsymbol{y} - \boldsymbol{A}_s\boldsymbol{x}_s$.

3.3.3.1.D Optimization over non-zero signal coefficients $\boldsymbol{x}_s$ Given the estimates of support, noise variance, and non-zero coefficients' covariance—$\mathbf{s}$, σ_n, and $\boldsymbol{\Sigma}_{x,s}$, the subproblem for $\boldsymbol{x}_s$ is

$$\min_{\boldsymbol{x}_s} \frac{1}{\sigma_n}(\boldsymbol{y} - \boldsymbol{A}_s\boldsymbol{x}_s)^T(\boldsymbol{y} - \boldsymbol{A}_s\boldsymbol{x}_s) + \boldsymbol{x}_s^T\boldsymbol{\Sigma}_{x,s}^{-1}\boldsymbol{x}_s, \tag{3.34}$$

which shows a closed-form updated equation as

$$\boldsymbol{x}_s^{new} = (\sigma_n\boldsymbol{\Sigma}_{x,s}^{-1} + \boldsymbol{A}_s^T\boldsymbol{A}_s)^{-1}\boldsymbol{A}_s{}^T\mathbf{y}. \tag{3.35}$$

How to solve Eq. (3.13) is summarized in Algorithm 3.6 where the sparse signal, support , and noise and signal parameters are jointly estimated in a unified framework.

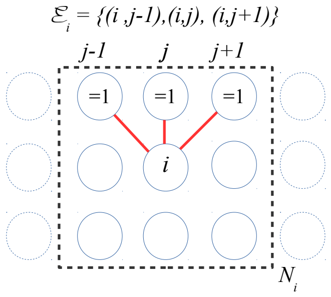


FIGURE 3.3: Example of how the graph is updated

In Algorithm 3.6, we solve the support estimation problem Eq. (3.20) in step 1 by performing graphical inference using the belief propagation implemented by [83].

Next, we turn to the MRF parameters estimation Eq.(3.12) to update the MRF parameters—-the BM parameters and the underlying graph.

3.3.3.2 MRF parameters estimation

This section focuses on solving the MRF parameter estimation problem Eq.(3.12) to update the BM parameters and the underlying graph. Given the point estimates of sparse signals, we calculate a binary vector $\boldsymbol{d}$ whose coefficients correspond to the high-energy coefficients of the resulting sparse signal $\boldsymbol{x}$. Notice that $\boldsymbol{d}$ is not necessarily similar to the intermediate estimate support $\boldsymbol{s}$, thus, preventing overfitting to the previous estimated MRF parameters.

3.3.3.2.A Graph update procedure In practice, we can simplify the graph estimation task, as suggested in [27], by forming a graph according to non-zero coefficients or high energy coefficients in sparse signals, which carry information about signal structure. Let $\boldsymbol{d} \in \{-1, 1\}^N$ be a binary vector whose coefficients correspond to the high-energy coefficients of the resulting sparse signal $\boldsymbol{x}$. $d_i = 1$ indicates that x_i has a high-energy coefficient, and $d_i = -1$ indicates that x_i has a negligible value. Here, each coefficient in $\boldsymbol{d}$ is mapped to each node in the graph, and each node d_i only forms edges to adjacent nodes with a positive value within a radius of neighborhood $\mathbb{N}_i$.

Figure 3.3 illustrates how each edge in the graph is updated for capturing the two-dimensional structure in an image. Each pixel is mapped onto a node in the graph. Let d_i be the node of interest and $\mathcal{E}_i$ denote a local edge set where $\mathcal{E}_i \in \mathcal{E}$. Edges are

Algorithm 3.7 Graph update procedure $\mathcal{G}$

```
Input: Binary vector $\boldsymbol{d}$.
Initialization : $\mathcal{E}_i = \emptyset \quad \forall i = 1, ..., N$, $\mathcal{E} = \emptyset$, and the node set contains the node where each of which corresponds to each coefficient in the binary vector $\mathcal{V} = \{d_1, ..., d_N\}$ .
  for $i = 1, ..., N$ do
    for each $j \in \mathbb{N}_i$ do
      Include the edge $(i, j)$, if $d_j = 1$ and the edge $(j, i) \notin \mathcal{E}$ is not present
      $\mathcal{E}_i = \mathcal{E}_i \bigcup (i, j)$ .
    end for
    $\mathcal{E} = \mathcal{E} \bigcup \mathcal{E}_i$.
  end for
Output: $\mathcal{G} = (\mathcal{V}, \mathcal{E})$.
```

established by connecting a node d_i to adjacent nodes with value '1' located within a radius of neighborhood $\mathbb{N}_i$ covering 8-neighborhood. As the adjacent nodes d_{j-1}, d_j, and d_{j+1} are equal to one, the edges $(i, j-1)$, (i, j), and $(i, j+1)$ are included into the local edge set of the node $\mathcal{E}_i$. If all adjacent nodes in the radius of the neighborhood of d_i have the value -1, $\mathcal{E}_i$ is an empty set. Algorithm 3.7 summarizes the graph update procedure. The graph update does not require high computation and has higher flexibility and adaptability than using a fixed neighborhood graph in the clustered structure sparsity models [37]–[44] where every node is connected to all adjacent nodes in each neighborhood.

3.3.3.2.B BM parameter estimation Given the binary vector $\boldsymbol{d}$ and the graph $\mathcal{G}$, we solve the MAP problem (3.12) using the pseudo-likelihood algorithm [84], [112] which requires low computation and is suitable for adaptively estimating the BM parameters. In pseudo-likelihood [84], it is assumed that d_i and d_j are conditionally independent given the neighborhood of d_i. Thus, the work [84] aims to maximize $\prod_i p(d_i|\boldsymbol{d}_{\mathcal{E}_i}, \boldsymbol{\Theta}_{\mathcal{G}})$, where $\boldsymbol{d}_{\mathcal{E}_i}$ are the adjacent neighbors connected to the node d_i through edges defined by a local edge set $\mathcal{E}_i$:

$$\boldsymbol{\theta}_{\mathcal{G}} = \max_{\boldsymbol{\theta}_{\mathcal{G}}} \prod_{i=1}^{N} p(d_i|\boldsymbol{d}_{\mathcal{E}_i}, \boldsymbol{\Theta}_{\mathcal{G}}), \tag{3.36}$$

$$\text{where} \quad p(d_i|\boldsymbol{d}_{\mathcal{E}_i}, \{\gamma_{i,j}, \gamma_i\}_{(i,j)\in\mathcal{E}, i\in\mathcal{V}}) = \frac{1}{Z(\boldsymbol{d}_{\mathcal{E}_i}; \{\gamma_{i,j}, \gamma_i\})} \exp\big(d_i\gamma_i + \sum_{(i,j)\in\mathcal{E}_i} d_i\gamma_{i,j}d_j\big).$$

```
Algorithm 3.8 Two-step-Adaptive Markov Random Field (TA-MRF).
Input: Measurements $\boldsymbol{y}$ and random matrix $\boldsymbol{A}$
Initialization: Get $\boldsymbol{x}$ from Algorithm 3.6 where step 1 is removed and replaced with a fixed $\boldsymbol{s} = \mathbf{1}$
  while A stopping criterion is not satisfied do
      1. Obtain a binary vector $\boldsymbol{d}$ from thresholding each of $\boldsymbol{x}$, i.e., $d_i = 1$, if $\mathrm{abs}(x_i) > \mathrm{mean}(\mathrm{abs}(\boldsymbol{x}))$,
         and $d_i = -1$ otherwise;
      2. Calculate $\mathcal{G}$ from $\boldsymbol{d}$ following Algorithm 3.7;
      3. Learn $\boldsymbol{\Theta}_{\mathcal{G}}$ from $\boldsymbol{d}$ and $\mathcal{G}$ by solving Eq. (3.12);
      4. Update $\boldsymbol{x}$ by solving Eq.(3.13) with Algorithm 3.6;
  end while
Output: Recovered $\boldsymbol{x}_{rec}$.
```

$\{\gamma_{i,j}\}$ and $\{\gamma_i\}$ are the pairwise and unary parameters in the Boltzmann machine. To estimate these parameters, a gradient descent is employed. The gradient function to update each parameter is defined as follows:

$$\begin{aligned}
&\gamma_i(t+1) = \gamma_i(t) + \rho \left(d_i + 1 - \left(\frac{2}{1 + \exp -2(\gamma_i(t) + \sum_{(i,j)\in\mathcal{E}_i} \gamma_{i,j}(t) d_j)} \right) \right); \\
&\gamma_{i,j}(t+1) = \gamma_{i,j}(t) + \rho \left(2d_i d_j + d_i + d_j - T_{i,j} \right),
\end{aligned} \tag{3.37}$$

where

$$T_{i,j} = \left(\frac{2d_j}{1 + \exp -2(\gamma_i(t) + \sum_{(i,j)\in\mathcal{E}_i} \gamma_{i,j}(t) d_j)} \right) + \left(\frac{2d_i}{1 + \exp -2(\gamma_j(t) + \sum_{(i,j)\in\mathcal{E}_i} \gamma_{i,j}(t) d_i)} \right).$$

For more details on the background and derivation, we refer readers to Chapter 2.4.2.

Given Algorithm 3.6 to update sparse signals, the whole Two-step-Adaptive MRF is summarized in Algorithm 3.8 where the MRF parameter estimation is performed in steps 2 and 3. Notice that the underlying graph in step 2 is estimated based on the binary vector $\boldsymbol{d}$ from step 1 that truncates the negligible sparse signal coefficients in $\boldsymbol{x}$ to zero. Thus, the underlying graph of the MRF captures the structure of high energy coefficients in $\boldsymbol{x}$, which can be different from the intermediate estimates of the support in Algorithm 3.6. The alternative minimization scheme reduces the objective functions—the MRF parameter estimation and sparse signal estimation in Algorithm 3.8—in each iteration. The objective functions can be proved to be bounded from below. Thus, the Two-step-Adaptive MRF converges well as [102]. To confirm this, we also provide empirical convergence of Two-steps-Adaptive MRF in Section 3.6.8.

3.4 The essence of the adaptive signal prior to guarantee PRIP.

The goal of adaptive MRF is to adaptively estimate the MRF for the actual structure of sparse signal. Here, we propose the Theorem 3.4.2 that reveals the connection between adaptive support prior and probabilistic RIP (PRIP) [29], for achieving the theoretical optimal sample complexity. At first, we will review the PRIP [29] (Lemma 2.3.3). Then, Theorem 3.4.2 will be presented.

Let $\boldsymbol{x}$ denote a ground truth k-sparse signal whose support $\hat{\mathbf{s}} = \mathrm{supp}(\boldsymbol{x})$ is generated by a known probabilistic model $\mathcal{P}$. $\Omega_{k,\varepsilon}$ denotes the smallest set of candidate support captured by a learned model $\mathcal{M}$.

Lemma 3.4.1. *[29] . If the probability that the true support can be represent by a candidate support in $\Omega_{k,\varepsilon}$ is higher than $1-\varepsilon$, i.e. $p(\hat{\mathbf{s}} \in \Omega_{k,\varepsilon}) > 1-\varepsilon$, a sub-Gaussian random matrix $\boldsymbol{A} \in \mathcal{R}^{M\times N}$ satisfies the (k,ε)-PRIP with probability at least $1-e^{-c_2 M}$ with $M \geq c_1(k+\log(|\Omega_{k,\varepsilon}|))$, where $c_1, c_2 > 0$ depends only on the PRIP constant $\delta_k \in [0,1]$.*

Notice that the members $\Omega_{k,\varepsilon}$ are chosen based on the support prior model $\mathcal{M}$, e.g. an MRF learned from training. However, if $\mathcal{M}$ is learned based on the training data that cannot well represent the testing signals, then the necessary condition of the lemma can be violated.

To address this problem, we propose the concept of adaptive support prior to realize the $\Omega_{k,\varepsilon}$ that can well represent the test signal. To do this, we study a sequence of random support vector $S_1, ..., S_n$ corresponding to the adapted support prior $\mathcal{M}_1, ..., \mathcal{M}_n$ such as Eq. (3.3).

Theorem 3.4.2. *For a fixed ground truth support $\hat{\mathbf{s}} = supp(\boldsymbol{x})$, if $S_1, ..., S_n$ converges to $\hat{\mathbf{s}}$ in distribution, i.e. $\lim_{n\to\infty} \mathcal{M}_n(S_n) = \mathcal{P}(\hat{\mathbf{s}})$, then we can show that $\lim_{n\to\infty} p(\hat{\mathbf{s}} \in \Omega_k^n) = 1$ where Ω_k^n is the ball containing the random variable support S_n with center c and radius 2ϵ.*

Proof. Since $\hat{s}$ is a fixed ground truth support, then convergence in distribution implies convergence in probability. That is, if $\lim_{n\to\infty} \mathcal{M}_n(S_n) = \mathcal{P}(\hat{\mathbf{s}})$, then $\lim_{n\to\infty} p(||S_n - \hat{\mathbf{s}}|| < \epsilon) = 1$. Given that Ω_k^n is the ball containing the ensembles of the random variable support S_n with center c and radius 2ϵ, then $\hat{\mathbf{s}} \in \Omega_k^n$ with probability one.

Theorem 3.4.2 suggests that if the adapted support model $\mathcal{M}_n(S_n)$ can represent the true probability $\mathcal{P}$, it is guaranteed that the set Ω_k^n always contain a candidate support that truly represents the ground truth support (i.e. $\varepsilon = 0$). The smallest size of Ω_k^n is one. Therefore, the minimum measurements can achieve the theoretical sampling complexity, i.e. $M \geq c_1(k + \log(|\Omega_{k,0}|))$ where $|\Omega_{k,0}|$ is smallest (e.g. $|\Omega_k^n| = 1$). Thus, $M \approx \mathcal{O}(k)$.

3.5 Algorithm Complexity

The dominant computation of the proposed method is the computation in the sparse signal estimation algorithm (Algorithm 3.6) involving the computation of:

1. *Matrix inversion* with complexity of $\mathcal{O}(N^3 + M^3 + \hat{k}^3)$ from the signal coefficient variance estimation Eq. (3.27), in noise estimation Eq. (3.33); and in estimating the sparse coefficient Eq. (3.35);

2. *Matrix production* from the support estimation (3.20), the signal coefficient variance estimation Eq. (3.27), in noise estimation Eq. (3.33); and in estimating the sparse coefficient Eq. (3.35) that has a complexity of $\mathcal{O}(5N^2 + 2N^2M + MN + 5N)$, $\mathcal{O}(N^2M + 3N^2 + 3N)$, $\mathcal{O}(M^2\hat{k} + M\hat{k} + 4M)$, and $\mathcal{O}(M\hat{k}^2 + M\hat{k} + \hat{k}^2 + 2k)$, respectively;

3. *Support estimation* Eq. (3.20) . The complexity of support estimation Eq. (3.20) using the belief propagation algorithm based on MAP LP-relaxations[83] is $\mathcal{O}(t_{max}|\mathcal{E}|)$, where $|\mathcal{E}| = N|\mathbb{N}|$ and $\mathbb{N}$ is the largest set of neighboring nodes $\{\mathbb{N}_i\}$. t_{max} denotes the maximum number of iterations for performing the belief propagation. If $\mathbb{N}$ covers only two adjacent nodes such as in a chain graph, $|\mathbb{N}|$ can be very small. For this special case, $\mathcal{O}(t_{max}N|\mathbb{N}|) = \mathcal{O}(2t_{max}N)$.

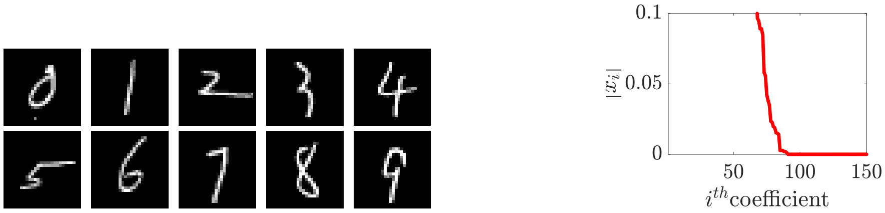
The computational complexity of the proposed sparse signal estimation is $\mathcal{O}(N^3 + M^3 + \hat{k}^3 + 8N^2 + 4N^2M + M^2N + 3MN + t_{max}N|\mathbb{N}|)$. The computational complexity can be reduced to $\mathcal{O}(2M^3 + 2MN^2 + 4M^2N + 5N^2 + MN + t_{max}N|\mathbb{N}|)$ by employing off-line computation for $\boldsymbol{A}^T\boldsymbol{A}$ and a matrix inversion property:

$$\begin{aligned}(\boldsymbol{\Sigma}'^{-1}_{x} + \frac{1}{\sigma_n}\boldsymbol{V}^T\boldsymbol{A}^T\boldsymbol{A}\boldsymbol{V})^{-1} = \boldsymbol{\Sigma}'_x - \\ \boldsymbol{\Sigma}'_x\boldsymbol{V}^T\boldsymbol{A}^T(\sigma_n\boldsymbol{I} + \boldsymbol{A}\boldsymbol{V}\boldsymbol{\Sigma}'_x\boldsymbol{V}^T\boldsymbol{A}^T)^{-1}\boldsymbol{A}\boldsymbol{V}\boldsymbol{\Sigma}'_x.\end{aligned} \tag{3.38}$$

Meanwhile, the value of $\boldsymbol{A}^T\boldsymbol{y}$ needs to be computed only once and can be reused. The computational complexity depends not only on the dimension of the signals but also the structure of the graph $\mathcal{G}$.

In comparison with existing MRF-based methods which are MAP-OMP [32] and Gibbs[31] that solves Eq. (3.8), the complexity of our method can be seen as higher in general. The complexity of MAP-OMP [32] is $\mathcal{O}(N(\hat{k}^3 + \hat{k}^2 + M^2\hat{k} + M\hat{k}))$ where $\mathcal{O}(N(\hat{k}^3))$ corresponds to computing the matrix inversion of size $\hat{k} \times \hat{k}$ and the rest $\mathcal{O}(N(\hat{k}^2 + M^2\hat{k} + M\hat{k}))$ corresponds to the computation of matrix multiplications. These two matrix operations are performed up to N times in each iteration (see Section 2.2.2.2). In the worst case scenarios, where the estimated sparse signal contains all non-zero elements $\hat{k} = N$, the computational complexity can approach $\mathcal{O}(N^4 + M^2N^2 + N^3 + MN^2)$ per iteration, which is one order higher than ours. Meanwhile, the complexity of Gibbs [31] is $\mathcal{O}(\hat{k}N^2 + N^2)$ which can rise to $\mathcal{O}(N^3 + N^2)$ as the Gibbs sampling approach requires $\mathcal{O}(\hat{k}N^2)$ to update the covariance matrix Eq. (2.19) for the chosen $\hat{k}$ support coefficients and $\mathcal{O}(N^2)$ to calculate the vector multiplication for N support coefficients (see Section 2.2.2.1). Despite having the lowest computational cost in each iteration, the convergence of Gibbs sampling can be very slow. This problem becomes more obvious when applying MAP-OMP and Gibbs to estimate sparse signals in the Two-step-Adaptive MRF framework. We compare the performance of our sparse signal estimation against those of MAP-OMP [32] and Gibbs [31] in Section 3.10. It can be seen that our runtime per iteration is moderately stable, compared with the other two methods. The runtime of MAP-OMP increases obviously in recovering wavelet sparse representation of CMU-IDB and CIFAR-10. Meanwhile, Gibbs converges very slowly.

The computation of Algorithm 3.6 is included in step 4 of Algorithm 3.8 (our Two-step-Adaptive MRF). The total runtime performance of our Two-step-Adaptive



(A) Ground-truth digit images

(B) The image pixels decay

FIGURE 3.4: MINST. (A) The ground truth handwritten digit images. (B) The pixel coefficient's decay.

MRF is provided in Figure 3.16. It can be seen that the runtime of our method is moderate among all methods.

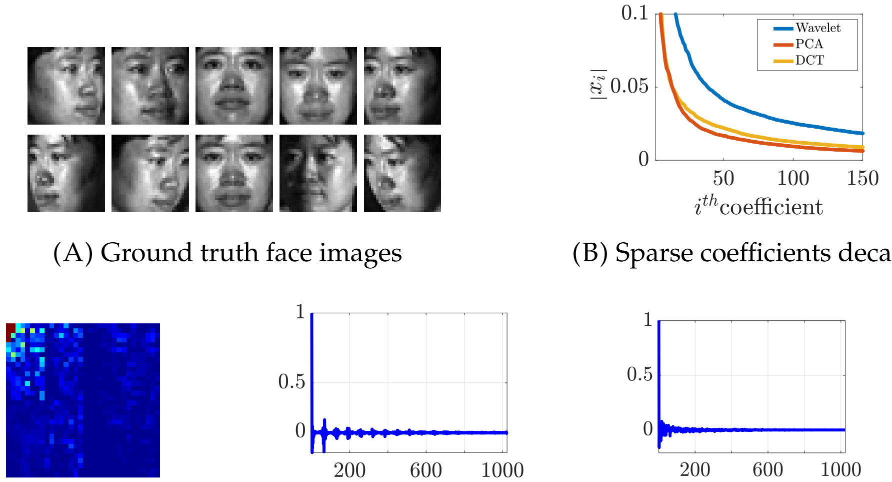
3.6 Experimental Results and Analysis

In this section, we study the performance of the proposed Two-step-Adaptive MRF through performing three different experiments: (i) to study the effectiveness of the adaptive mechanism, we study the performance of the adaptive MRF in comparison with using a fixed MRF that is learned from training samples in Section 3.6.5; (ii) then, we study the performance of our proposed sparse signal estimation that solves Eq. (3.13) in comparison with the existing MRF-based methods [32] and [31] that solve Eq. (3.8) in a two-step framework in Section 3.6.6; (iii) we compare the performance of the proposed Two-step-Adaptive MRF with state-of-the-art competitors in compressibility, noise tolerance, and runtime in Section 3.6.7; and (iv) finally, we study the empirical convergence of the proposed algorithm in Section 3.6.8.

We test the performance on three datasets— MNIST [113], CMU-IDB [114], and CIFAR-10 [115]— which exhibit different characteristics, detailed in Section 3.6.1. The experiment setting, comparison methods, and evaluation criteria are given in Section 3.6.2, 3.6.3, and 3.6.4.

3.6.1 Dataset

We evaluate the performance on three datasets— MNIST [113], CMU-IDB [114], and CIFAR-10 [115]— which exhibit different characteristics: *i) MNIST handwritten digit images* [113] contain few lines and are strictly sparse where the clustering of the non-zero coefficients is structured in a long-continued line; *ii) the CMU-IDB face*



(A) Ground truth face images

(B) Sparse coefficients decay

(C) Wavelet signal

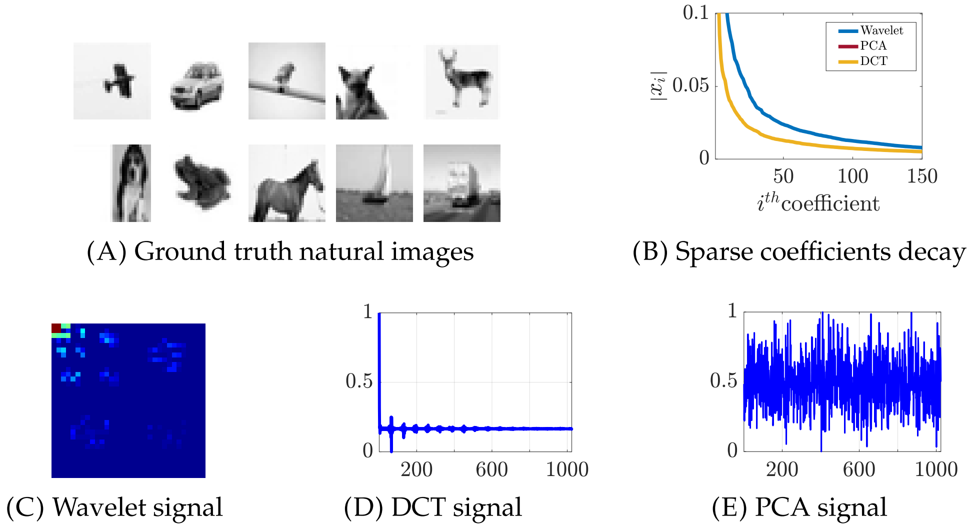
(D) DCT signal

(E) PCA signal

FIGURE 3.5: CMU-IDB. (A) The ground truth face images; (B) The decay of sparse signal coefficients in wavelet, PCA and DCT domains. Examples of (C) the wavelet signal, (D) DCT signal, and (E) PCA signal.

images [114] contain facial features which have dense spatial information and are more diverse than the MNIST images; *iii) the CIFAR-10 natural images* [115] are more diverse and less synthesized than the previous two datasets. They reflect performance on typical images. The test images selected from each dataset for the experiment are shown in Figures 3.4A, 3.5A, and 3.6A.

The MNIST digit images are strictly sparse, as shown in the pixel decay curve Figure 3.4B. The compression process can be applied onto the signals directly. However, the images from CMU-IDB and CIFAR-10 datasets are not sparse. Their sparse representation can be obtained by transforming these images into an appropriate basis. Here, we exploit i) wavelet transform, ii) discrete cosine transform (DCT), and iii) principal component analysis (PCA) to obtain these sparse representations. Examples of the sparse representations in wavelet, DCT, and PCA domains of CMU-IDB and CIFAR-10 images are in Figures 3.5C, 3.5D, 3.5E and Figures 3.6C, 3.6D, 3.6E, respectively. Note that all these signal representations are compressible, except the signal representations of CIFAR-10 images in the PCA domain. The PCA signal is very dense; thus, it violates the sparsity assumption of compressive sensing. As a result, we omit the discussion of CIFAR-10 images in the PCA domain, and focus on the results of six sets of images: (1) the MNIST digit images in the spatial domain, (2) CMU-IDB images in the PCA domain, (3) CMU-IDB images in the wavelet domain,



(A) Ground truth natural images

(B) Sparse coefficients decay

(C) Wavelet signal

(D) DCT signal

(E) PCA signal

FIGURE 3.6: CIFAR-10. (A) The ground truth natural images. (B) The decay of sparse signal coefficients in wavelet, PCA and DCT domains. Examples of (C) the wavelet signal, (D) DCT signal, and (E) PCA signal.

(4) CMU-IDB images in the DCT domain, (5) CIFAR images in the wavelet domain, and (6) CIFAR images in the DCT domain.

3.6.2 Experimental settings

In the compression, the sparse signal $\boldsymbol{x}$ is sampled by a random Bernoulli matrix $\boldsymbol{A}$ to generate the linear measurements $\boldsymbol{y}$. The recovery performance is tested across different sampling rates (M/N), i.e., 0.2, 0.25, 0.3, 0.35, and 0.4. To simulate the noise corruption on the measurements, four different levels of Gaussian white noise are added into the measurements $\boldsymbol{y}$, which results in the signal to noise ratio (SNR) being 5 dB, 10 dB, 20 dB, and 30 dB. Note that at the lowest SNR (5 dB), the measurements are mostly corrupted by noise. Thus, the lowest SNR indicates the highest noise corruption[1].

Algorithm Setting. The proposed Two-step-Adaptive MRF (Algorithm 3.8: the main algorithm) will stop when the minimum update difference of $\boldsymbol{x}$ from step 4 is less than 10^{-3}, or when the iteration reaches 5 iterations. In step 2, to capture the 2-D structure in wavelet and handwritten images, $\mathbb{N}_i$ is set to cover 8 neighboring nodes. Meanwhile, to capture the 1-D structure of DCT and PCA signals, $\mathbb{N}_i$ is set to cover the two adjacent nodes of the i^{th} node. In step 3, the maximum iteration

[1] The noise level (in SNR) from 5 dB to 30 dB indicates the highest to the lowest noise corruption

for gradient descent to estimate the BM parameters is set to 20. In step 4, the sparse signal estimation is performed by Algorithm 3.6 which is set to terminate when the minimum update difference of x_s is less than 10^{-3}, or when the iteration reaches 200. The minimum update difference between the two consecutive estimates of x is defined as

$$\eta = \frac{||\mathbf{x}^{new} - \mathbf{x}||_2}{||\mathbf{x}||_2}. \tag{3.39}$$

3.6.3 Comparison methods

The performance of our method is compared with 8 state-of-the-art competitors:

- **Existing MRF-based methods**: MAP-OMP[2] [32], Gibbs[2] [31] —whose support estimations are based on solving the optimization problem Eq.(3.8) with heuristic and stochastic approaches;
- **Clustering structured sparsity-based methods**: Bernoulli[3][39] and Pairwise MRF[3] [42];
- **Graph sparsity-based methods**: GCoSamp [27] and StructOMP [25]
- **Sparsity-based methods**: a Bayesian-based method RLPHCS[103] and a standard signal recovery method OMP[106].
- We use *the oracle estimator* suggested in [32] that uses **the ground truth support** to estimate the signal (via Eq. (3.35)). Note that all other methods *do not have* access to the ground truth support. The oracle estimator has this unfair advantage, and we use it to show the best possible result using ground truth support with homogeneous noise parameters.

All of the comparison methods, except Pairwise MRF [42], are implemented by the code of the authors with tuned parameters to the best performance. For Pairwise MRF, we implemented the code ourselves. We set the Pairwise MRF algorithm to terminate when the minimum update difference is less than 10^{-3}, or when the iteration reaches 200.

[2] The graphical model, noise and signal variance parameters, provided to MAP-OMP and Gibbs, is from training data.

[3] For both Bernoulli and Pairwise MRF , we use the same setting for neighboring set $\mathbb{N}_i$, as described in Algorithm Setting in Section 3.6.2

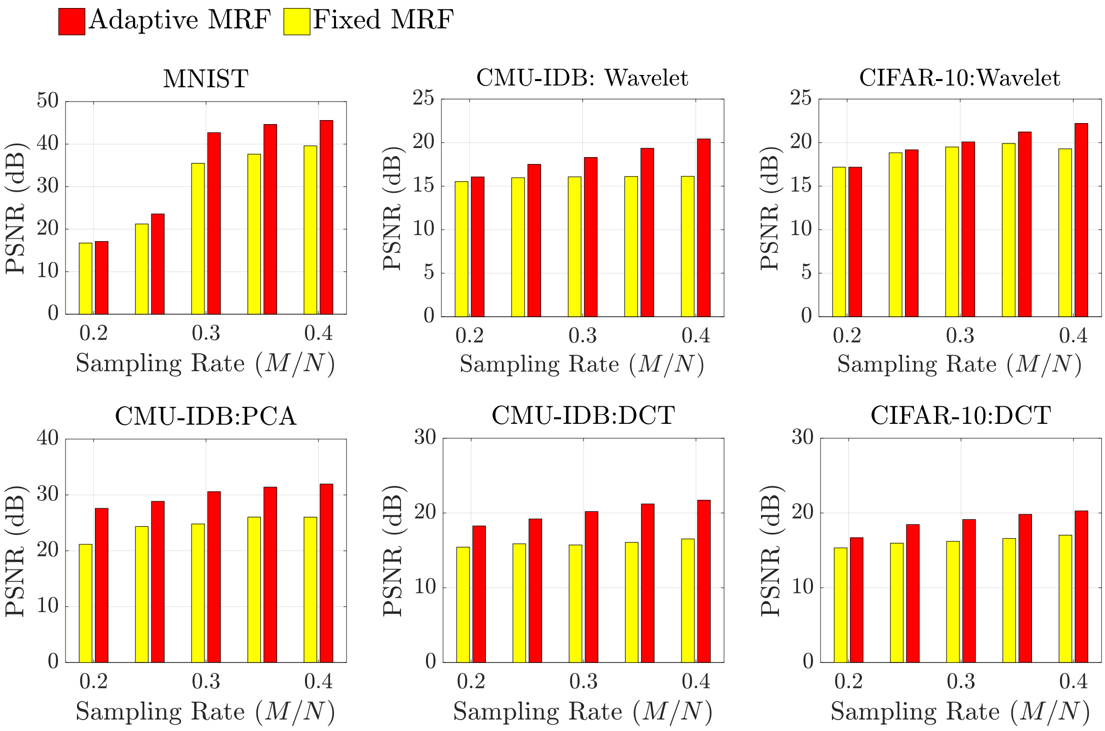


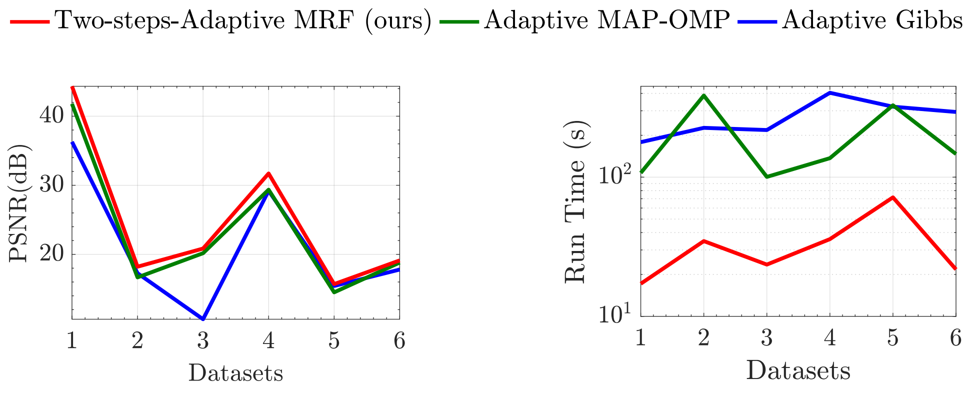
FIGURE 3.7: Comparison of Adaptive-MRF versus Fixed-MRF under noise level (SNR) of 30 dB on MNIST images; PCA, wavelet, and DCT signals of CMU-IDB images; and wavelet, and DCT signals of CIFAR-10 images.

3.6.4 Evaluation criterion

We demonstrate the proposed Two-step-Adaptive MRF performance on recovery accuracy, noise tolerance, and runtime performance. The recovery accuracy is evaluated by the peak signal to noise ratio (PSNR). To evaluate the runtime performance, we provide the total runtime curves across different sampling rates (M/N).

3.6.5 Effectiveness of the proposed adaptive MRF

To demonstrate the improved performance of the proposed adaptive MRF, we compare the performance of the Two-step-Adaptive MRF when the MRF is adaptive versus when the MRF is fixed. When we describe that the MRF as fixed, we mean the sparse signal estimation (Algorithm 3.6) exploits an MRF whose underlying graph and parameters are obtained from training (off-line), thus, is fixed throughout the signal recovery process. Thus, the performance of the Two-step-Adaptive MRF when using adaptive MRF is denoted as *Adaptive-MRF*; meanwhile, the performance of the Two-step-Adaptive MRF when using a fixed MRF is denoted as the *Fixed-MRF*.



(A) Accuracy

(B) Runtime

FIGURE 3.8: Solving Eq. (3.10) (our Two-step-Adaptive MRF) vs solving Eq. (3.8) directly (Adaptive-Gibbs and Adaptive-MAP-OMP): (A) recovery accuracy and (B) total runtime on 6 sets of images: (1) the MNIST images, (2)(3)(4) CMU-IDB images in wavelet, DCT, and PCA domains, and (5)(6) CIFAR-10 images in wavelet and DCT domains. The sampling rate is 0.3 and noise level (SNR) is 30 dB.

Figure 3.7 shows the bar graph of the average PSNR value across different sampling rates on the three datasets—MNIST, CMU-IDB, and CIFAR-10— at noise level (SNR) of 30 dB. It is clear that the Adaptive MRF outperforms the Fixed-MRF in all cases, especially when the sampling rate (M/N) is higher than 0.2. On MNIST images, the Adaptive MRF outperforms the Fixed-MRF by at least 2 dB. On CMU-IDB images, the Adaptive MRF outperforms the Fixed-MRF by at least 2 dB in recovering the wavelet images, 3 dB in recovering the PCA signals, and 2 dB in recovering the DCT signals. On CIFAR-10 images, the Adaptive MRF outperforms the Fixed-MRF by at least 0.5 dB in recovering the wavelet signals and 2 dB in recovering the DCT signals.

3.6.6 Effectiveness of the proposed sparse signal estimation

In this section, we demonstrate the effectiveness of our sparse signal estimation to obtain the sparse signal from solving the new optimization problem Eq. (3.10) whose cost function is the upper bound approximation of Eq. (3.8). Here, we compare the performance of our sparse signal estimation against Gibbs [31] and MAP-OMP [32] that attempt to solve Eq. (3.8) directly with the stochastic and heuristic approaches. All the algorithms are tested in the same two-step framework setting: first, the MRF parameters are adaptively estimated based on an estimated sparse signal, and then the sparse signal is estimated by each algorithm given the resulting MRF. Thus, we compare our Two-step-Adaptive MRF against Adaptive-Gibbs (Gibbs + the two-step

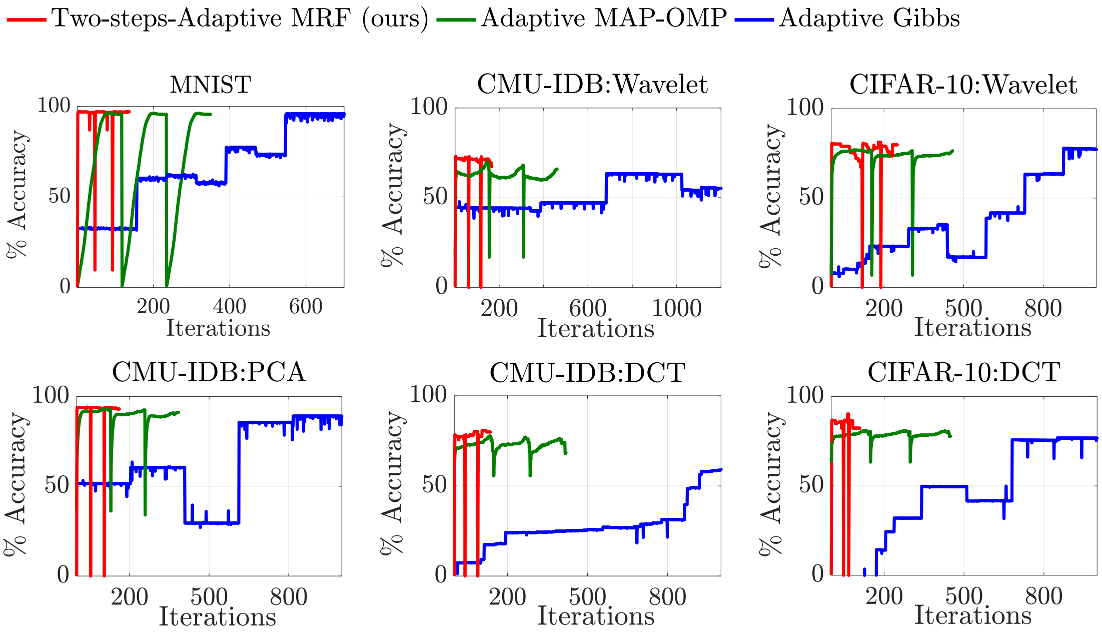


FIGURE 3.9: Convergence of accuracy: Solving Eq. (3.10) (our Adaptive-MRF) vs solving Eq. (3.8) directly (Adaptive-Gibbs and Adaptive-MAP-OMP) on MNIST images; PCA, wavelet, and DCT signals of CMU-IDB images; and wavelet, and DCT signals of CIFAR-10 images. Sampling rate is 0.3 and noise level (SNR) is 30 dB.

framework) and Adaptive-MAP-OMP (MAP-OMP + the two-step framework). The two-step framework performs at the main-loop which is set to terminate when its iterations reach 3. The sparse signal estimation performs at the inner-loop which is set to terminate when its iterations reach 1000, or when minimum update differences between two consecutive estimates of x are less than 10^{-5}.

Figure 3.8 illustrates the recovery performance across six sets of images (no. 1-6): no. (1) denotes the set of the MNIST images ; no. (2)(3)(4) denote the sets of sparse representation of CMU-IDB images in the wavelet, DCT, and PCA domains; and no. (5)(6) denote the sets of sparse representation of CIFAR-10 natural images in the wavelet and DCT domains. The performance is tested at the sampling rate and noise level (SNR) of 0.3 and 30 dB, respectively. It is clear that Two-step-Adaptive MRF requires the least runtime and provides the highest accuracy in all cases. Adaptive-MAP-OMP and Adaptive-Gibbs have their performance improved in comparison when using the trained MRF (see Figure 3.11). This suggests that the adaptive MRF helps improve the performance of these algorithms as well.

Figure 3.9 and 3.10 further examine the convergence in terms of recovery accuracy and runtime of the proposed Two-step-Adaptive MRF against Adaptive-Gibbs and

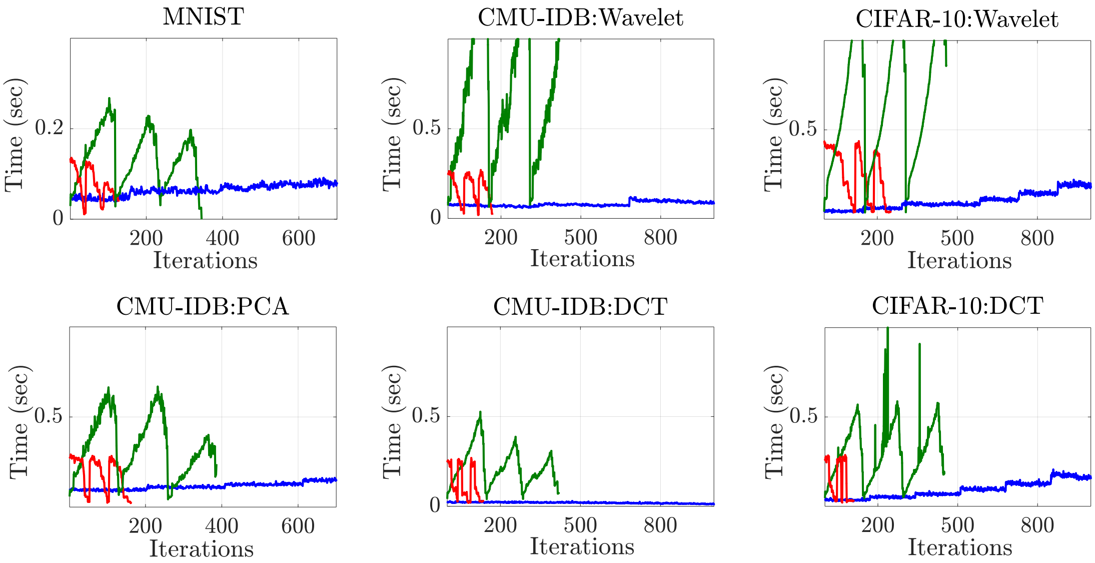


FIGURE 3.10: Executing time per iteration: Solving Eq. (3.10) (our Adaptive-MRF) vs solving Eq. (3.8) directly (Adaptive-Gibbs and Adaptive-MAP-OMP) on MNIST images; PCA, wavelet, and DCT signals of CMU-IDB images; and wavelet, and DCT signals of CIFAR-10 images. Sampling rate is 0.3 and noise level (SNR) is 30 dB.

Adaptive-MAP-OMP on MNIST, CMU-IBD, and CIFAR-10 datasets. These results are averaged over 10 images in each image set. *iterations* on the horizontal axis of each graph denotes the total iterations that the sparse signal estimation performs throughout the two-step framework. Here, we measure the recovery accuracy and runtime in the process of the sparse signal estimation, which is recursively performed by the two-step framework.

In Figure 3.9, our Two-step-Adaptive MRF achieves the highest accuracy and requires many fewer iterations to converge. Note that there are three ripples on the accuracy curves of both the proposed Two-step-Adaptive MRF and Adaptive-MAP-OMP, according to the setting to execute the main-loop 3 times. All these curves contain spikes and downward curves in addition to these ripples. Because all these methods only try to achieve a point estimate, the resulting accuracy can be slightly unstable. The proposed sparse signal estimation in Two-step-Adaptive MRF jointly and recursively estimates the sparse signal and support; thus, the proposed Two-step-Adaptive MRF is more stable than the others. It is slightly unstable in recovering CMU-IDB images in the wavelet domain and in recovering CIFAR-10 images in the DCT domain. Meanwhile, the sparse signal estimations in Adaptive-MAP-OMP and Adaptive-Gibbs are non-recursive. The error in support estimation can be accumulated in the sparse signal estimation (Algorithm 2.3). These methods are prone to error accumulation problems. The recovery accuracy curves of Adaptive-Gibbs are much worse than the others because the Gibbs samplings [31] can get stuck in a local minima [32], [67]. Meanwhile, the recovery accuracy curves of Adaptive-MAP-OMP gradually decrease in many cases such as in recovering CMU-IDB images in the wavelet and DCT domains, and CIFAR-10 images in the DCT domain.

In Figure 3.10, the proposed Two-step-Adaptive MRF converges the fastest and requires the least runtime. Conversely, the runtime accumulation of Adaptive-MAP-OMP and Adaptive-Gibbs are extremely high. The runtime of Adaptive-MAP-OMP increases sharply while performing each support estimation. Meanwhile, Adaptive-Gibbs suffers severely from slow convergence. This demonstrates the superior performance of the proposed sparse signal estimation in Two-step-Adaptive MRF. The ending of each ripple does not appear as a sharp vertical drop because they are resulted from averaging over 10 images.

3.6.7 Performance evaluation

In this section, we compare the performance of the proposed Two-step-Adaptive MRF with several state-of-the-art CS methods.

3.6.7.1 Compressibility.

In this section, we evaluate the performance in terms of compressibility by performing sparse signal recovery across different sampling rates (M/N). Figure 3.11 shows the average PNSR curves across different sampling rates on the three datasets, when the noise level (SNR) is 30 dB. The Two-step-Adaptive MRF offers the highest performance in most cases:

On MNIST, the proposed Two-step-Adaptive MRF yields the best performance. The proposed Two-step-Adaptive MRF exceeds the second best method by at least 0.5 dB, when the sampling rate is higher than 0.25. The other structured CS methods such as the MAP-OMP, Pairwise MRF, Bernoulli, and GCoSamp, also offer good performance and outperform the methods that do not employ signal structures such

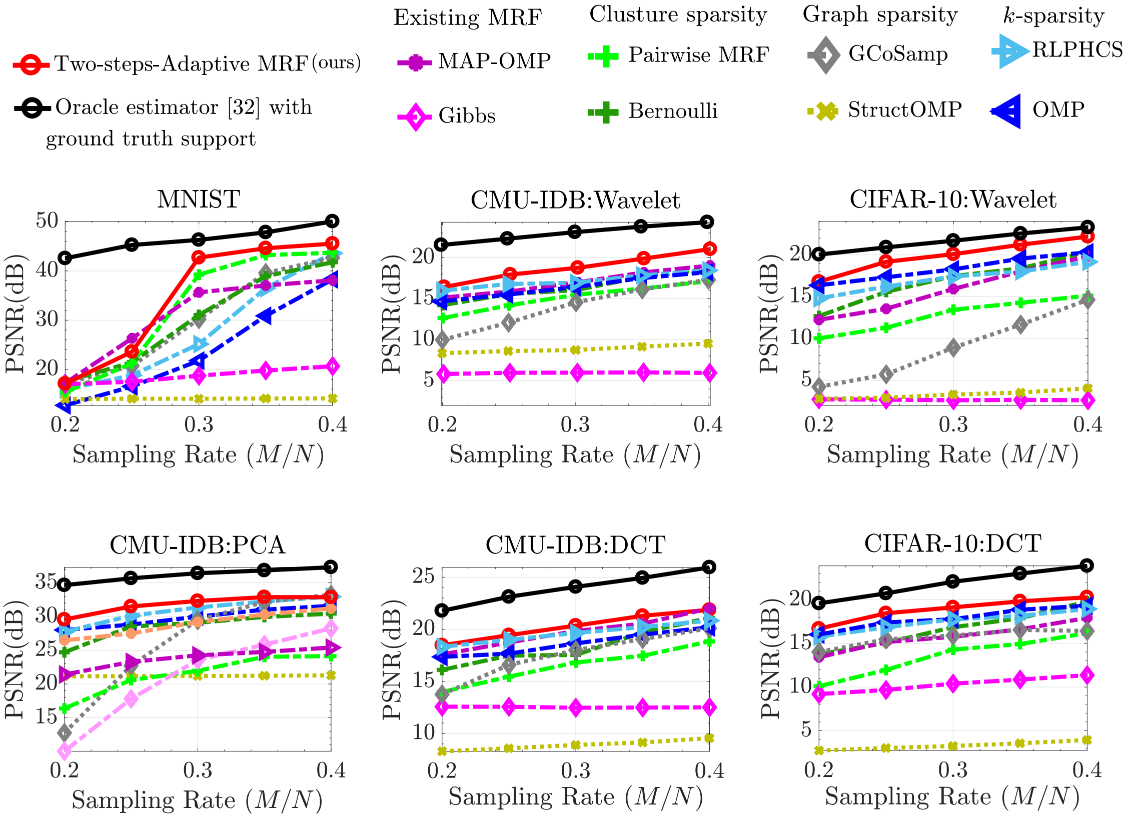


FIGURE 3.11: Compressibility. The PSNR curves across different sampling rates on three datasets: MNIST images; PCA, wavelet, and DCT signals of CMU-IDB images; and wavelet, and DCT signals of CIFAR-10 images. The noise level (SNR) is 30 dB.

as OMP and RLPHCS. This is mainly because the handwritten images of the MNIST dataset contain only lines and strokes which are highly structured and repetitive; thus, the underlying structure can be exploited by many structured CS algorithms.

On CMU-IDB, the proposed Two-step-Adaptive MRF offers the highest performance. When the sampling rate is higher than 0.25, the proposed Two-step-Adaptive MRF exceeds the second best method by at least 1 dB in the wavelet domain and 0.25 in the DCT domain. Meanwhile, for the sparse signal recovery in PCA domain, the proposed Two-step-Adaptive MRF provides comparable result to RLPHCS and GCoSamp that achieve the highest performance, but when the sampling rate is lower than 0.3 (less measurements), the proposed method outperforms the others by at least 0.25 dB. However, the other structured CS methods are only comparable with OMP and RLPHCS in most cases. This could be because the CMU-IDB face images contain more information with higher diversity than the MNIST images. With higher flexibility and adaptability, the proposed Two-step-Adaptive MRF can utilize the underlying structure of these sparse representations more effectively than the other structured CS methods.

On CIFAR-10, most of the structured CS methods, except the Two-step-Adaptive MRF, are beaten by OMP and RLPHCS. When the sampling rate is higher than 0.25, the proposed Two-step-Adaptive MRF exceeds the second best method by at least 1 dB in the wavelet domain and 0.25 dB in the DCT domain. The natural images of CIFAR-10 contain higher information which is less structured and more diverse than the two previous datasets. As the underlying structure of the sparse representation of CIFAR-10 are more challenging, many structured CS methods fail to capture the underlying structure of the sparse representation. With better flexibility and adaptability, the Two-step-Adaptive MRF is able to capture the underlying structure; thus, it outperforms the other structured CS methods.

With higher flexibility and adaptability, the Two-step-Adaptive MRF outperforms the other methods across different datasets. To further demonstrate the superior performance of the proposed Two-step-Adaptive MRF (TA-MRF), we show the visual results of a MNIST handwritten digit image, a CMU-IDB face image, and a CIFAR-10 natural image in Figure 3.12, Figure 3.13, and Figure 3.14, respectively. The Two-step-Adaptive MRF gives rise to the best results, which contain more details and less noise than its competitors. The full visual results are provided in Appendix A.1.

3.6.7.2 Noise tolerance.

To test the noise tolerance performance, we evaluate performance of the Two-step-Adaptive MRF across different noise levels (in SNR). Figure 3.15 provides the average PNSR curves across different noise levels on the three datasets, when the sampling rate is set to 0.3. Because the Two-step-Adaptive MRF employs the flexible and adaptive prior, it outperforms the other methods across different datasets, i.e. MNIST, CMU-IDB, and CIFAR-10, in most cases:

On MNIST, the proposed Two-step-Adaptive MRF outperforms the other methods in most cases. When the SNR is higher than 10 dB, the Two-step-Adaptive MRF

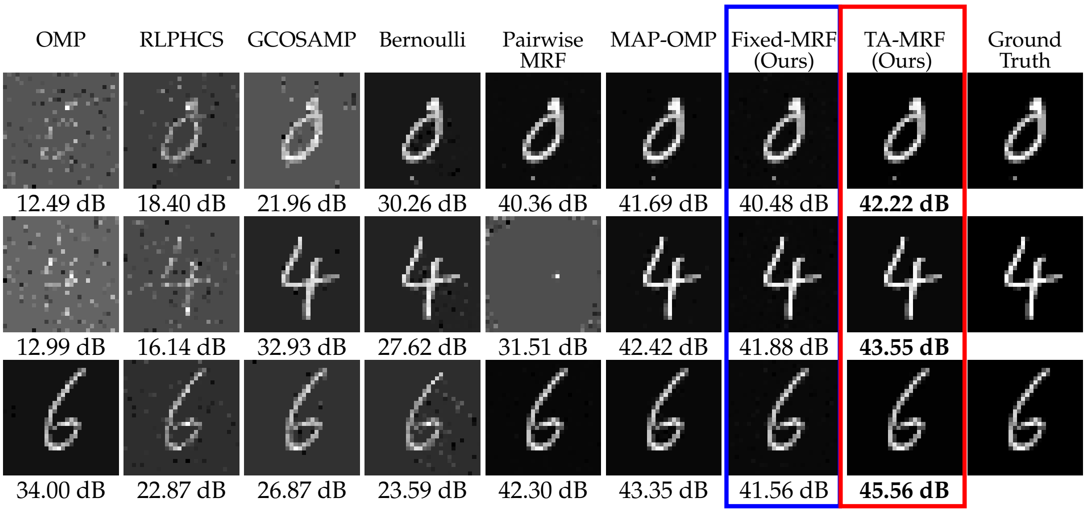


FIGURE 3.12: Visual results of the selected MNIST handwritten digit images by the top eight most competitive methods, i.e. OMP, RLPHCS, GCOSAMP, Bernoulli, Pairwise, MAP-OMP, and the proposed Fixed-MRF and TA-MRF, at $M/N = 0.3$, SNR = 30 dB.

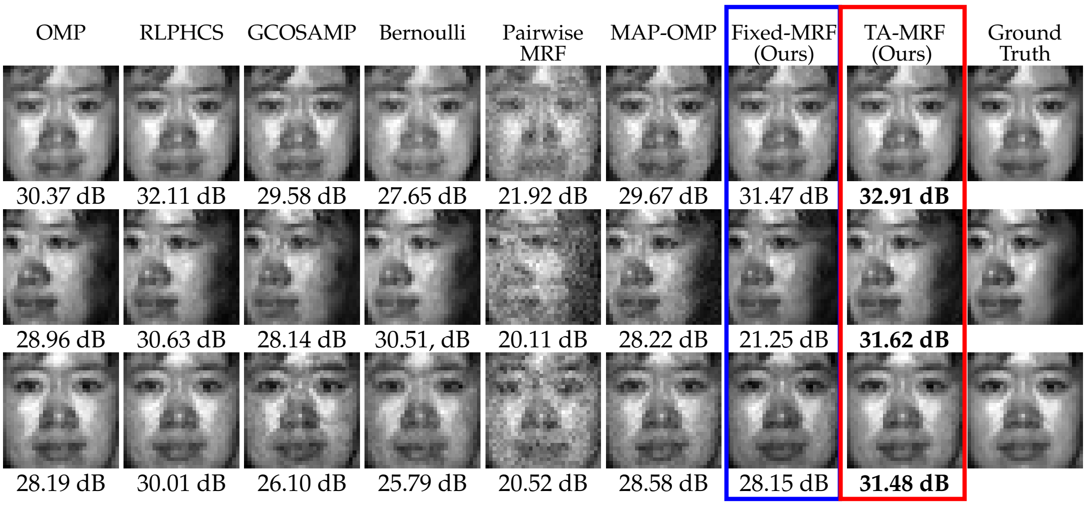


FIGURE 3.13: Visual results of the selected CMU-IDB face images from sparse signal recovery in the PCA domain by the top eight most competitive methods, i.e. OMP, RLPHCS, GCOSAMP, Bernoulli, Pairwise, MAP-OMP, and the proposed Fixed-MRF and TA-MRF, at $M/N = 0.3$, SNR = 30 dB.

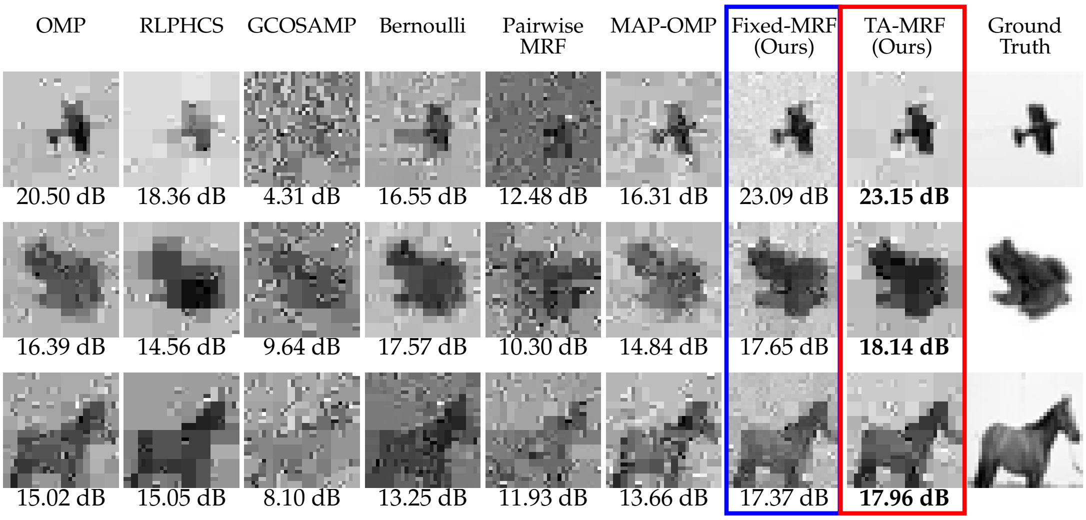


FIGURE 3.14: Visual results of the selected CIFAR-10 natural images from sparse signal recovery in the wavelet domain by the top eight most competitive methods, i.e. OMP, RLPHCS, GCOSAMP, Bernoulli, Pairwise, MAP-OMP, and the proposed Fixed-MRF and TA-MRF, at $M/N = 0.3$, SNR = 30 dB.

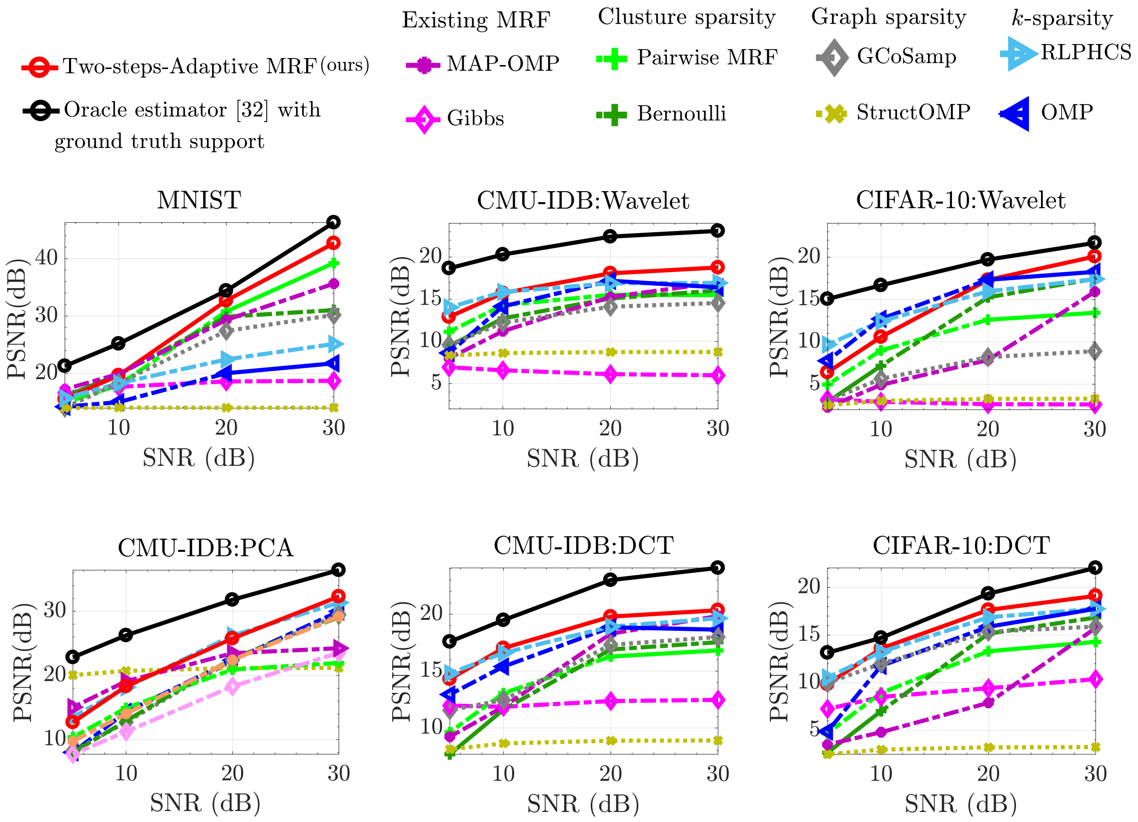


FIGURE 3.15: Noise Tolerance. The PSNR curves across different noise levels (SNR) on three datasets: MNIST images; PCA, wavelet, and DCT signals of CMU-IDB images; and wavelet, and DCT signals of CIFAR-10 images. The sampling rate is 0.3.

outperforms the second best method by at least 2 dB. The other structured CS methods perform well with the MNIST images with the handwritten patterns that are more repetitive and structured than the face images of CMU-IDB and the natural images of CIFAR-10.

On CMU-IDB, the proposed Two-step-Adaptive MRF outperforms the other methods in most cases. It exceeds the second best method by at least 0.25 dB in the wavelet domain, 0.5 dB in the PCA domains, and 0.25 dB in the DCT domain, when the noise is higher than 10 dB. Due to the dense information in the face images of CMU-IDB, the sparse representation is more diverse and less structured. Therefore, the other structured CS methods only perform as well as the non-structured CS approaches, i.e. OMP and RLPHCS.

On CIFAR-10, the proposed Two-step-Adaptive MRF outperforms the other methods in most cases. It exceeds the second best method by 0.25 dB in the DCT domain when the noise is higher than 10 dB. Nevertheless, in the recovery of the sparse representation of CIFAR-10 images in the wavelet domain, Two-step-Adaptive MRF is beaten by OMP and RLPHCS when noise level $\leq$ 20 dB, where it provides the third best performance on the signal recovery in the wavelet domain. Meanwhile, the other structured CS methods perform much worse (at least 2 dB lower than the Two-step-Adaptive MRF when the noise level $\geq$ 10 dB). This could be because the natural images from CIFAR-10 are more diverse and less structured than the two previous datasets. Nevertheless, when the SNR becomes higher ($>$ 20 dB), the measurements contain less noise; thus, the Two-step-Adaptive MRF outperforms RLPHCS and OMP.

Therefore, with higher flexibility and adaptability, the Two-step-Adaptive MRF outperforms the other methods across different noise levels in most cases. Nevertheless, the low noise tolerance in recovering the sparse representation of CIFAR-10 in the wavelet domain indicates the limited performance of the MRF parameter estimation in the Two-step-Adaptive MRF. More details and discussion regarding this problem are provided in Section 3.8. Our investigation indicates that this problem can be caused by the fact that the MRF parameter estimation fails to improve the sparse signal recovery. This is mainly because the MRF parameter estimation relies on the point estimate of sparse signals, which can lead to inaccurate parameter estimation. As a result, the Two-step-Adaptive MRF becomes less competitive than the methods that do not employ signal structures.

3.6.7.3 Runtime performance.

In this section, we study the runtime of the proposed Two-step-Adaptive MRF in comparison with the competitors by observing the runtime performance across different sampling rates (M/N). All the methods were implemented by 64-bit MATLAB R2016b and were executed on a PC with Intel Core i7-4770 CPU and 16GB of RAM. Figure 3.16 provides runtime performance across different sampling rates (M/N) on the three datasets. The noise level (SNR) is 30 dB:

On MINST handwritten images, the average runtime of our Two-step-Adaptive MRF is faster than MAP-OMP, StructOMP, and Gibbs, but slower than structured CS

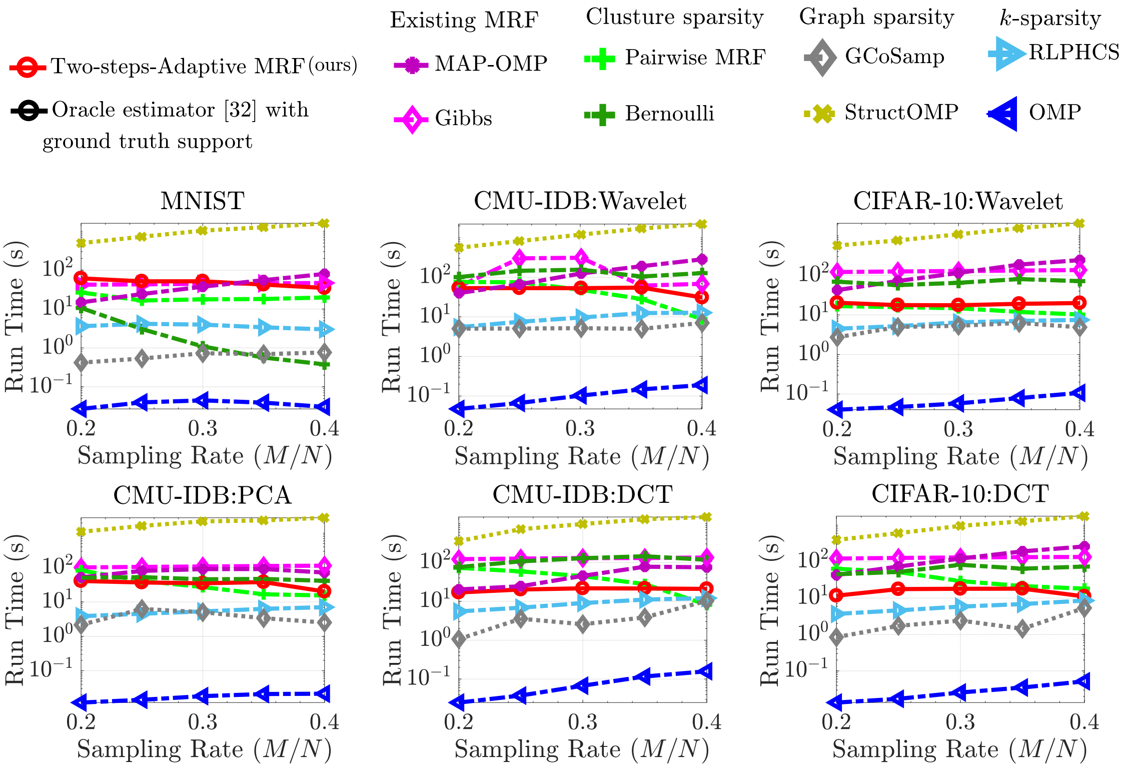


FIGURE 3.16: Runtime performance. Runtime curves across different sampling rates on three datasets: MNIST images; PCA, wavelet, and DCT signals of CMU-IDB images; and wavelet, and DCT signals of CIFAR-10 images. The noise level (SNR) is 30 dB.

approaches— Pairwise MRF, Bernoulli, and GCoSAMP—and the non-structured CS approaches—OMP and RLPHCS.

For CMU-IDB and CIFAR-10 datasets, our runtime performance is much better than many structured CS algorithms. The runtime performance is similar across the wavelet, DCT, and PCA domains, i.e., the proposed Two-step-Adaptive MRF is faster than MAP-OMP, Gibbs, Bernoulli, and StructOMP. The proposed Two-step-Adaptive MRF is comparable to Pairwise MRF and slower than GCOSAMP, OMP, and RLPHCS. Note that OMP and RLPHCS require less computation because they do not exploit the signal structure. GCoSAMP is a fast algorithm, but the accuracy is much lower. Therefore, this demonstrates that the Two-step-Adaptive MRF offers a moderate runtime performance in most cases.

3.6.8 Empirical convergence

In this section, we verify the convergence of the Two-step-Adaptive MRF through the decay of the recovery error percentage (%) with respective to the ground truth.

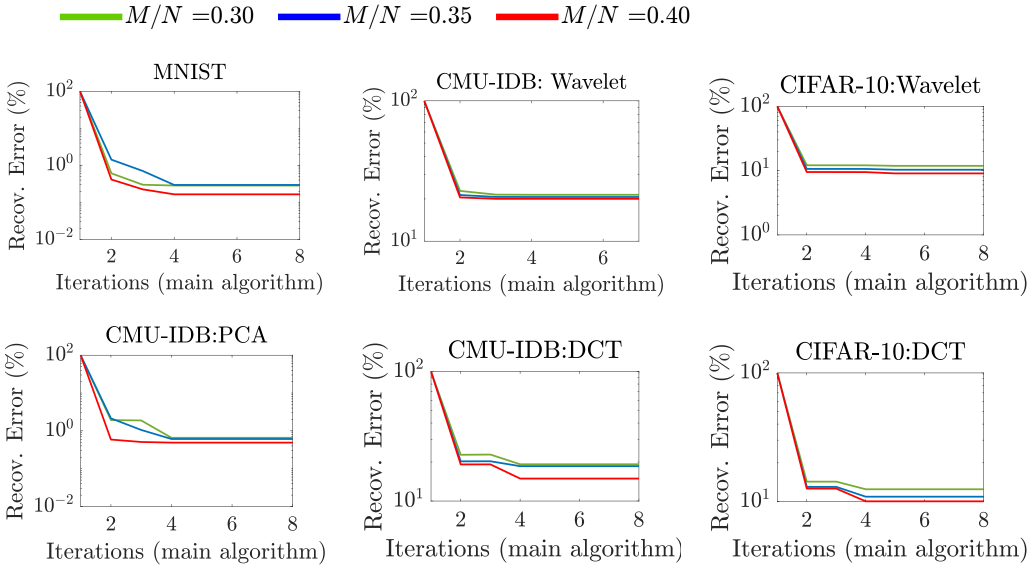
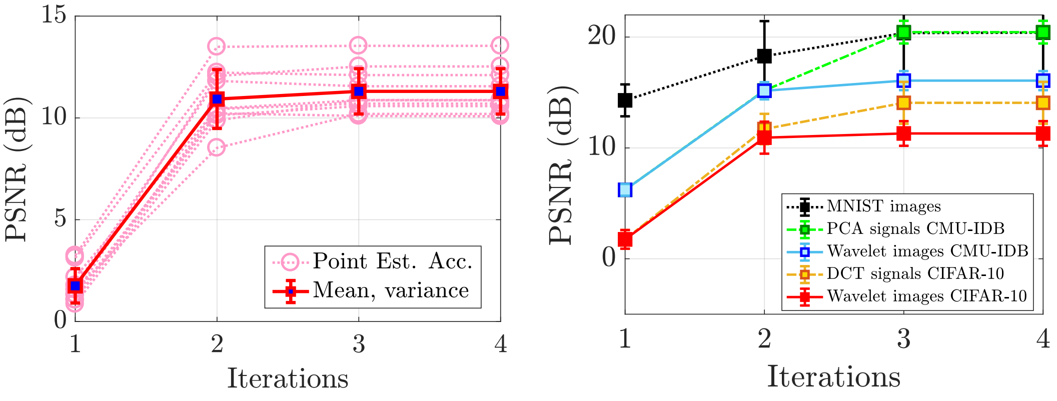


FIGURE 3.17: Convergence of the Two-step-Adaptive MRF in the percentage of the recovery error at the sampling rate and noise level (SNR) of 0.3 and 30 dB on MNIST images; the PCA, wavelet, and DCT signals of CMU-IDB images; and the wavelet, and DCT signals of CIFAR-10 images.

Our Two-step-Adaptive MRF aims to solve Eq. (3.6) by solving two sub-problems, i.e. optimizing Eq. (3.10) and Eq. (3.12). Given a fixed support s, the optimization problem Eq. (3.10) is convex [102], [103]. Given the estimated support, the sub-optimization problem Eq.(3.12) to estimate the MRF parameters is also convex. Thus, the cost function Eq. (3.12) keeps decreasing. Although both the Eq. (3.10) and Eq. (3.12) are convex; they do not necessarily imply the Eq. (3.6). Therefore, to confirm that the algorithm converges, we provide the empirical convergence. The empirical convergence is demonstrated by the recovery error in each iteration in Figure 3.17. The noise level is 30 dB. We can see that in most cases, Adaptive-MRF converges after iterate for 3 times.

3.7 Conclusion

We propose a new adaptive MRF-based CS method with the flexibility to capture and adapt for any signal structures. To flexibly capture different signal structures, a full Boltzmann machine is employed to model the signal distribution. To realize an adaptive MRF, the MRF parameters (both the BM parameters and underlying graph) are adaptively estimated based on the intermediate estimation of the sparse signals. To maximize adaptability, a new sparse signal estimation is proposed to jointly estimate



(A) Point estimation accuracy

(B) Comparison with the other image sets

FIGURE 3.18: Examining the cause of low performance in the signal recovery of the wavelet images (CIFAR-10) when the noise level is 10 dB. Sampling rate is 0.3: (A) the point estimation accuracy on the wavelet images, and (B) the estimation accuracy comparison with the other image sets. The low performance is caused by the majority of point estimates failing to improve after the 2^{nd} iteration.

the sparse signal, support, signal and noise parameters. Extensive experiments on the three real-world datasets demonstrates the promising performance of the proposed method.

3.8 Discussion

We have demonstrated the performance of the Two-step-Adaptive MRF with three different experiments. The Two-step-Adaptive MRF provides good performance in many experiments. Nevertheless, we also notice the problem of the low noise tolerance of the proposed Two-step-Adaptive MRF on the recovery of sparse representation of CIFAR-10 images in the wavelet domain with moderate to high noise corruption (noise level in SNR $<$ 15 dB) in Figure 3.15.

To examine this problem, Figure 3.18A provides the point estimation performance on ten CIFAR-10 signals (images) in the wavelet domain. Only a few point estimates improve after the second iteration. This indicates that the majority of the adapted MRF parameters do not improve the sparse signal estimation. Figure 3.18B compares the performance on CIFAR-10 images in the wavelet domain with the performance on the other image sets, where the proposed Two-step-Adaptive MRF performs well: MNIST images, CMU-IDB images in the PCA domain, CMU-IDB images in the

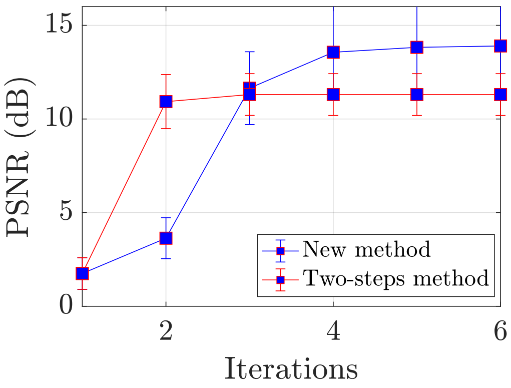


FIGURE 3.19: Comparison of the recovery improvement by the new method and the Two-step-Adaptive MRF. Noise level is 10 dB. Sampling rate is 0.3.

wavelet domain, and CIFAR-10 images in the DCT domain. The overall performance on CIFAR-10 is lower than the other two datasets, which could be improved with the adapted MRF parameters in the Two-step-Adaptive MRF. The MRF parameter estimation depends on the point estimation of the latent sparse signals. However, the majority of the point estimates do not achieve high PSNR on recovering the sparse representation of CIFAR-10 images in wavelet domain, which leads to inaccurate parameter estimation. These point estimates do not necessarily represent the latent sparse signals well. As a result, the Two-step-Adaptive MRF has limited performance because of how the MRF parameters are estimated.

To address this problem, we reformulate the MRF parameter estimation into a maximum marginal likelihood problem in Chapter 4 that estimates the MRF parameters directly from the measurements to better depict the statistical uncertainty of the latent sparse signals. Figure 3.19 compares the signal estimation improvement using the new method and the Two-step-Adaptive MRF. The new method can further improve the overall performance by at least 3 dB from the Two-step-Adaptive MRF.

Chapter 4

One-step Adaptive MRF for Structured CS

4.1 Introduction

Previously, we have proposed an adaptive Markov random field (MRF) and developed the Two-step-Adaptive MRF that can adaptively estimate both the parameters and the underlying graph of the MRF. A full Boltzmann machine (BM) with both pairwise and unary potentials is employed to model signal distribution. Consequently, adaptive MRF has a higher flexibility and adaptability to capture and adapt to any signal structures, compared with all the previous structured sparsity models [29]–[33], [35], [36], [38]–[42]. To adaptively estimate an MRF for a signal structure, this method employs two major estimation steps—i) sparse signal estimation, and ii) based on the resulting sparse signal, the MRF parameters estimation which includes the *BM parameters* and the *underlying graph* of MRF estimations. However, Two-step-Adaptive MRF has two main problems:

- The estimated MRF parameters do not always capture the underlying structure of the entire signal population: the MRF parameter estimation is based on the point estimation of the latent sparse signal. The point estimate cannot depict the statistical uncertainty of the latent sparse signals.
- High computational cost: the Two-step-Adaptive MRF performs the two estimation steps, MRF estimation and signal estimation iteratively, until convergence. Thus, the total cumulative computational cost is high.

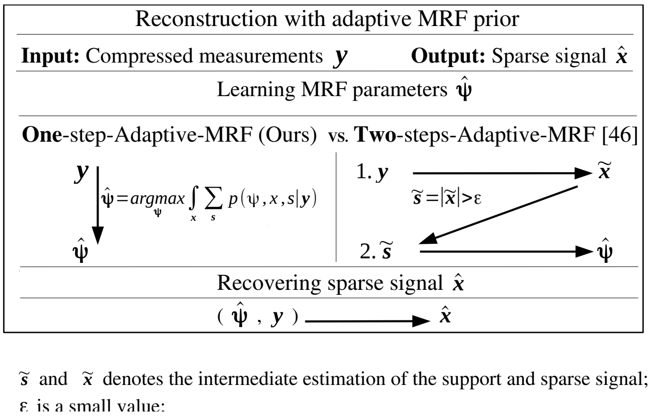
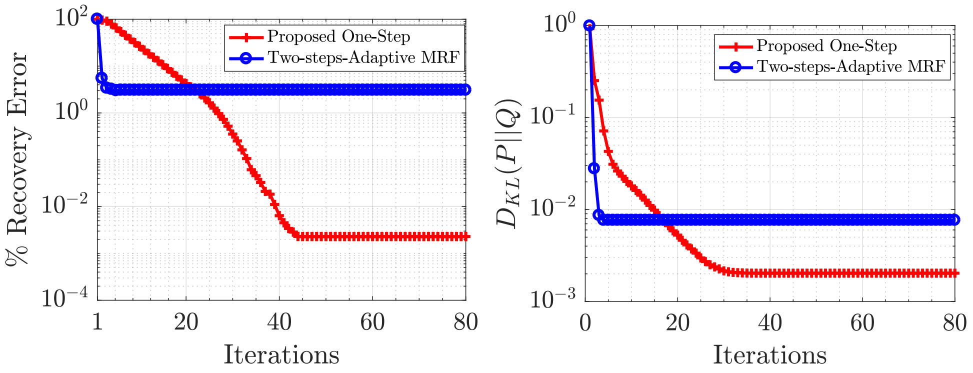

Reconstruction with adaptive MRF prior
Input: Compressed measurements $\boldsymbol{y}$
Output: Sparse signal $\hat{\boldsymbol{x}}$
Learning MRF parameters $\hat{\boldsymbol{\psi}}$
One-step-Adaptive-MRF (Ours) vs. **Two**-steps-Adaptive-MRF [46]
$\boldsymbol{y}$
$\hat{\psi} = \underset{\psi}{argmax} \int_x \sum_s p(\psi, x, s|\boldsymbol{y})$
$\hat{\boldsymbol{\psi}}$
1. $\boldsymbol{y}$
$\tilde{\boldsymbol{x}}$
$\tilde{s} = |\tilde{x}| > \varepsilon$
2. $\tilde{\boldsymbol{s}}$
$\hat{\boldsymbol{\psi}}$
Recovering sparse signal $\hat{\boldsymbol{x}}$
$(\hat{\boldsymbol{\psi}}, \boldsymbol{y})$
$\hat{\boldsymbol{x}}$
$\tilde{s}$ and $\tilde{x}$ denotes the intermediate estimation of the support and sparse signal;
ε is a small value;

FIGURE 4.1: Comparison between the two frameworks. Our One-step-Adaptive MRF estimates the parameters from measurements based on Bayesian estimation directly, while the Two-step-Adaptive MRF [45] estimates the parameters based on the estimation of sparse signal.

To address these problems, we propose to take a Bayesian approach to provide a better generalization over the latent sparse signals. This process is shown in comparison with the Two-step-Adaptive MRF in Figure 4.1. Instead of finding a point estimate for a sparse signal, the proposed approach captures the statistical uncertainty by considering the marginal likelihood for the model parameters given the measurements. The marginal likelihood is obtained by integrating out all the unknown variables, which can be seen as *weighted averaging* with the probability of each variation of sparse signal from the entire population. Thus, this offers better generalization to the underlying structure of the sparse signals population. As the latent sparse signals are integrated out, the MRF parameters are estimated directly from the measurements in one step. Thus, the proposed method is referred to as *One-step-Adaptive MRF*.

To implement this, we first approximate the BM with a new MRF distribution which is the product of two simpler priors, i.e., the Bernoulli model [39] and the pairwise MRF [42] to enable a closed-form update for MRF parameter estimation. The Bernoulli model represents the bias toward zero for each signal coefficient, while the pairwise MRF represents the correlation between these coefficients. Then, the parameters of the new MRF distribution are estimated directly from the measurements by solving a maximum marginal likelihood (MML) problem. More importantly, the estimation of all the unknown variables resulting from the MML problem gains closed-form updates with low computational cost.



(A) Signal recovery improvement

(B) Learning improvement

FIGURE 4.2: Performance comparison between One-step-Adaptive MRF (proposed) and Two-step-Adaptive MRF in (A) signal recovery and (B) MRF parameter estimation improvement (measured by the KL-divergence of the estimated MRF with respect to the ground truth distribution). Our One-step approach is able to minimize recovery errors and KL-divergence further.

Figure 4.2 compares the effectiveness of the proposed One-step-Adaptive MRF versus the Two-step-Adaptive MRF [1]in signal recovery and the MRF parameter estimation on 1000 synthesized sparse signals sampled from a known distribution. The accuracy of MRF parameter estimation is measured by the KL-divergence with respect to the ground truth. As the Two-step-Adaptive MRF estimates the MRF parameters based on the point estimation of sparse signals, it often converges too early, thus, limits the ultimate recovery accuracy. On the contrary, the proposed One-step-Adaptive MRF can minimize the recovery error and KL-divergence further due to its better generalization. Extensive experiments demonstrate the superior performance of the proposed One-step-Adaptive MRF. In summary, this chapter makes the following contributions:

1. We propose a new MRF distribution that approximates the Boltzmann machine (BM) of MRFs to enable closed-form updates for the MRF parameters with a low computational cost. To achieve this, the proposed MRF distribution is the product between a *Bernoulli* model and a *pairwise MRF*. It offers the best approximation to the BM as compared with using the Bernoulli model [39] or the pairwise MRF [42] alone (see Section 4.5.5).

[1] Here, the recovery accuracy and KL-divergence of Two-step-Adaptive is measured at the main algorithm, rather than at the subroutine (signal estimation).

2. With the proposed MRF distribution, we propose One-step-adaptive MRF to better generalize the sparse signal population, by solving the maximum marginal likelihood (MML) problem to obtain the MRF parameters from given measurements. We employ a variational expectation maximization (EM) [116] to efficiently solve the MML problem. Thus, we improve (i) the generalization in MRF estimation and (ii) the runtime as the estimation for all the unknown variables gains closed-form updates (see Section 4.5.6).

3. We demonstrate state-of-the-art recovery performance on three benchmark datasets: i) MNIST, ii) CMU-IDB, and iii) CIFAR-10 images in terms of recovery accuracy, noise tolerance, and runtime performance (see Section 4.5).

In the following, we provide the observation model for the MRF based structured CS in Section 4.2. Then, we discuss how signal structure is modelled with a general MRF, and present the proposed MRF distribution in Section 4.2.1. Subsequently, we show how to infer the MRF parameters from compressed measurements based on a Bayesian estimation approach, where the inference is done by applying a variational EM [116] (see Section 4.3). Details about the optimization for each unknown is provided in Section 4.3.2. To this end, the algorithm complexity of the proposed One-step-Adaptive MRF is presented and compared with that of Two-step-Adaptive MRF in Section 4.4. Experimental results to demonstrate the performance of the proposed One-step-Adaptive MRF are provided in Section 4.5.

4.2 Graphical compressive sensing

Inspired by [39], we decompose the sparse signal $\boldsymbol{x} \in \mathcal{R}^N$ into a support vector $\boldsymbol{s} \in \{0,1\}^N$ with a scale vector $\boldsymbol{t} \in \mathcal{R}^N$, which can be denoted as $\boldsymbol{x} = \boldsymbol{t} \odot \boldsymbol{s}$. The support vector $\boldsymbol{s}$ indicates the position of non-zero coefficients in the sparse signal $\boldsymbol{x}$. Thus, our goal is to recover $\boldsymbol{t}$ and $\boldsymbol{s}$ from the following linear observation model

$$\boldsymbol{y} = \boldsymbol{A}(\boldsymbol{t} \odot \boldsymbol{s}) + \boldsymbol{n}, \tag{4.1}$$

where $\boldsymbol{A} \in \mathcal{R}^{M \times N}$ is the measurement matrix, and the measurements $\boldsymbol{y}$ is corrupted by additive Gaussian white noise $\boldsymbol{n}$ with the noise precision σ_n^{-1}. Thus, the corresponding observation likelihood can be formulated as

$$p(\boldsymbol{y}|\boldsymbol{t}, \boldsymbol{s}; \sigma_n) = \mathcal{N}(\boldsymbol{A}(\boldsymbol{t} \odot \boldsymbol{s}), \sigma_n^{-1} \boldsymbol{I}). \tag{4.2}$$

where $\boldsymbol{I}$ is an identity matrix with proper size. Generally, given some appropriate prior models, e.g., $p(\boldsymbol{t})$ and $p(\boldsymbol{s})$, the latent $\boldsymbol{t}$ and $\boldsymbol{s}$ can be inferred by solving the following MAP problem

$$\{\hat{\boldsymbol{t}}, \hat{\boldsymbol{s}}\} = \max_{\boldsymbol{t}, \boldsymbol{s}} p(\boldsymbol{t}, \boldsymbol{s}|\boldsymbol{y}) \propto (\boldsymbol{y}|\boldsymbol{t}, \boldsymbol{s}) p(\boldsymbol{t}) p(\boldsymbol{s}). \tag{4.3}$$

In the following sections, we will discuss the prior models $p(\boldsymbol{t})$ and $p(\boldsymbol{s})$, respectively.

4.2.1 Markov random field based support prior

Since MRFs are flexible and expressive enough to model complex dependency, the majority of the existing works [29]–[33], [35], [36] employ the MRF to capture the underlying structure of a sparse representation through its support $\boldsymbol{s}$. The MRF represents the dependency between the support coefficients by defining the probability distribution over an undirected graph. Let $\mathcal{G} = \{V, E\}$ denotes the underlying undirected graph of the MRF, where V and E are the set of nodes and undirected edges in $\mathcal{G}$. Each coefficient is mapped one-to-one to a node in the graph $\mathcal{G}$. The probability distribution is defined as a Boltzmann machine (BM):

$$p(\boldsymbol{s}) = \frac{1}{Z} \prod_{c} \prod_{i \in \mathbb{N}_c} \exp(s_i \delta_i^c + s_i \sum_{j \in \mathcal{E}_i} \gamma_{ij}^c s_j) \tag{4.4}$$

where $Z(\cdot)$ is a normalizing constant; $\{\delta_i^c, \gamma_{ij}^c\}$ are local parameters that model the interaction among signal coefficients. δ_i^c defines bias toward zero potential (e.g., confidence) for each s_i and γ_{ij}^c weights the dependency between s_i and its adjacent s_j $\forall j \in \mathcal{E}_i$ defined by the local edge set $\mathcal{E}_i$, where the edge set $E = \{\mathcal{E}_i\}_{i \in V}$. The neighborhood set $\mathbb{N}_c$ defines how these parameters are shared among the support coefficients.

An important key for applying the MRFs is to estimate the parameters $\{\delta_i^c, \gamma_{ij}^c\}$ in Eq. (4.4). Generally, these model parameters are learned from the training data. However, the learned model cannot adapt for new signal structures. Meanwhile, the Two-step-Adaptive MRF estimates these parameters based on a point estimate of the sparse signal, which is required to perform both the parameter estimation and sparse signal estimation in every iteration. However, this can lead to high computation.

To address this problem, we propose to approximate the BM Eq.(4.4) with a new probability distribution. Inspired by [117], we assume conditional independence between each node, given its adjacent nodes. Thus, the joint distribution is written as the product of conditional probabilities. Then, we approximate each conditional probability distribution with the product of two simpler distributions. Each of them corresponds to the unary and pairwise potentials in the BM distribution. The proposed MRF distribution for support $\boldsymbol{s}$ is given as

$$p(\boldsymbol{s}) = \prod_c \prod_{i \in \mathbb{N}_c} p(s_i | \boldsymbol{s}_{\mathcal{E}_i}, \theta_i^c) \tag{4.5}$$

$$\text{where} \quad \log(p(s_i|\boldsymbol{s}_{\mathcal{E}_i}, \theta_i^c)) \propto \phi_u(s_i|\theta_i^u) + \phi_p(s_i|\boldsymbol{s}_{\mathcal{E}_i}\theta_i^p)),$$

and $\phi_u(s_i|\theta_i^u) = \log(p_u(s_i|\theta_i^u)), \phi_p(s_i|\boldsymbol{s}_{\mathcal{E}_i}, \theta_i^p) = \log(p_p(s_i|\boldsymbol{s}_{\mathcal{E}_i}; \theta_i^p))$.
Here, $p(s_i|\boldsymbol{s}_{\mathcal{E}_i}, \theta_i^c)$ is the conditional distribution of a support s_i given $\boldsymbol{s}_{\mathcal{E}_i}$ where $\boldsymbol{s}_{\mathcal{E}_i} = [s_j]_{j \in \mathcal{E}_i}$ contains the support coefficients connected to the node s_i with the edges specified by $\mathcal{E}_i$. Then, it is approximated with the product of $p_u(s_i|\theta_i^u)$ and $p_p(s_i|\boldsymbol{s}_{\mathcal{E}_i}, \theta_i^p)$ which are associated with the unary $\phi_u(\cdot)$ and pairwise $\phi_p(\cdot)$ potentials. The superscript u and p denote the parameters/distributions belonging to the unary and pairwise potentials. In the following, we will introduce the specific forms of $p_u(s_i|\theta_i^u)$ and $p_p(s_i|\boldsymbol{s}_{\mathcal{E}_i}, \theta_i^p)$.

Unary potential. To control local sparsity in a fixed-size neighboring region, we employ the Bernoulli model [39] where every support coefficient in the neighboring region shares a common parameter b_c, i.e., $\forall i \in \mathbb{N}_c$

$$p_u(s_i|b_i) = \text{Bernoulli}(s_i|b_i) \quad \text{with} \quad b_i = b_c \sim \text{Beta}(\alpha, \beta). \tag{4.6}$$

b_c defines the tendency toward non-zero according to the setting of α and β. The neighborhood set $\mathbb{N}_c$ defines support coefficients that share the same unary parameters b_c. Therefore, the distribution $p_u(s_i)$ alone only reflects the bias toward zero on support coefficients within a neighborhood but cannot reflect the interaction that the support coefficients have towards each other.

Pairwise potential. To reflect the interaction between the support coefficients, we employ the pairwise MRF [42], where the connection between the i^{th} support coefficient and the other coefficients is defined by $\mathcal{E}_i$. The pairwise MRF is defined as

$$p_p(s_i|\boldsymbol{s}_{\mathcal{E}_i}, w_i) = \frac{1}{Q(w_i, \boldsymbol{s}_{\mathcal{E}_i})} \exp(s_i w_i \sum_{j \in \mathcal{E}_i} s_j), \tag{4.7}$$

where w_i weights the dependency between s_i and other non-zero coefficients defined by $\mathcal{E}_i$. The edge set $E = \{\mathcal{E}_i\}$ defines a whole pairwise connection between nodes in the underlying graph $\mathcal{G}$. Here, the normalizing constant $Q(\cdot)$ is in a closed-form formulation, i.e., $Q(w_i, \boldsymbol{s}_{\mathcal{E}_i}) = 2\cosh(w_i \sum_{j \in \mathcal{E}_i} s_j)$.

With the defined probability distributions associated with the unary and pairwise potentials, we represent the proposed MRF distribution of $\boldsymbol{s}$ as

$$p(\boldsymbol{s}|\boldsymbol{b}, \boldsymbol{w}) = \prod_c \prod_{i \in \mathbb{N}_c} p_u(s_i|b_c) p_p(s_i|\boldsymbol{s}_{\mathcal{E}_i}, w_i) = \prod_c p(\boldsymbol{s}_{\mathbb{N}_c}|b_c, \boldsymbol{w}_{\mathbb{N}_c}), \tag{4.8}$$

where $\forall i \in \mathbb{N}_c$

$$\begin{aligned} p_u(s_i|b_i) &= \text{Bernoulli}(s_i|b_i) \quad \text{with} \quad b_i = b_c \sim \text{Beta}(\alpha, \beta); \\ p_p(s_i|\boldsymbol{s}_{\mathcal{E}_i}, w_i) &= \frac{1}{Z(w_i, \boldsymbol{s}_{\mathcal{E}_i})} \exp(s_i w_i \sum_{j \in \mathcal{E}_i} s_j). \end{aligned}$$

$\boldsymbol{s}_{\mathbb{N}_c} = [s_i]_{i \in \mathbb{N}_c}$ and $\boldsymbol{w}_{\mathbb{N}_c} = [w_i]_{i \in \mathbb{N}_c}$ represent the vector of support coefficients and pairwise parameters in $\mathbb{N}_c$.

Because the distributions associated with the unary and pairwise potentials are separately modelled in Eq. (4.8), their parameters can be separately estimated. This benefits simplifying the following MRF parameter estimation using a variational EM in Section 4.3; The parameters of the Bernoulli model obtain a closed-form solution in inference, and the parameters of a pairwise MRF are obtained by solving an MML problem, which also results in a closed-form formulation. More details will be further clarified in Section 4.3.

The proposed MRF distribution Eq. (4.8) can be viewed as a surrogate for the BM Eq. (4.4) where $\delta_i^c = \delta_c \; \forall i \in \mathbb{N}_c$ and $\gamma_{ij}^c = \gamma_i \quad \forall j \in \mathcal{E}_i$. The effectiveness of the proposed MRF distribution in approximating the BM is measured by the Kullback-Leibler (KL) divergence between them. This result is compared with that of some existing approximation schemes [39], [42]. The results are provided in Section 4.5.5. The KL-divergence of the proposed MRF distribution is smaller than that of other existing schemes. Thus, the proposed MRF distribution Eq. (4.8) can well approximate the BM.

4.2.2 The signal scale prior

In connection with the support model, we impose statistical models onto the signal scale coefficients in each neighborhood site. Specifically, let $\boldsymbol{t}_{\mathbb{N}_c} = [t_i]_{i \in \mathbb{N}_c}$ be a vector of scale coefficients in $\mathbb{N}_c$. We impose an iid Gaussian distribution as a prior of the scale coefficients $\boldsymbol{t}_{\mathbb{N}_c}$. Gamma distribution is used as a hyperprior over the Gaussian variance σ_{ti}:

$$p(\boldsymbol{t}_{\mathbb{N}_c}; {\sigma_t}^c) = \prod_{i \in \mathbb{N}_c} \mathcal{N}(t_i | 0, {\sigma_{t_i}}^{-1} \boldsymbol{I}) \quad \text{with} \quad \sigma_{ti} = {\sigma_t}^c \sim \text{Gamma}(\varpi, \xi) \quad \forall i \in \mathbb{N}_c. \tag{4.9}$$

${\sigma_t}^c$ is the signal precision shared among the scale coefficients in $\mathbb{N}_c$, and $\boldsymbol{I}$ is an identity matrix with a proper size. ϖ and ξ are constant with appropriate settings [42], [118]. This model weakly imposes structure among the scale coefficients in $\mathbb{N}_c$ to help control the sparsity level, in addition to the bias toward zero from the unary term.

4.2.3 The hyperprior for noise precision

As we assume that the small perturbation to the measurements $\boldsymbol{n}$ is Gaussian white noise, the Gamma prior is imposed on σ_n to facilitate the inference of noise precision σ_n.

$$\sigma_n \sim \text{Gamma}(\varpi_0, \xi_0). \tag{4.10}$$

4.3 One-step-Adaptive MRF

With the hyperpriors $p(b_c;\alpha,\beta)$, $p(\sigma_t{}^c;\varpi,\xi)$, and $p(\sigma_n;\varpi_0,\xi_0)$, the posterior of the latent sparse signal scale $\boldsymbol{t}$ and support $\boldsymbol{s}$, given the measurements $\boldsymbol{y}$ is defined as

$$\begin{aligned}&p(\boldsymbol{t},\boldsymbol{s}|\boldsymbol{y},\boldsymbol{\Theta})\\&\propto p(\boldsymbol{y}|\boldsymbol{t},\boldsymbol{s},\boldsymbol{\Theta})p(\boldsymbol{t},\boldsymbol{s}|\boldsymbol{\Theta})p(\boldsymbol{\Theta})\\&=p(\boldsymbol{y}|\boldsymbol{t},\boldsymbol{s},\sigma_n)\prod_c p(\boldsymbol{t}_{\mathbb{N}_c},\boldsymbol{s}_{\mathbb{N}_c}|\sigma_t{}^c,b_c,\boldsymbol{w}_c)\prod_c p(\sigma_t{}^c;\varpi,\xi)p(b_c;\alpha,\beta)p(\sigma_n;\varpi_0,\xi_0).\end{aligned} \tag{4.11}$$

where $p(\boldsymbol{t}_{\mathbb{N}_c},\boldsymbol{s}_{\mathbb{N}_c}|\sigma_t{}^c,b_c,\boldsymbol{w}_c)=p(\boldsymbol{t}_{\mathbb{N}_c}|\sigma_t{}^c)p(\boldsymbol{s}_{\mathbb{N}_c}|b_c,\boldsymbol{w}_c)$ and $\boldsymbol{\Theta}=\{\sigma_n,\boldsymbol{\sigma}_t,\boldsymbol{b},\boldsymbol{w}\}$; $\boldsymbol{\sigma}_t=[\sigma_t{}^c]$, $\boldsymbol{b}=[b_c]$, $\boldsymbol{w}=[\boldsymbol{w}_{\mathbb{N}_c}]$. Most existing MRF-based methods [29]–[33], [35], [36] estimate the model parameters $\boldsymbol{\Theta}$ with training samples. However, the resulting $\boldsymbol{\Theta}$ cannot adapt for actual sparse signals. The two-step method in [45] adaptively estimates $\boldsymbol{\Theta}$ based on the point estimation of sparse signals. However, the point estimation cannot capture the statistical uncertainty of the latent sparse signal, which can lead to inaccurate parameter estimation. To address this problem, we estimate $\boldsymbol{\Theta}$ directly from the noisy measurements with a statistical inference process described in the following section. Then, given $\boldsymbol{\Theta}$, the sparse signal is estimated by solving MAP Eq.(4.3).

4.3.1 Model parameter estimation with variational EM

Our objective is to adaptively estimate the unknown parameters $\boldsymbol{\Theta}$ directly from measurements $\boldsymbol{y}$. With the hyperprior imposed on $\sigma_n,\boldsymbol{\sigma}_t$, and $\boldsymbol{b}$, these unknowns can be considered as the unknown random variables; meanwhile, $\boldsymbol{w}$ is the only unknown parameter. Thus, we aim to solve the following maximum marginal likelihood (MML) problem

$$\max_{\boldsymbol{w}}\ln p(\boldsymbol{y}|\boldsymbol{w})\propto\int\ln p(\boldsymbol{y},\boldsymbol{\Lambda}|\boldsymbol{w})\mathrm{d}\boldsymbol{\Lambda}. \tag{4.12}$$

where $\boldsymbol{\Lambda}=\{\boldsymbol{t},\boldsymbol{s},\sigma_n,\boldsymbol{\sigma}_t,\boldsymbol{b}\}$ is the set of all unknown variables. To solve this MML problem, all the unknown variables in $\boldsymbol{\Lambda}$ are to be integrated out. Since calculating the integral in Eq. (4.12) is intractable, we resort to the variational expectation maximization (EM) [116] to estimate the unknown parameters. In the variational

EM [116], the integral problem is addressed by introducing the pseudo probabilities of the unknown variables $q(\mathbf{\Lambda})$. The log likelihood in Eq. (4.12) is reformulated as:

$$\ln p(\boldsymbol{y}; \boldsymbol{w}) = F(q, \boldsymbol{w}) + KL(q||p), \tag{4.13}$$

with

$$F(q, \boldsymbol{w}) = \int q(\mathbf{\Lambda}) \ln \frac{p(\boldsymbol{y}, \mathbf{\Lambda}; \boldsymbol{w})}{q(\mathbf{\Lambda})} \mathrm{d}\mathbf{\Lambda} \tag{4.14}$$

and

$$KL(q||p) = -\int q(\mathbf{\Lambda}) \ln \frac{p(\mathbf{\Lambda}|\boldsymbol{y}; \boldsymbol{w})}{q(\mathbf{\Lambda})} \mathrm{d}\mathbf{\Lambda}, \tag{4.15}$$

where $KL(q||p)$ is the Kullback-Leibler divergence between $p(\boldsymbol{y}|\mathbf{\Lambda}, \boldsymbol{w})$ and $q(\mathbf{\Lambda})$. Since $KL(q||p) \geq 0$, it holds that $F(q, \boldsymbol{w})$ is the lower bound of $\ln p(\boldsymbol{y}|\boldsymbol{w})$. Therefore, we turn to maximize the lower bound $F(q, \boldsymbol{w})$, by iteratively performing [116]:

- **Expectation**: It is assumed that $q(\mathbf{\Lambda})$ has a factorized form, that is, $q(\mathbf{\Lambda}) = q(\sigma_n)q(\boldsymbol{t})q(\boldsymbol{s}) \prod_c q(\sigma_t{}^c) \prod_c q(b_c)$. The optimal distribution of one of the latent variables Λ_p follows [116]:

$$\hat{q}(\Lambda_p) = \langle p(\boldsymbol{y}, \mathbf{\Lambda}; \boldsymbol{w}) \rangle_{q(\mathbf{\Lambda} \backslash \Lambda_p)}, \tag{4.16}$$

- **Maximization**: Given $\hat{q}(\mathbf{\Lambda})$, calculated from the VB-E step, the unknown parameter $\boldsymbol{w}$ is estimated by solving the following optimization problem:

$$\hat{\boldsymbol{w}} = \arg\max_{\boldsymbol{w}} F(\hat{q}(\mathbf{\Lambda}), \boldsymbol{w}), \tag{4.17}$$

where $\langle f(\cdot) \rangle_{\mathbf{\Lambda} \backslash \Lambda_p}$ represents the expectation of $f(\cdot)$ with respect to the distribution $q(\mathbf{\Lambda} \backslash \Lambda_p)$ where $\mathbf{\Lambda} \backslash \Lambda_p$ represents the set $\mathbf{\Lambda}$ without Λ_p.

As a result, each unknown variable in $\mathbf{\Lambda} = \{\boldsymbol{t}, \boldsymbol{s}, \sigma_n, \boldsymbol{\sigma}_t, \boldsymbol{b}\}$ is calculated by approximating the true posterior $p(\boldsymbol{y}, \mathbf{\Lambda}|\boldsymbol{w})$ in Eq. (4.16) (the Expectation step). As $\boldsymbol{t}$ and $\boldsymbol{s}$ are estimated in the Expectation step, there is no needed to solve MAP Eq. (4.3). The updating rules for each parameter in $\boldsymbol{w}$ are calculated by maximizing the lower bound

$F(\boldsymbol{q}, \boldsymbol{w})$ Eq. (4.17) (the Maximization step). Due to the conditional independence assumption, each w_i can be estimated separately.

4.3.2 Optimization

In this part, we give the optimization details for all unknown variables. In the following, the updates from 4.3.2.1 to 4.3.2.5 belong to the expectation step, while the update in 4.3.2.6 is the maximization step. To better fit the specific signal structure, we update the underlying graph which is the edge set $E = \{\mathcal{E}_i\}$ in 4.3.2.7. Here, we can employ the graph update technique from Chapter 3 since it requires low computation.

4.3.2.1 Estimation for sparse signal scale $\boldsymbol{t}$

Given the update parameters and variables (i.e., $\hat{\boldsymbol{\sigma}}_t$, $\hat{\mathbf{s}}$ and $\hat{\sigma}_n$), and according to Eq.(4.16), we obtain the following update equation for estimation of $\boldsymbol{t}$:

$$\hat{q}(\boldsymbol{t}) \propto \langle p(\boldsymbol{y}|\boldsymbol{t}, \boldsymbol{s}; \sigma_n) p(\boldsymbol{t}; \boldsymbol{\sigma}_t) \rangle_{q(\boldsymbol{\Lambda} \setminus \boldsymbol{t})}. \tag{4.18}$$

Substituting the prior of coefficient scale $\boldsymbol{t}$ Eq. (4.9) and the likelihood of the measurements Eq. (4.2) into Eq. (4.18), we obtain a Gaussian distribution $\mathcal{N}(\boldsymbol{u}_t, \boldsymbol{C}_t^{-1})$ with mean $\boldsymbol{u}_t$ and covariance $\boldsymbol{C}_t$:

$$\begin{aligned} \boldsymbol{u}_t &= \hat{\sigma}_n \boldsymbol{C}_t^{-1} \hat{\mathbf{S}} \boldsymbol{A}^T \boldsymbol{y} \\ \boldsymbol{C}_t &= \hat{\boldsymbol{\Sigma}}_t + \hat{\sigma}_n \langle \boldsymbol{S} \boldsymbol{A}^T \boldsymbol{A} \boldsymbol{S} \rangle_{q(\boldsymbol{s})}, \end{aligned} \tag{4.19}$$

where $\boldsymbol{S} = \text{diag}(\boldsymbol{s})$; $\hat{\mathbf{s}}$ is the update value of $\boldsymbol{s}$ from previous iteration; $\langle \boldsymbol{S} \boldsymbol{A}^T \boldsymbol{A} \boldsymbol{S} \rangle_{q(\boldsymbol{s})} = (\boldsymbol{A}^T \boldsymbol{A}) \odot (\hat{\mathbf{s}} \hat{\mathbf{s}}^T + \text{diag}(\hat{\mathbf{s}} \odot (1 - \hat{\mathbf{s}})))$; and $\hat{\boldsymbol{\Sigma}}_t = \text{diag}([\hat{\sigma}_1, ..., \hat{\sigma}_N])$. Therefore, the update for $\boldsymbol{t}$ is as follows:

$$\hat{\boldsymbol{t}} = \boldsymbol{u}_t. \tag{4.20}$$

4.3.2.2 Estimation for signal support s

Given $\hat{\sigma}_t, \hat{\sigma}_n, \hat{\boldsymbol{t}}, \hat{\boldsymbol{w}}, \hat{\mathbf{s}}$, and $\{\hat{\mathcal{E}}_i\}$ from the previous iteration, the log posterior probability of each element of $\boldsymbol{s}$ is given as

$$\hat{q}(s_i) \propto \langle p(\boldsymbol{y}|\boldsymbol{t}, \boldsymbol{s}; \sigma_n) p(\boldsymbol{s}|\boldsymbol{w}, \boldsymbol{b}) \rangle_{q(\boldsymbol{\Lambda} \backslash s_i)}. \tag{4.21}$$

The probability when $s_i = 1$ is given as

$$\begin{aligned} \ln \hat{q}(s_i = 1) \propto -\sigma_n (\boldsymbol{y}^T \boldsymbol{y} + \langle t_i^2 \rangle \boldsymbol{a}_i^T \boldsymbol{a}_i - 2\hat{t}_i \boldsymbol{a}_i^T (\boldsymbol{y} - \sum_{i>j} \boldsymbol{a}_j \hat{t}_j \hat{s}_j))) \\ + w_i \sum_{j \in \hat{\mathcal{E}}_i} \hat{z}_j - \ln(2 \cosh(w_i \sum_{j \in \hat{\mathcal{E}}_i} \hat{z}_j)) + \langle \ln(p_u(s_i = 1)|b_c) \rangle_{q(b_c)}, \end{aligned} \tag{4.22}$$

The probability when $s_i = 0$ is given as

$$\begin{aligned} \ln \hat{q}(s_i = 0) \propto - w_i \sum_{j \in \hat{\mathcal{E}}_i} \hat{z}_j - \ln(2 \cosh(w_i \sum_{j \in \hat{\mathcal{E}}_i} \hat{z}_j)) \\ + \langle \ln(p_u(s_i = 0|b_c)) \rangle_{q(b_c)}, \end{aligned} \tag{4.23}$$

where $\hat{z}_i = 2\hat{s}_i - 1$, and $\langle t_i^2 \rangle \propto \hat{t}_i^2 + \text{var}(\hat{t}_i)$. $\text{var}(\hat{t}_i)$ is the variance of $\hat{t}_i$ which can be obtained from Eq. (4.19), i.e., $\text{var}(\hat{t}_i) = \text{diag}\{\text{inv}(\boldsymbol{C}_t)\}_{i,i}$. The update for $\langle \ln(p_u(s_i = 1)) \rangle_{q(b_c)}$ and $\langle \ln(p_u(s_i = 0)) \rangle_{q(b_c)}$ are given in Eq. (4.27).

The update of s_i is given as its expectation which is as follows:

$$\hat{s}_i = \frac{\hat{q}(s_i = 1)}{\hat{q}(s_i = 1) + \hat{q}(s_i = 0)} \tag{4.24}$$

Then, update $\hat{z}_i = 2\hat{s}_i - 1$ and update $\hat{\boldsymbol{x}} = \hat{\boldsymbol{t}} \odot \hat{\mathbf{s}}$.

4.3.2.3 Estimation for unary potentials $p_u(s_i|b_{j=c})$

For each b_c,

$$\begin{aligned} \hat{q}(b_c) \propto & \langle \prod_j \prod_{i \in \mathbb{N}_j} p_u(s_i|b_j) p(b_j; \alpha, \beta) \rangle_{q(\boldsymbol{\Lambda} \backslash b_{j=c})} \\ \propto & \text{Beta}(\hat{\alpha}, \hat{\beta}), \end{aligned} \tag{4.25}$$

which calculates expectation over all unknown random variables, except every term that involves with b_c. Since Bernoulli and Beta distributions are a conjugate pair, the posterior hyperparameters are given as

$$\begin{aligned}\hat{\alpha} &= \alpha + \sum_{i \in \mathbb{N}_c} \hat{s}_i \\ \hat{\beta} &= \beta + |\mathbb{N}_c| - \sum_{i \in \mathbb{N}_c} \hat{s}_i,\end{aligned} \tag{4.26}$$

Thus, we have

$$\begin{aligned}\langle \ln(p_u(s_i = 1|b_c)) \rangle_{q(b_c)} &= \psi(\hat{\alpha}) - \psi(\hat{\alpha} + \hat{\beta}) \\ \langle \ln(p_u(s_i = 0|b_c)) \rangle_{q(b_c)} &= \psi(\hat{\beta}) - \psi(\hat{\alpha} + \hat{\beta}),\end{aligned} \tag{4.27}$$

where $\psi(x) = (d/dx) \ln \Gamma(x)$.

4.3.2.4 Estimation for sparse signal variance $\sigma_t{}^c$

The estimation for σ_{tc} is obtained as follows:

$$\begin{aligned}q(\sigma_t{}^c) &\propto \langle \prod_j \prod_{i \in \mathbb{N}_j} p(t_i|\sigma_t{}^j) p(\sigma_t{}^j; \varpi, \xi) \rangle_{q(\mathbf{\Lambda} \backslash \sigma_t{}^{j=c})} \\ &\propto \text{Gamma}(\hat{\varpi}, \hat{\xi}).\end{aligned} \tag{4.28}$$

The Gaussian and Gamma distributions are a conjugate pair. The posterior hyperparameters are given as follows:

$$\begin{aligned}\hat{\varpi} &= \varpi + \frac{|\mathbb{N}_c|}{2} \\ \hat{\xi} &= \xi + \frac{\sum_{i \in \mathbb{N}_c} (\hat{t}_i^2 + \text{var}(\hat{t}_i))}{2}.\end{aligned} \tag{4.29}$$

The update for $\sigma_t{}^c$ is therefore: $\forall i \in \mathbb{N}_c$,

$$\hat{\sigma}_{ti} = \hat{\sigma}_t{}^c = \frac{\hat{\varpi}}{\hat{\xi}}. \tag{4.30}$$

Then, $\hat{\mathbf{\Sigma}}_t$ is updated by plugging in the value of its diagonal entries from $\hat{\sigma}_{t1}, ..., \hat{\sigma}_{tN}$.

4.3.2.5 Estimation for noise variance σ_n

Given $\hat{\boldsymbol{t}}$, $\hat{\mathbf{s}}$, the estimation for σ_n is obtained according to Eq. (4.17)

$$\begin{aligned}\hat{q}(\sigma_n) &\propto \langle p(\sigma_n|\varpi_0,\xi_0)p(\boldsymbol{y}|\boldsymbol{t},\boldsymbol{s},\sigma_n)\rangle_{q(\boldsymbol{\Lambda}\setminus\sigma_n)} \\ &\propto \mathrm{Gamma}(\hat{\varpi}_0,\hat{\xi}_0)\end{aligned} \tag{4.31}$$

With the conjugate property, the hyperparameters of the posterior distribution are given as follows:

$$\begin{aligned}\hat{\varpi}_0 &= \varpi_0 + \frac{M}{2} \\ \hat{\xi}_0 &= \xi_0 + \frac{\langle \| \boldsymbol{y} - \boldsymbol{A}(\boldsymbol{t}\odot\boldsymbol{s}) \|^2\rangle_{q(\boldsymbol{t}),q(\boldsymbol{s})}}{2}.\end{aligned} \tag{4.32}$$

Given $\hat{\boldsymbol{t}}$ and $\boldsymbol{s}$, the expectation is as follows:

$$\begin{aligned}&\langle \| \boldsymbol{y} - \boldsymbol{A}(\boldsymbol{t}\odot\boldsymbol{s}) \|^2\rangle_{q(\boldsymbol{t}),q(\boldsymbol{s})} \\ &= \boldsymbol{y}^T\boldsymbol{y} - 2(\hat{\boldsymbol{t}}\odot\hat{\mathbf{s}})^T\boldsymbol{A}^T\boldsymbol{y} + \mathbf{1}^T[\langle\boldsymbol{s}\boldsymbol{s}^T\rangle\odot\langle\boldsymbol{t}\boldsymbol{t}^T\rangle\odot\langle\boldsymbol{A}^T\boldsymbol{A}\rangle]\mathbf{1}\end{aligned} \tag{4.33}$$

where $\langle\boldsymbol{s}\boldsymbol{s}^T\rangle = \hat{\mathbf{s}}\hat{\mathbf{s}}^T + \mathrm{diag}(\hat{\mathbf{s}}\odot(\mathbf{1}-\hat{\mathbf{s}}))$ and $\langle\boldsymbol{t}\boldsymbol{t}^T\rangle = \hat{\boldsymbol{t}}\hat{\boldsymbol{t}}^T + \hat{\boldsymbol{\Sigma}}_t$. The update for σ_n is therefore:

$$\hat{\sigma}_n = \frac{\hat{\varpi}_0}{\hat{\xi}_0}. \tag{4.34}$$

4.3.2.6 Estimation for pairwise parameters w

Give the updated $\hat{\mathbf{z}}$,$\{\hat{\mathcal{E}}_i\}$ the estimation for each w_i is obtained by solving the following problem:

$$\begin{aligned}\hat{w}_i &= \arg\max_{w_i}\langle\ln p_p(z_i;w_i)\rangle_{q(z_i)} \\ &\equiv \hat{z}_i w_i \sum_{j\in\hat{\mathcal{E}}_i}\hat{z}_j \\ &- \ln(\exp(w_i\sum_{j\in\hat{\mathcal{E}}_i}\hat{z}_j) + \exp(-w_i\sum_{j\in\hat{\mathcal{E}}_i}\hat{z}_j)).\end{aligned} \tag{4.35}$$

Take the gradient with respect to w_i, and equate it to zero, the update of w_i is as follows:

$$\hat{w}_i = \frac{1}{2\sum_{j\in\hat{\mathcal{E}}_i} z_j}\ln\left(\frac{1+\hat{z}_i}{1-\hat{z}_i}\right). \tag{4.36}$$

Algorithm 4.9 Edge update procedure $E = \{\mathcal{E}_i\}$

```
Input: Binary vector $\boldsymbol{d}$.
Initialization : $\mathcal{E}_i = \emptyset \quad \forall i = 1, ..., N$, and $\mathcal{E} = \emptyset$ .
  for $i = 1, ..., N$ do
    for each $j \in \mathbb{N}_i$ do
      If $d_j = 1$ and the edge $(j, i)$ is not present, establish the edge $(i, j)$ by
      including the index $j$ of the node $d_j$ to the local edge set $\mathcal{E}_i$:
      $\mathcal{E}_i = \mathcal{E}_i \bigcup j$ .
    end for
    $E = E \bigcup \mathcal{E}_i$.
  end for
Output: The updated edge set $E$.
```

Algorithm 4.10 One-step-Adaptive MRF (OA-MRF).

```
Input: A measurement signal $\boldsymbol{y}$, $\boldsymbol{A}$, $\{\mathcal{E}_i\}_{initialized}$.
Initialization: $\boldsymbol{\Sigma}_t = \boldsymbol{I}_{N\times N}$, $\sigma_n = 1$, $\boldsymbol{s} = 1$ , $\boldsymbol{w} = \boldsymbol{0}$ and $\boldsymbol{t} = \boldsymbol{0}$;
  while A stopping criterion is not satisfied do
    1. Estimate $\hat{\boldsymbol{t}}$ as Eq. (4.20);
    2. Estimate $\hat{\mathbf{s}}$ by Eq. (4.24) ;
    3. Estimate $\hat{\boldsymbol{b}}$ as Eq. (4.27) ;
    4. Estimate $\hat{\sigma_n}$ as Eq. (4.34) ;
    5. Estimate $\hat{\boldsymbol{\sigma}}_t$ as Eq. (4.30);
    6. Estimate $\hat{\boldsymbol{w}}$ as Eq. (4.36);
    7. Update the edge set $\{\mathcal{E}_i\}_{i=1}^{N}$;
  end while
Output: Recovered $\boldsymbol{x} = \boldsymbol{t} \odot \boldsymbol{s}$.
```

4.3.2.7 Edge set update

Inspired by [45], we can update the underlying graph (i.e., edges set) constructed based on the non-zero coefficient of the support $\hat{\mathbf{s}}$. Since $\hat{s}_i$ has a continuous value, the binary support vector for $\hat{\mathbf{s}}$ is obtained by introducing an appropriate threshold over $\hat{\mathbf{s}}$ [37], [39], [45]: Let $\boldsymbol{d}$ be a binary vector indicating non-zero elements in $\hat{\mathbf{s}}$, and $T_{\hat{\mathbf{s}}} = \frac{\sum_i^N abs(\hat{s}_i)}{N}$ be the mean value of $\hat{\mathbf{s}}$; Specifically, we update $\boldsymbol{d}$ as follows:

$$\hat{d}_i = \begin{cases} 1, & \text{if} \quad \text{abs}(\hat{s}_i) > T_{\hat{\mathbf{s}}} \\ 0, & \text{otherwise.} \end{cases} \tag{4.37}$$

Given the binary support vector $\boldsymbol{d}$, each of the binary coefficients is mapped to each node in a graph $\mathcal{G}$, and each edge is established from one node to other non-zero nodes within a predefined neighborhood $\mathbb{N}_i$. The update procedure is summarized in Algorithm 4.9.

How to solve Eq. (4.12) is summarized in Algorithm 4.10, where the update equations to calculate the expectation for the latent variables $\boldsymbol{t}, \boldsymbol{s}, \sigma_n, \boldsymbol{\sigma_t}$, and $\boldsymbol{b}$ in the Expectation Step are in 4.3.2.1 to 4.3.2.5 , and the update equation to solve a maximization problem for the unknown parameter $\boldsymbol{w}$ in the Maximization Step is in 4.3.2.6. Finally, the edge set E that constitutes a whole pairwise connection in the graph $\mathcal{G}$ is updated. These update rules are performed iteratively until convergence. In most cases, the convergence of the variational EM algorithm is guaranteed [116].

4.4 Algorithm Complexity

All the update steps in Algorithm 4.10 are in closed-form solutions, where most require matrix-vector product operations. The matrix inversion in Step 1 (Eq. (4.19)) in Algorithm 4.10 has the dominant computational cost. The total computational complexity is $\mathcal{O}(N^3 + 2MN^2 + 11N^2 + 5MN + (9 + 4\mathbb{N})N + M + k_0 + k_1)$ which can be reduced to $\mathcal{O}(M^3 + 3MN^2 + 2M^2N + 7N^2 + MN + 4N)$.

The computational complexity of $\mathcal{O}(N^3 + 2MN^2 + 11N^2 + 5MN + (9 + 4\mathbb{N})N + M + k_0 + k_1)$ consists of:

1. $\mathcal{O}(N^3)$ is associated with the matrix inversion that is performed to update the value of $\boldsymbol{C}_t$ of size $N \times N$ in the signal scale estimation Eq. (4.19);

2. The matrix-vector production: (1) the estimation for the sparse signal scale Eq. (4.19) that calculates $\boldsymbol{\mu}_t$ and $\boldsymbol{C}_t$ which requires $\mathcal{O}(N^2 + MN + 2N)$ and $\mathcal{O}(MN^2 + 3N^2 + 2N)$; (2) the estimation of the support Eq. (4.24) that requires $\mathcal{O}(N^2 + 3MN + 4N\mathbb{N} + M)$; (3) the estimation for the sparse signal variance Eq. (4.30) which requires $\mathcal{O}(3N)$; and (4) the estimation for the noise variance Eq. (4.33) which requires $\mathcal{O}(MN^2 + 6N^2 + MN + M + 2N)$;

3. The rest $\mathcal{O}(k_0)$ is from vector product operations, and $\mathcal{O}(k_1)$ is from updating the graph, both of which are linear in N, which is the size of the sparse signal vector.

The computation of the matrix inversion dominates the other computational costs. The computation for matrix inversion can be reduced, however, with the trade-off of performing more vector-matrix multiplications, which will be discussed as follows:

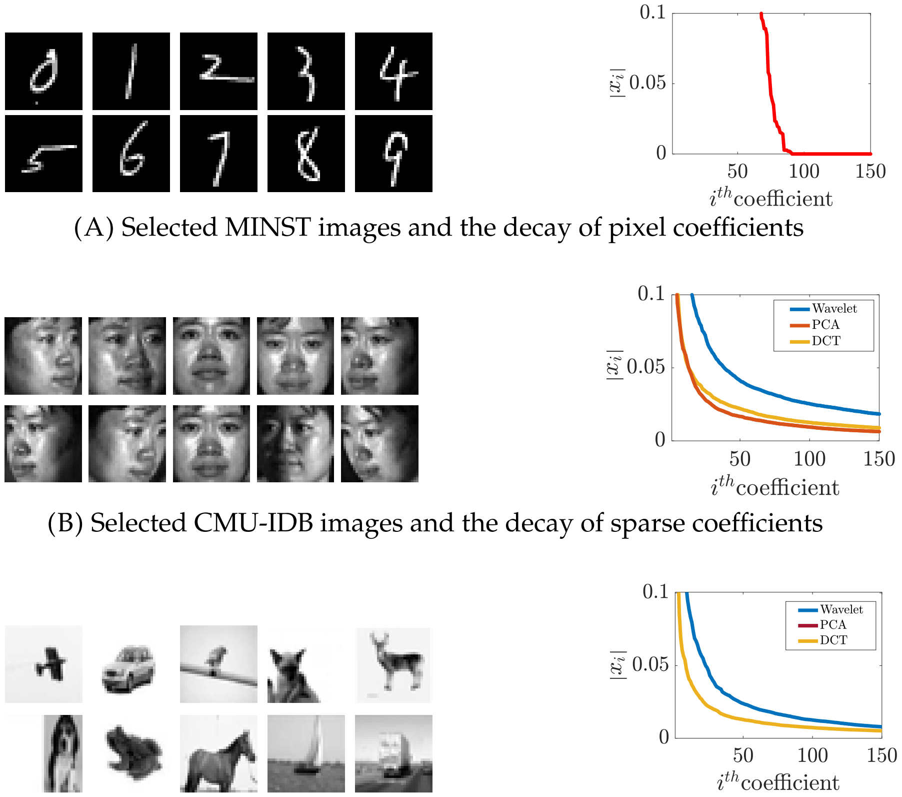
The matrix inversion $\mathcal{O}(N^3)$ can be reduced to $\mathcal{O}(M^3)$ where $M \ll N$ by applying the matrix inverse property. With the matrix property, Eq. (4.19) can be rewritten as

$$\boldsymbol{C}_t^{-1} = \boldsymbol{P}^{-1} - \boldsymbol{P}^{-1}\hat{\boldsymbol{S}}^T\boldsymbol{A}^T(\sigma_n^{-1}\boldsymbol{I} + \boldsymbol{A}\hat{\boldsymbol{S}}\boldsymbol{P}^{-1}\hat{\boldsymbol{S}}^T\boldsymbol{A}^T)^{-1}\boldsymbol{A}\boldsymbol{S}\boldsymbol{P}^{-1}, \tag{4.38}$$

where $\boldsymbol{P} = \boldsymbol{\Sigma}_t + \sigma_n \left(\mathrm{diag}(\hat{\mathbf{s}} \odot (1 - \hat{\mathbf{s}})) \odot (\boldsymbol{A}^T\boldsymbol{A})\right)$ is a diagonal matrix whose inverse can be computed easily. The complexity is reduced to $\mathcal{O}(M^3)$, where $M \ll N$.

The cost of matrix production is another dominant cost. The cost can be reduced by computing $\boldsymbol{A}^T\boldsymbol{A}$ offline, and reusing $\boldsymbol{A}^T\boldsymbol{y}$ and $\boldsymbol{y}^T\boldsymbol{y}$ which are required to calculate only once. With the matrix property (4.38), the total cost of matrix multiplication is $\mathcal{O}(MN^2 + 3M^2N + 2MN + 8N^2 + (9 + 4\mathbb{N})N + M)$. Therefore, the total complexity is reduced to $\mathcal{O}(M^3 + MN^2 + 3M^2N + 2MN + 8N^2 + (9 + 4\mathbb{N})N + M + k_0 + k_1) \approx \mathcal{O}(M^3 + MN^2 + 3M^2N + 2MN + 8N^2)$ where $M \ll N$.

This complexity is much less than that of the Two-step-Adaptive MRF (see Section 3.5). The complexity of the Two-step-Adaptive MRF is $\mathcal{O}(c_1(2M^3 + 4MN^2 + 3M^2N + 4N^2 + MN) + c_1|\mathcal{E}| + C(\mathcal{G}))$ per iteration which consists of the computational complexity from sparse recovery $\mathcal{O}(c_1(2M^3 + 4MN^2 + 3M^2N + 4N^2 + MN))$, the support estimation $\mathcal{O}(c_1|E|)$, and the MRF parameter estimation $\mathcal{O}(C(E))$. $|E|$ is the cardinality of the edge set in the graph, and c_1 is the number of iterations in which that sparse recovery is performed. Therefore, unlike the Two-step-Adaptive MRF, the proposed One-step-Adaptive MRF estimates the support and MRF parameters without performing additional subroutines, i.e., firstly, we estimate the support based on the expectation value Eq.(4.24) which is in a closed-form solution, rather than performing support inference. Secondly, with the proposed MRF distribution, we can update the MRF parameters with two closed-form solutions Eq.(4.27) and (4.36), rather than executing MRF parameter estimation as a subroutine.



(A) Selected MINST images and the decay of pixel coefficients

(B) Selected CMU-IDB images and the decay of sparse coefficients

(C) Selected CIFAR-10 images and the decay of sparse coefficients.

FIGURE 4.3: The ground truth images and the decay of sparse coefficients of (A) MNIST, (B) CMU-IDB, and (C) CIFAR-10 databases.

4.5 Experiment

In this section, we study the effectiveness of the proposed MRF distribution and the proposed One-step-Adaptive MRF through performing three different experiments: (i) we demonstrate the effectiveness of the proposed MRF distribution in approximating the original BM in comparison with the existing approximation schemes in Section 4.5.5; (ii) to study the improved performance due to the one-step approach, we demonstrate the effectiveness of the proposed one-step versus the two-step approaches in Section 4.5.6; and (iii) ultimately, the performance of the One-step-Adaptive MRF is shown in comparison with state-of-the-art algorithms in Section 4.5.7.

The details about the datasets, experiment settings, comparison methods, and evaluation criteria are described in the following sections.

4.5.1 Datasets

In this section, we evaluate the performance of the proposed One-step-Adaptive MRF on the three benchmark datasets— MNIST [113], CMU-IDB [114], and CIFAR-10 [115] (which are also used in Chapter 3.6.1 consistently). The test images selected for the experiment are shown in Figure 4.3. We employ the same compression process and linear transformations as described in Chapter 3.6.1. Therefore, we pay attention to the experiment results of (1) the MNIST digit images in the spatial domain, (2) CMU-IDB images in the PCA domain, (3) CMU-IDB images in the wavelet domain, (4) CMU-IDB images in the DCT domain, (5) CIFAR images in the wavelet domain, and (6) CIFAR images in the DCT domain, and we omit the discussion of the CIFAR-10 images in the PCA domain as the PCA signals are not sparse. All the reconstructed images are provided in Appendix A.2.

4.5.2 Experiment Setting

We employ the same experimental setting as the previous chapter (Section 3.6.2), i.e., in compression, the sparse signal $\boldsymbol{x}$ is sampled by a random Bernoulli matrix $\boldsymbol{A}$ to generate the linear measurements $\boldsymbol{y}$. The compression ratios (M/N) are set to 0.2, 0.25, 0.3, 0.35, and 0.4 to show their impact on the accuracy and run time at different measurement sizes. To simulate the noise corruption on measurements, three different levels of Gaussian white noise are added into $\boldsymbol{y}$, which results in the signal to noise ratio (SNR) of $\boldsymbol{x}$ to be 30 dB, 20 dB, 10 dB, and 5 dB. Please note that at the lowest SNR (5 dB), the measurements are mostly corrupted by noise; thus, the lowest SNR indicates the highest noise corruption. All the experiments were implemented by 64-bit MATLAB R2016b and were executed on a PC with Intel Core i7-4770 CPU and 16GB of RAM.

Algorithm setting: One-step-Adaptive MRF is initialized as follows: the hyper-parameters ϖ and ξ in Eq. (4.29) and ϖ_0 and ξ_0 in Eq. (4.32) are set to 10^{-6}. The initial value for α and β is set according to [39]. The edge set $E = \{\mathcal{E}_i\}$ is initialized as an empty set. For 2D signals, i.e., handwritten images and sparse representation in the wavelet domain, $\mathbb{N}_c$ and each $\mathbb{N}_i$ are set to cover 8-neighbors of each node. For 1D signals, i.e., sparse signal representations in the PCA and DCT domains, $\mathbb{N}_c$ and each $\mathbb{N}_i$ are set to cover two adjacent nodes. The algorithm stops when the minimum update difference, i.e. $\frac{||\mathbf{x}^{prev}-\mathbf{x}^{new}||_2}{||\mathbf{x}^{prev}||_2}$, is less than 10^{-3}, or when the iteration reaches 200.

4.5.3 Comparison methods

The performance of the proposed One-step-Adaptive MRF is compared with 7 state-of-the-art competitors:

- **Adaptive MRF based method**: Two-step-Adaptive MRF (TA-MRF) [45] and Pairwise MRF [42][2] ;
- **MRF-based methods (Non-Adaptive)**: MAP-OMP [32] and Gibbs [31] [3];
- **Cluster sparsity-based methods**: Bernoulli model [39][4];
- **Sparsity-based methods**: a Bayesian method RLPHCS[103] and a standard recovery method OMP[106].
- *The oracle estimator* [32] that employs *the ground truth support* in estimating the sparse signal (via Eq. (4.20)) shows the best possible result using ground truth support with homogeneous noise parameters. Note that all the other methods do not have access to the ground truth support. The oracle estimator has this unfair advantage.

All of the comparison methods, except Pairwise MRF [42], are implemented using the code of the authors with tuned parameters for the best performance. The Pairwise MRF is coded by ourselves and uses the same setting for $\mathbb{N}$ and terminating criterion as the proposed One-step-Adaptive MRF.

4.5.4 Evaluation criterion

We demonstrate the proposed One-step-Adaptive MRF performance on recovery accuracy and runtime performance. Similar to Chapter 3.6, the recovery accuracy

[2] For both TA-MRF and Pairwise MRF, we use the same setting for neighboring set $\mathbb{N}_i$, as described in Algorithm Setting in Section 4.5.2

[3] The graphical model, noise and signal variance parameters provided to MAP-OMP and Gibbs are from the training data.

[4] We use the same setting for neighboring set $\mathbb{N}_i$, as described in Algorithm Setting in Section 4.5.2

Sparsity (k)	Averaged KL-divergence between Our distribution Eq. (4.5) & the BM Eq. (4.4)	Bernoulli [39] Eq. (4.6) & the BM Eq. (4.4)	Pairwise [42] Eq. (4.7) & the BM Eq. (4.4)
10	**0.0020**	0.0281	3.0179
20	**0.0025**	0.0617	2.1777
30	**0.0026**	0.1103	1.4552

(A) Approximating the original BM across different sparsity levels.

Num. of edges (N)	Averaged KL-divergence between Our distribution Eq. (4.5) & the BM Eq. (4.4)	Bernoulli [39] Eq. (4.6) & the BM Eq. (4.4)	Pairwise [42] Eq. (4.7) & the BM Eq. (4.4)
$2^{\dagger}N^{\star}$	**0.0036**	0.0671	0.7409
$10^{\dagger}N^{\star}$	**0.0125**	0.0781	0.5337
$20^{\dagger}N^{\star}$	**0.0743**	0.1219	0.1039

†2, 10, and 20 are the number of pairwise edges connecting to each node.

$^{\star}N$ is the signal dimension

(B) Approximating the original BM across different numbers of edges.

TABLE 4.1: Effectiveness of the proposed MRF distribution Eq. (4.5) in approximating the original BM vs. existing approximation schemes: the Bernoulli model [39] Eq. (4.6), and the pairwise MRF [42] Eq. (4.7) across (a) different sparsity levels and (b) different numbers of edges.

is evaluated by the peak signal to noise ratio (PSNR). We consider the total runtime required by each algorithm across different sampling rates (M/N).

4.5.5 Effectiveness of the proposed distribution for the MRF

This section demonstrates the effectiveness of the proposed MRF distribution Eq. (4.5) in approximating the Boltzmann machine (BM) Eq. (4.4) by measuring the KL-divergence between these two distributions. The effectiveness of the proposed distribution is compared with those of some existing approximation schemes, i.e., the Bernoulli model [39] Eq. (4.6) and the pairwise MRF [42] Eq. (4.7). Table 4.1A and Table 4.1B demonstrate the approximation to the BM across different configurations: (i) sparsity levels, $k = 10, 20, 30$, and (ii) number of edges, $2N$, $10N$, $20N$. The KL-divergence is averaged over 1000 empirical distributions.

Table 4.1A provides the effectiveness in approximating BM across different sparsity levels (k): 10, 20, and 30. Each sparsity is induced by tuning the unary parameters. The unary parameters are randomly from $\mathcal{N}(\mu_b, 1)$ with $\mu_b = -2.5, -2$, and -1.5, each of which enforces a different sparsity level. The pairwise parameters are randomly selected from $\mathcal{N}(-0.1, 1)$. Clearly, the KL-divergence of the proposed MRF distribution much smaller by at least *one order* of magnitude and *three orders* of magnitude in comparison with the Bernoulli model and the Pairwise MRF.

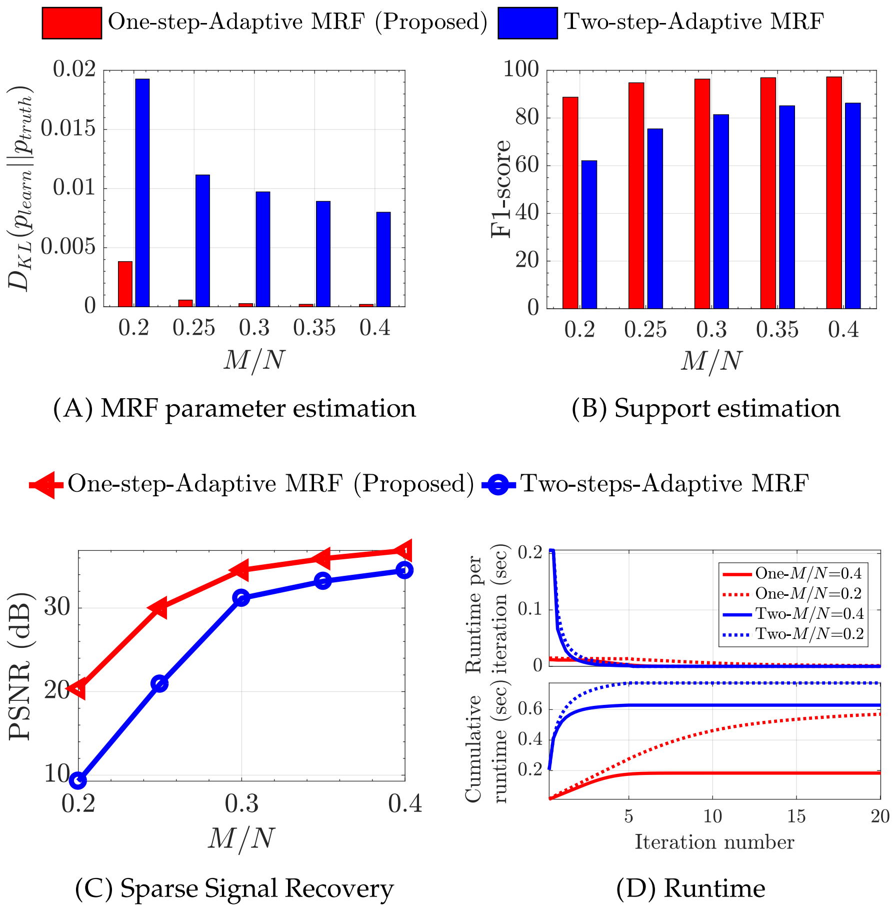
Table 4.1B provides the effectiveness in approximating the BM across different numbers of edges ($|E|$): $2N$, $10N$, $20N$ where $2, 10$, and 20 are the number of pairwise edges connecting to each node, and N is the total number of nodes corresponding to N support coefficients. Here, both the unary and the pairwise parameters are randomly selected from $\mathcal{N}(\cdot, 1)$ with mean of -1 and -0.3, respectively. The proposed MRF distribution provides the smallest KL-divergence in all cases. The KL-divergence of the proposed MRF distribution is, at most, 17% and 3% of the Bernoulli model and the Pairwise MRF when $|E| < 20N$. When $|E| = 20N$, our KL-divergence is approximately 60% and 70% of the Bernoulli model and the Pairwise MRF. These two experiments demonstrate that with unary and pairwise parts, the proposed MRF distribution Eq. (4.5) can best approximate the BM (4.4) across different configurations.

4.5.6 Effectiveness of MRF parameter estimation: One-step vs Two-step

We compare the effectiveness of the proposed One-step-Adaptive MRF versus the Two-step-Adaptive MRF [5] in estimating MRF parameters for 10,000 signals sampled from 10 distributions. The effectiveness is evaluated by the parameter estimation, measured by the KL-divergence between the estimated model and the ground truth, and the final performance, measured by the F1-score, recovery accuracy, and runtime. Figures 4.4 and 4.5 show the results across different sampling rates (M/N) and noise levels (in the SNR). In Figure 4.4, the KL divergence of the proposed One-step-Adaptive MRF is less than 25% of the Two-step-Adaptive MRF. Our approach also yields at least a 5% higher F1-score[6], with 2 dB higher accuracy with a lower runtime. Although the proposed One-step-Adaptive MRF uses more iterations to converge, its total runtime is much lower than the Two-step-Adaptive approach that has to perform MRF estimations and sparse signal estimations in each iteration. In Figure

[5] Here, the recovery accuracy and KL-divergence of the Two-step-Adaptive approach is measured at the main algorithm, rather than at the subroutine (signal estimation).

[6] For the proposed One-step-Adaptive MRF, the F1-score is calculated from the binary support obtained from Eq. (4.37).



(A) MRF parameter estimation

(B) Support estimation

(C) Sparse Signal Recovery

(D) Runtime

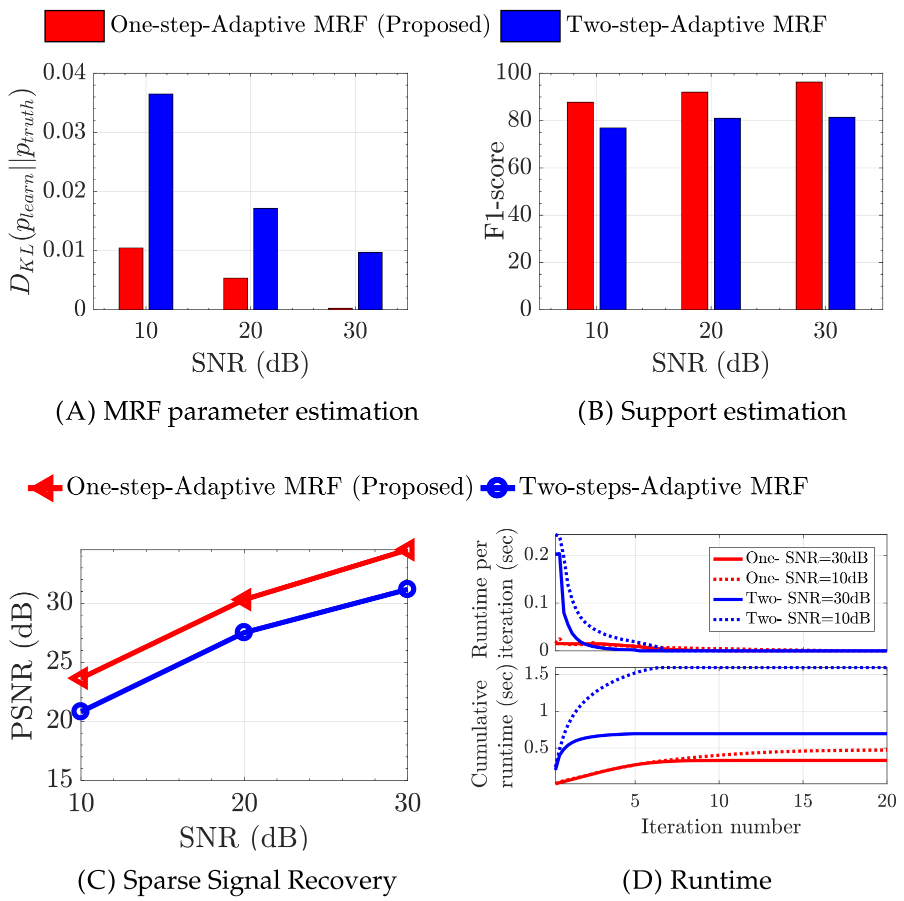
FIGURE 4.4: Effectiveness of the MRF parameter estimation by the proposed One-step-Adaptive MRF vs Two-step-Adaptive MRF [45] across different sampling rates: (a) quality of MRF parameters estimation, (b) accuracy of support estimation, (c) accuracy of sparse signal recovery, and (d) average runtime per iteration. Noise level (SNR) is 30 dB.

4.5, the KL divergence of the proposed One-step-Adaptive MRF is less than 30% of the Two-step-Adaptive MRF. Our approach also yields at least a 5% higher F1-score[3] and 3 dB higher accuracy with less runtime. Thus, the proposed One-step-Adaptive MRF offers more efficient MRF parameter estimation than the Two-step-Adaptive MRF.

4.5.7 Performance Evaluation

4.5.7.1 Compressibility.

This section demonstrates the compressibility performance of the proposed One-step-Adaptive MRF. We evaluate the recovery performance across different sampling rates (M/N). Figure 4.6 shows the average PNSR curves across different sampling



(A) MRF parameter estimation

(B) Support estimation

(C) Sparse Signal Recovery

(D) Runtime

FIGURE 4.5: Effectiveness of the MRF parameter estimation by the proposed One-step-Adaptive MRF vs Two-step-Adaptive MRF [45] across different noise levels: (a) quality of MRF parameter estimation, (b) accuracy of support estimation, (c) accuracy of sparse signal recovery, and (d) average runtime per iterations. Sampling rate is 0.3.

rates on the six image sets, when the noise level (SNR) is 30 dB. The proposed One-step-Adaptive MRF offers the highest performance in most cases. Because both the proposed One-step-Adaptive MRF and Two-step-Adaptive MRF employs the flexible and adaptive prior, they outperform the other methods across different datasets of images with different types of signal structures, i.e. MNIST, CMU-IDB, and CIFAR-10. With the improved parameter estimation, the proposed One-step-Adaptive MRF yields the highest performance, which is higher than the Two-step-Adaptive MRF, across different datasets: for MNIST, the proposed One-step-Adaptive MRF exceeds the second most competitive method by at least 2 dB, when the sampling rate is higher than 0.25. For CMU-IDB, it exceeds the second most competitive method by at least 1 dB in the wavelet domain, 0.5 dB in the PCA domains and 2 dB in the DCT domain when the sampling rate is higher than 0.25. For CIFAR-10, it exceeds the

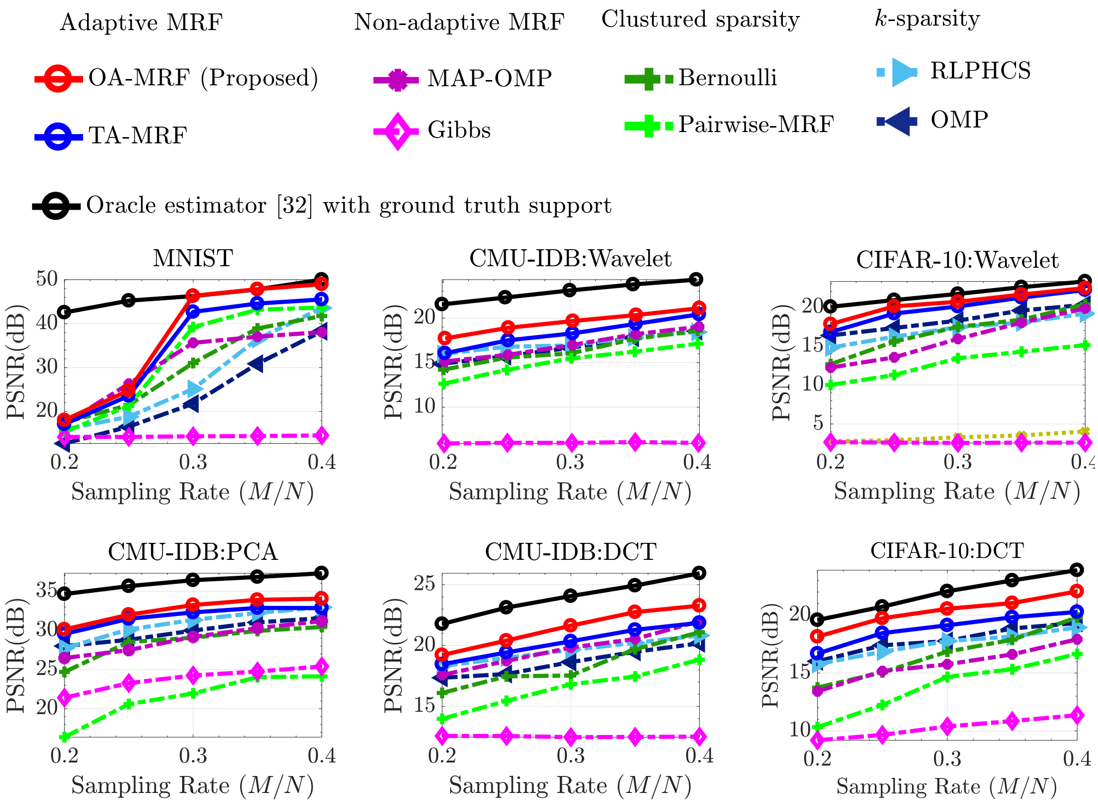


FIGURE 4.6: Compressibility. The PSNR curves across different sampling rates on three datasets. The noise level (SNR) is 30 dB.

second most competitive method by at least 0.25 dB in the wavelet domain and 2 dB in the DCT domain. With the improved adaptive MRF parameter estimation, the proposed One-step-Adaptive MRF yields the highest performance.

The visual results of the proposed method on a MNIST handwritten digit image, a CMU-IDB face image, and a CIFAR-10 natural image are provided in Figures 4.7, 4.8, and 4.9. The sampling rates are 0.3. The proposed One-step-Adaptive MRF clearly gives rise to the best results with more detail and less noise than its competitors. The full visual results are in Appendix A.2.

4.5.7.2 Noise tolerance.

This section demonstrates the noise tolerance performance. We test the performance of the proposed One-step-Adaptive MRF across different noise levels (in the SNR). Figure 4.10 provides the average PNSR curves across different noise levels on six image sets. The sampling rate (M/N) is set to 0.3. The proposed One-step-Adaptive MRF outperforms the other comparison methods in most cases.

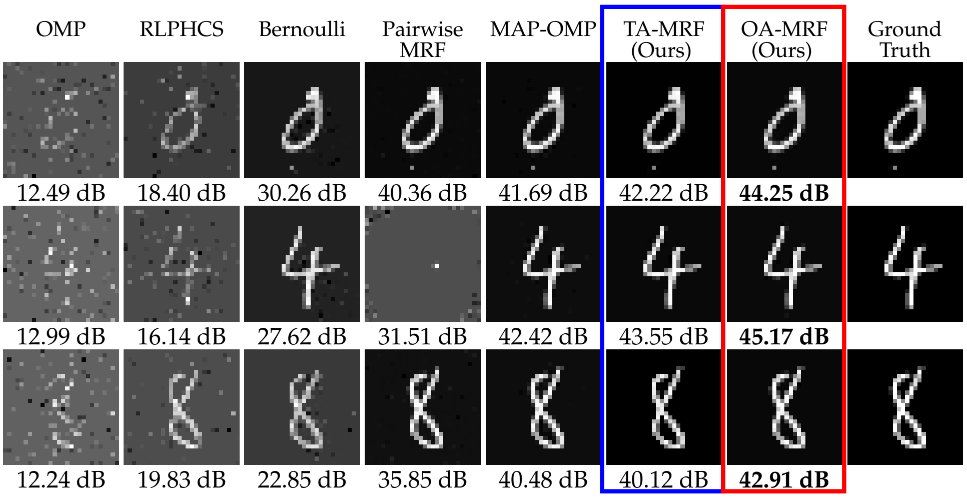


FIGURE 4.7: Visual results of the selected MNIST handwritten digit images by the top seven most competitive methods, i.e. OMP, RLPHCS, Bernoulli, Pairwise MRF, MAP-OMP, TA-MRF, and the proposed OA-MRF, at $M/N = 0.3$, SNR = 30 dB.

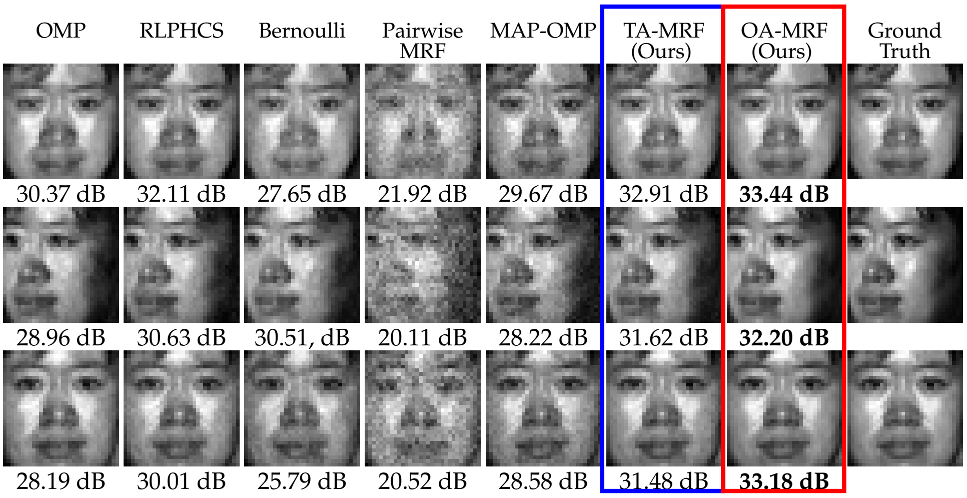


FIGURE 4.8: Visual results of the selected CMU-IDB face images from sparse signal recovery in the PCA domain by the top seven most competitive methods, i.e. OMP, RLPHCS, Bernoulli, Pairwise MRF, MAP-OMP, TA-MRF, and the proposed OA-MRF, at $M/N = 0.3$, SNR = 30 dB.

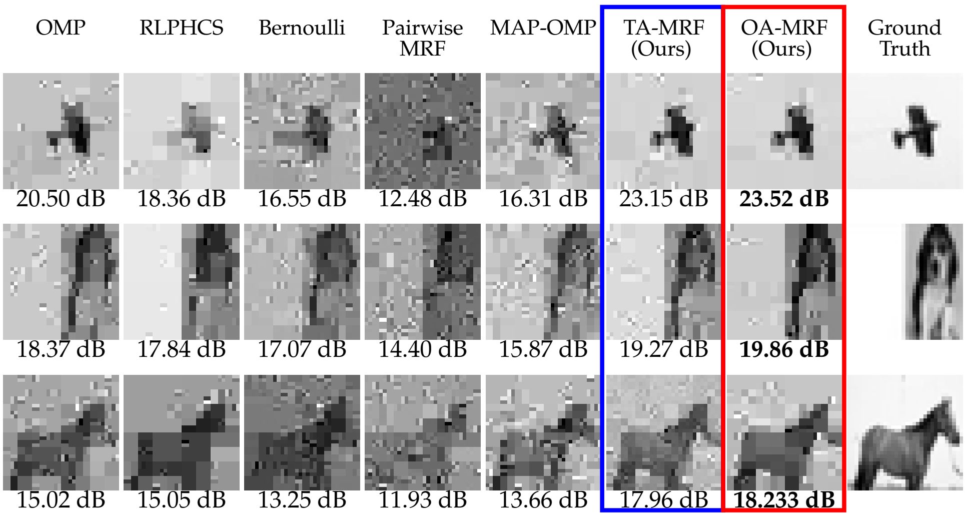


FIGURE 4.9: Visual results of the selected CIFAR-10 from sparse signal recovery in the wavelet domain by the top seven most competitive methods, i.e. OMP, RLPHCS, Bernoulli, Pairwise MRF, MAP-OMP, TA-MRF, and the proposed OA-MRF, at M/N = 0.3, SNR = 30 dB.

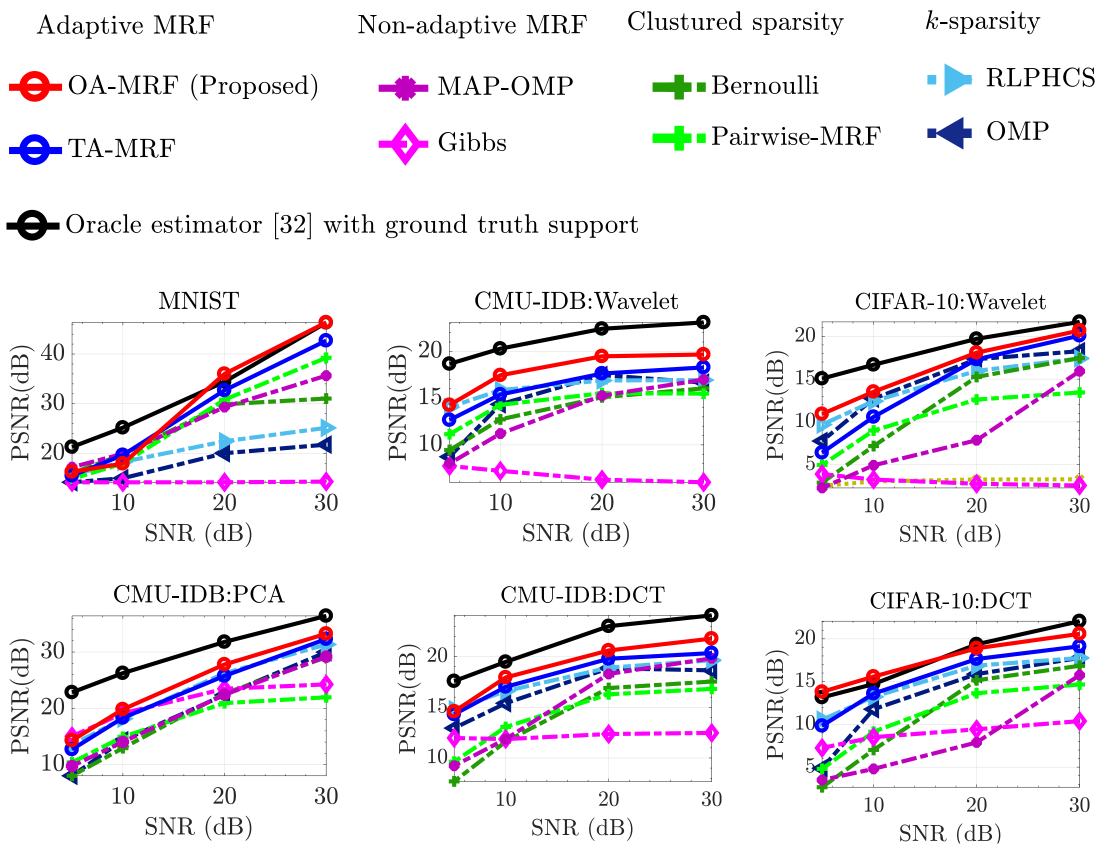


FIGURE 4.10: Noise Tolerance. The PSNR curves across different noise levels (SNR) on three datasets. The sampling rate is 0.3.

Because both the proposed One-step-Adaptive MRF and the Two-step-Adaptive MRF employ the flexible and adaptive prior, they outperform the other structured and non-structured CS methods. With the improved MRF parameter estimation, the proposed One-step-Adaptive MRF significantly improves the performance of the Two-step-Adaptive MRF as well as outperforms the other methods across different datasets: On MNIST images, the proposed One-step-Adaptive MRF outperforms the second best method by at least 2 dB. For CMU-IDB, it exceeds the second best method by at least 2 dB in the wavelet domain, 1 dB in the PCA domains, and 1 dB in the DCT domain. For CIFAR-10, it exceeds the second best method by at least 1 dB in the wavelet domain and 2 dB in the DCT domain.

Note that for the recovery of MNIST images and CIFAR-10 images in the DCT domain, the proposed One-step-Adaptive MRF even outperforms the oracle estimator. This is because the proposed One-step-Adaptive MRF enables heterogeneous noise parameters, which are obtained from the adaptive noise estimation, whereas the

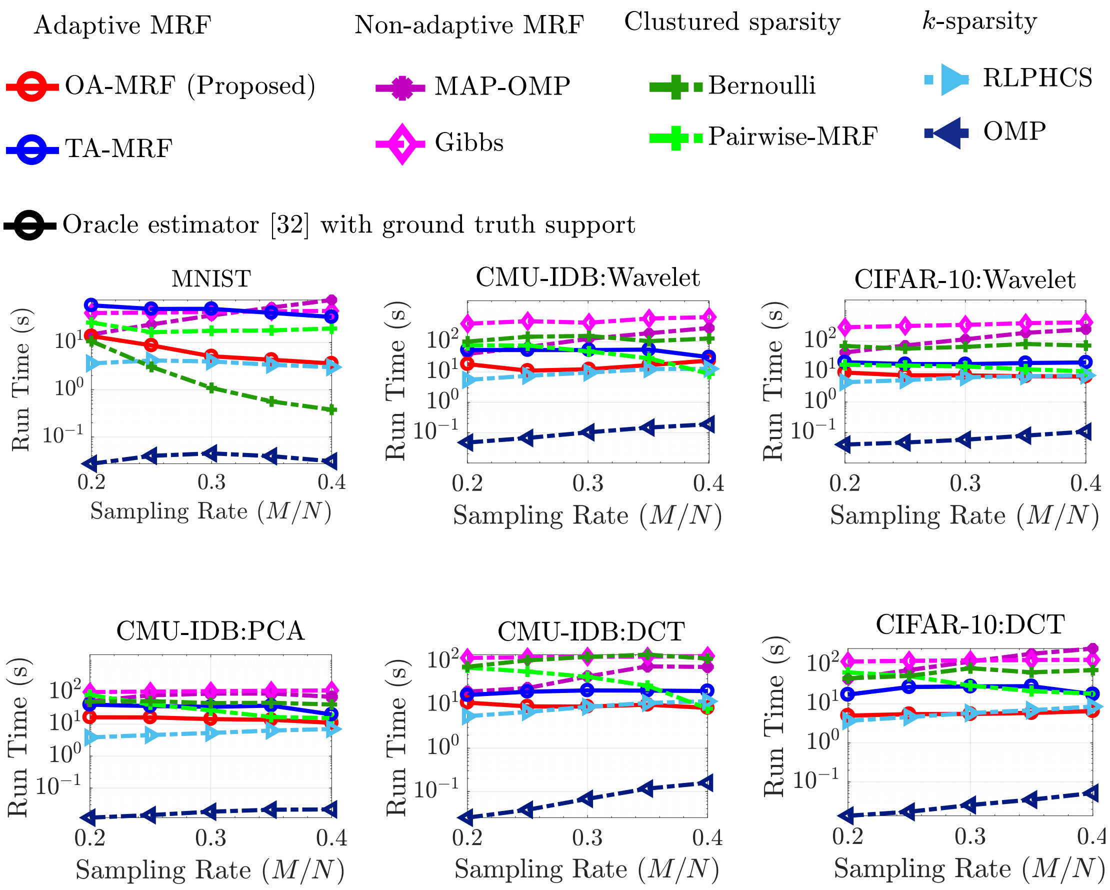


FIGURE 4.11: Runtime performance. Runtime curves across different sampling rates on three datasets. The noise level (SNR) is 30 dB.

oracle estimator uses homogenous noise parameters, which are obtained from the training data. This indicates that the adaptive mechanism provides a good prior to help identify signal information from noisy measurements. The proposed One-step-Adaptive MRF is more tolerant to noise than the Two-step-Adaptive MRF in most cases. This demonstrates that with improved adaptive parameter estimation, the proposed One-step-Adaptive MRF is able to achieve superior noise tolerance performance.

4.5.7.3 Runtime.

In this section, we study the runtime of the proposed One-step-Adaptive MRF in comparison with other competitors by observing the runtime performance across different sampling rates (M/N). Figure 4.11 provides the runtime performance at different sampling rates on the three datasets (the noise level is 30 dB.): On MINST, the average runtime of the proposed One-step-Adaptive MRF is moderate compared with the others. It is faster than the Two-step-Adaptive MRF, MAP-OMP, and Pairwise-MRF; is comparable to RLPHCS; but is slower than Bernoulli and OMP. For CMU-IDB and CIFAR-10, the runtime performance of the proposed One-step-Adaptive MRF is much better than many structured CS methods. The runtime performance is similar across the wavelet, DCT, and PCA domains, i.e., the proposed One-step-Adaptive MRF is faster than the Two-step-Adaptive MRF, MAP-OMP, Bernoulli, and Pairwise MRF. Its runtime is comparable with RLPHCS and only slower than OMP. Note that OMP and RLPHCS require low computation in general because they do not exploit the structure in sparse signal coefficients.

This demonstrates that the proposed One-step-Adaptive MRF has a moderate runtime in most cases, and its runtime performance is obviously improved from that of the Two-step-Adaptive MRF.

4.6 Conclusion

We have presented a novel one-step Markov random field (MRF) based structured CS to adaptively estimate the MRF parameters from a few measurements. A very recent method estimates the MRF parameters from a point estimation of the sparse signals. However, the point estimation cannot depict the statistical uncertainty of the latent sparse signals. Therefore, we propose to estimate the MRF parameters from the measurement by solving a maximum marginal likelihood. The marginal likelihood is obtained from averaging over all the sparse signal population, thus, it generalizes over all the latent sparse signals more effectively. A new MRF distribution is proposed to enable closed-form formulations to estimate the MRF parameters. Extensive experiments demonstrate the performance of the two novel components of the proposed One-step-Adaptive MRF—the new MRF distribution and the one-step approach. First, the proposed MRF distribution best approximates the Boltzmann machine in comparison with some of the existing approximation schemes. Second, we conduct experiments that demonstrate the superior performance in the MRF parameter estimation of the proposed one-step method over the two-step method on synthesized data. Finally, we demonstrates the overall performance in comparison with the state-of-the-arts in compressibility, noise tolerance, and runtime. The proposed One-step-Adaptive MRF can achieve the best performance in most cases and with using a moderate runtime.

Chapter 5

Application to Human Activity Recognition

We have proposed an adaptive Markov random field (MRF) that offers high flexibility to capture and adapt for any structure of the sparse signals. The MRF parameter estimation underpins the performance of the proposed adaptive MRF. In the previous chapter, we proposed the One-step-Adaptive MRF that estimates the MRF parameters from measurements directly by solving a maximum marginal likelihood problem. As the marginal likelihood can effectively depict the statistical uncertainly of the latent sparse signals, the resulting adapted MRF parameters can well generalize the underlying structure of the entire sparse signal population, which leads to state-of-the-art performance over existing methods [29]–[33], [35], [36], [38]–[42]. Therefore, the underlying structure of the sparse signals extracted from the measurements offers a good prior knowledge for sparse signal recovery.

One-step-Adaptive MRF can be useful for many applications related to sparse signal recovery. Among many applications, collaborative-representation based classifications (CRCs) can directly benefit from One-step-Adaptive MRF to extract the underlying structure directly from the query sample which can be a good indication of the class label. The underlying structure brings the new information that is unique to its corresponding query sample and independent of the quality of the training samples. Motivated by this, this chapter presents an application of the proposed One-step-Adaptive MRF to improve the performance of CRCs.

CRCs have offered state-of-the-art performance in many applications, including wearable sensor-based human activity recognition, when the training samples are limited. Most of the existing methods are based on the shortest Euclidean distance from a query sample to each group of training data. These methods can be susceptible to noise and correlation in the training samples. To improve the robustness, we propose to employ the adaptive MRF to extract the underlying structure of the representation vector from the query sample. The underlying structure offers additional information that can be a good indication of the class and is unique to the corresponding query sample. Thus, it can improve the discriminative power of the classifier. We apply the proposed One-step-Adaptive MRF to effectively estimate the adaptive MRF from the given query sample. The adaptive MRF can be customized to further reduce the ambiguity due to the correlation in the training samples. With adaptive MRF, the classification performance significantly improves.

5.1 Introduction

Human activity recognition has played a crucial role in behavioral monitoring and human-computer interactions driven by a wide variety of applications, ranging from health care and assistive technology [119]–[126] to underground mining [127], [128]. With the increasing interest in healthcare applications fueled by the Internet of Things (IoT), daily human activity recognition technologies have received much attention to realize robust and continuous health monitoring [122]. Among many the human activity recognition technologies, wearable sensor systems have the main advantages of being unobtrusive, privacy-preserving, maintenance free, and economical in power consumption and size. Therefore, some human activity recognition research has been conducted, based on wearable sensor technologies suitable for acquiring salient information about gestures and body motions, without having direct contact with users/objects of interest. Human activity recognition based on wearable sensors is, however, a challenging task, since it often has to handle streams of data with a large variability, either due to the changes in human body behaviors or noises in the system. The difficulty of recognition is multiplied, especially when the amount of training data is small, which is a typical situation when obtaining large training samples is not financially viable [129], [130] or when a low sampling rate is employed to limit the power consumption in parsimonious and self-powered systems [131], [132]. Consequently, this can lead to overfitting to a small number of training samples that can either be noisy or correlated with one another, since each activity is a combination of body motions and movements.

To effectively recognize human activities, many researchers have developed robust classification approaches. Parametric classification approaches such as support vector machines (SVMs) have been dominating the field of pattern recognition for their ability to extract the salient information through learning a model from the training samples [133]. However, when the number of training samples is small, these parametric approaches often become overfitting. To avoid the overfitting problem, researchers resort to non-parametric approaches [89]–[92], [98]–[101], [133]–[135] that employ the training samples to predict the class labels directly, without learning a classification model. Among these methods, the collaborative representation-based classifications (CRCs) offer state-of-the-art performance in many applications [89], [91], [98]–[101]. The performance of the CRCs depends on the reconstruction of a representation vector $\boldsymbol{x}$ that is used to identify the class label. Given a query sample, the representation vector $\boldsymbol{x}$ is obtained from solving the following linear model:

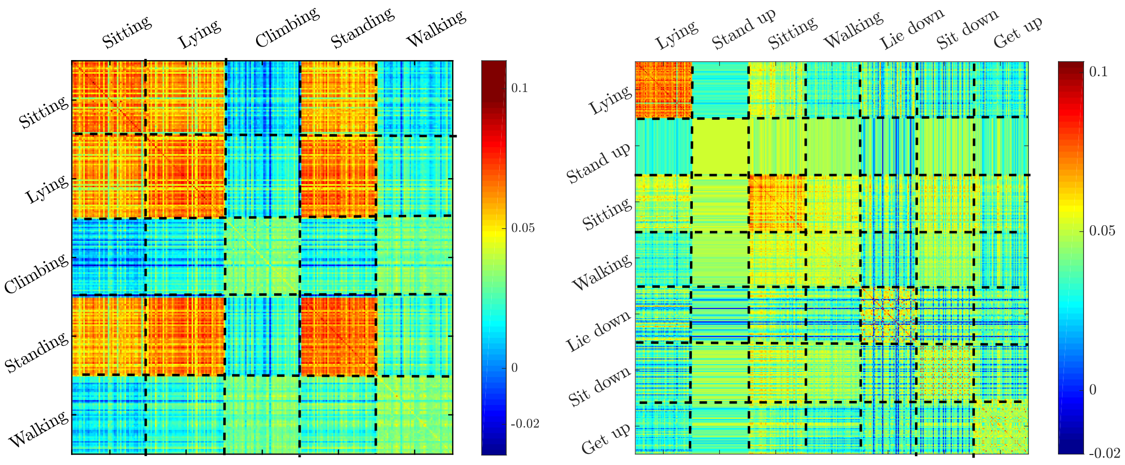
$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{n} \tag{5.1}$$

where $\boldsymbol{n}$ represents a small perturbation in the query sample. The samples matrix $\boldsymbol{A}$ contains all the training samples from different activity classes sorted according to the class labels:

$$\boldsymbol{A} = [\boldsymbol{A}_1, ..., \boldsymbol{A}_C], \tag{5.2}$$

where $\boldsymbol{A}_c = [\boldsymbol{a}_i^c, ..., \boldsymbol{a}_{i+n_c-1}^c] \in \mathcal{R}^{M \times n_c}$ contains training samples of the c^{th} class label. CRC aims to recover the representation vector that has the shortest Euclidean distance to the query sample $\boldsymbol{y}$.

A similar approach to CRCs employs the sparsity as a prior knowledge in the recovery of the representation vector $\boldsymbol{x}$ to increase robustness in the classification, especially when the query sample is noisy [89]. This method is commonly known as sparse representation-based classifications (SRCs). Although CRCs and SRCs have



(A) UCI-HAR

(B) Hospital dataset

FIGURE 5.1: Examples of the correlation level between (feature extracted) training samples sorted according to the class labels in the UCI-HAR and the Hospital dataset. 50 training samples per class.

been developed for vision applications, both methods have been applied to many applications, including wearable based-human activity recognition. In wearable based-human activity recognition, the training data are not guaranteed to be noise-free. Despite this, the general CRCs and SRCs offer state-of-the-art performance [91], [99]–[101]. Although CRCs and SRCs have shown promising results in these challenging applications, their intrinsic classification mechanism remains unclear. A recently developed ProCRC [98] offers the probabilistic interpretation of the CRCs and proposes jointly maximizing the likelihood that a test sample belongs to each of the multiple classes. To achieve this, ProCRC [98] aims at recovering the representation vector that minimizes the linear approximation residual across different classes (i.e. $||\boldsymbol{A}\boldsymbol{x} - \boldsymbol{A}_c\boldsymbol{x}_c||_2 \ \forall c \in \{1, ..., C\}$ in Eq. (2.59)), and provides the shortest Euclidean distance to the query. So far, ProCRC offers state-of-the-art performance in vision applications. However, when training samples from two or more classes are correlated, none of these methods, i.e. CRCs, SRCs, and ProCRC, has a mechanism to help discriminate these training samples from each other; as a result, the classification can be inaccurate.

Figure 5.1 demonstrates the examples of correlation in (feature extracted) training samples sorted according to the class labels. The training samples of the activities that involve similar body motions are more correlated; meanwhile, the training samples of the activities that involves different body motions are less correlated. For example, in

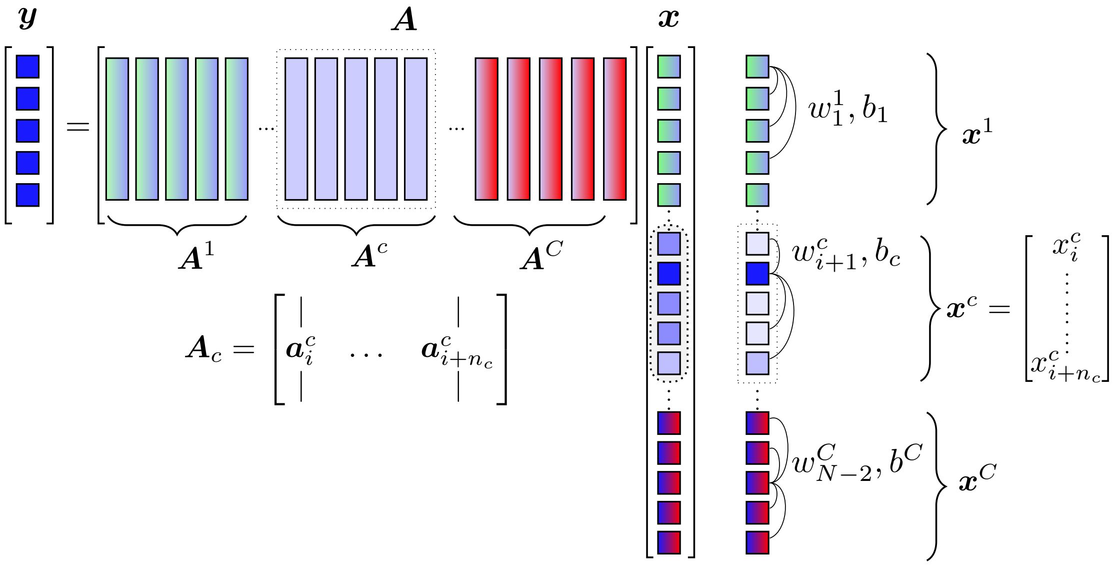

$\boldsymbol{y}$
$\boldsymbol{A}$
$\boldsymbol{x}$
$=$
$\boldsymbol{A}^1$
$\boldsymbol{A}^c$
$\boldsymbol{A}^C$
$\boldsymbol{A}_c = \begin{bmatrix} | & & | \\ \boldsymbol{a}_i^c & \dots & \boldsymbol{a}_{i+n_c}^c \\ | & & | \end{bmatrix}$
w_1^1, b_1
$\boldsymbol{x}^1$
w_{i+1}^c, b_c
$\boldsymbol{x}^c = \begin{bmatrix} x_i^c \\ \vdots \\ x_{i+n_c}^c \end{bmatrix}$
w_{N-2}^C, b^C
$\boldsymbol{x}^C$

FIGURE 5.2: Relationship between the training samples and the representation vectors in the linear observation model (5.1). The partition of training samples can be used to customize our adaptive-MRF to capture the local correlation among coefficients of the representation vectors in each class, and disconnect the underlying graph in the MRF across different classes.

the UCI-HAR dataset *Sitting* and *Lying* and *Standing* are highly correlated. Meanwhile, in the Hospital dataset [136], multiple pairs of activities are weakly correlated to one another, e.g., *Sitting* is weakly correlated to many classes, e.g. *Lying*, *Sit down*, and *Walking*, since many older hospitalized volunteers participate in collecting the Hospital dataset.

To address this problem, we propose employing the underlying structure of the representation vector extracted from the query. The underlying structure offers additional information that can be a good indicator of the class label of the corresponding query sample. The underlying structure brings new information that is unique to its corresponding query sample and independent of the quality of the training samples. Thus, the underlying structure can improve the discriminative power of the classifier. We apply the proposed One-step-Adaptive MRF in Chapter 4 to extract the underlying structure of the representation vector effectively and capture it with the adaptive MRF. The adaptive MRF can be customized according to the partition of the training samples to further reduce the ambiguity due to the correlated training samples.

To implement this, we first group the representation vector according to the partition of training samples to reduce the ambiguity due to the correlation in the training samples. Figure 5.2 demonstrates the group in the representation vectors and the corresponding training samples in the linear observation model Eq. (5.1). Then, our adaptive-MRF is applied to capture the local correlation or dependency in each group of the representation coefficients, e.g., $\boldsymbol{x}_{\mathbb{N}_c} = [x_i; ...; x_{i+n_c-1}] \in \mathcal{R}^{n_c}$ associated with the training samples $\boldsymbol{A}_c = [\boldsymbol{a}_i^c, ..., \boldsymbol{a}_{i+n_c-1}^c] \in \mathcal{R}^{M \times n_c}$ in the c^{th} class, where $\boldsymbol{x} = [\boldsymbol{x}_{\mathbb{N}_1}; ...; \boldsymbol{x}_{\mathbb{N}_C}]$. To represent the underlying structure among the coefficients $x_i, ..., x_{i+n_c-1}$, an MRF is employed. The parameters of the MRF are the pairwise parameters $\{w_i^c, ..., w_{n_c-1}^c\}$ and the unary parameter b_c. The pairwise parameters represent the interaction among the coefficients; meanwhile, the unary parameters provide the bias toward zero. The MRF parameters are adaptively estimated, based on a given query sample. These parameters represent the underlying structure of the representation vector that can be seen as the weights assigned to emphasize each group of training samples, according to the query. Then, the MRF is used as a prior in the recovery of the representation vector.

To efficiently reconstruct the representation vector and estimate the MRF parameters, we apply One-step-Adaptive MRF to perform the classification task. We propose a new adaptive-MRF-based classification method where the adaptive MRF is used as a prior in recovering the representation vector $\boldsymbol{x}$. To evaluate the performance of the proposed classification, we employ the UCI-HAR and the Hospital dataset in Figure 5.1 to demonstrate the performance in two different scenarios: (i) the UCI-HAR dataset is employed to demonstrate the scenario when training samples across different activities are highly correlated, and (ii) the Hospital dataset is used to demonstrate the scenario when training samples from different activities are weakly correlated. Based on our experiments on these two real-world datasets, and with the evaluation based on sample-based classification and activity-based label misalignment measures, our proposed method offers state-of-the-art performance across the different numbers of training samples.

5.2 Related works

To facilitate discussion, we review the following related non-parametric and parametric classification techniques in this research:

Parametric approaches

- *i)SVMs* [133], [137], [138] is an efficient parametric classifier that is associated with learning for an appropriate hyperplane that maximizes the margin between two classes. The hyperplane is learned over the training samples. SVMs employ the learned hyperplanes to determine the class labels given the query sample.

- *ii) CNN* [136], [139], [140] has been adopted to HAR for its deep architecture and the variety of processing units that can effectively extract the salient features representing the signals. The features extracted from the CNN are task-dependent and non-handcrafted [136], [139]. So far, CNN has offered state-of-the-art recognition performance because the learned CNN is able to extract the underlying correlation in the training samples. It also yields discriminative power, since the CNN is learned using the training samples with the respective class labels.

SVMs and CNN-based classifiers are very efficient parametric classifiers. These two methods rely on tremendous training samples. However, when training samples are small, their respective model learning can suffer severely from overfitting problems [89], [141], [142]. To address this approach, non-parametric classifiers are shown to be an alternative approach to these models [98].

Non-Parametric approaches

- *i) kNNs* [134] are based on the principle that the samples in a dataset will generally exist in close proximity to other samples that have similar properties. They determine the classes of the query samples based on the most frequent class labels of the k nearest training samples. To improve the robustness when training samples are correlated, k is often chosen to be small [135]. However, if k is too small, the classifier can be sensitive to noise in training and query samples.

- *CRCs* [89], [93] and their variations *SRCs* [90]–[92] and ProCRC [98] predict a class label of a given query sample $\boldsymbol{y}$ based on solving a linear problem $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}$ for a representation vector $\boldsymbol{x}$, where each column in $\boldsymbol{A}$ collects a training sample from different classes. *CRCs* employ l_2-norm regularization to weakly impose sparsity on representation vectors. *SRC* employs l_1-norm regularization, which

> strongly imposes sparsity on the representation vectors. This often improves robustness against any noise in the query sample [90]–[92]. However, when training samples are correlated, the solution $\boldsymbol{x}$ is not necessary sparse [89], [93], [98]. ProCRC [98] offers state-of-the-art performance by exploiting the likelihood between the query sample and each group of training samples. The review of CRCs, SRCs, and ProCRC [98] is provided in Chapter 2.

Although these non-parametric classifiers are able to address the overfitting problem, a kNN that relies on the number of neighboring samples is often highly sensitive to noise in both the query and the training samples. Meanwhile, CRCs, SRCs, and ProCRC are more robust [89], [90], [93], [98] since they are based on the shortest Euclidean distances between the query and all the training samples. SRCs are more robust than CRCs against noise, whilst CRCs and ProCRC can better reconstruct the representation vector when the training samples are correlated [89], [98]. Nevertheless, none of these methods has a mechanism to discriminate these training samples directly. Unlike these methods, our approach can exploit the underlying structure of the representation vector, and improve discriminative performance by customizing the underlying graph of the MRF to unlink the correlation between representation coefficients from different classes.

5.3 Graphical collaborative representation

To model the underlying structure in the representation vector $\boldsymbol{x}$, we model the underlying structure through the support coefficients of the representation vector. Inspired by the One-step-Adaptive MRF (Chapter 4), the representation vector $\boldsymbol{x}$ is decomposed into a support vector $\boldsymbol{s} \in \{0,1\}^N$ and a scale vector $\boldsymbol{t} \in \mathcal{R}^N$, i.e., $\boldsymbol{x} = \boldsymbol{t} \odot \boldsymbol{s}$. Thus, our goal is to recover $\boldsymbol{t}$ and $\boldsymbol{s}$ from the following linear observation model

$$\boldsymbol{y} = \boldsymbol{A}(\boldsymbol{t} \odot \boldsymbol{s}) + \boldsymbol{n}, \tag{5.3}$$

where $\boldsymbol{A} = [\boldsymbol{A}_1, ..., \boldsymbol{A}_C] \in \mathcal{R}^{M \times N}$ is a matrix where all the training samples are sorted according to the class label $1, ..., C$. The query sample $\boldsymbol{y}$ is corrupted by additive Gaussian white noise $\boldsymbol{n}$ with the noise precision σ_n^{-1}. Thus, the corresponding observation likelihood can be formulated as

$$p(\boldsymbol{y}|\boldsymbol{t}, \boldsymbol{s}; \sigma_n) = \mathcal{N}(\boldsymbol{A}(\boldsymbol{t} \odot \boldsymbol{s}), \sigma_n^{-1}\boldsymbol{I}). \tag{5.4}$$

where $\boldsymbol{I}$ is an identity matrix with proper size. Generally, given some appropriate prior models, e.g., $p(\boldsymbol{t})$ and $p(\boldsymbol{s})$, the latent $\boldsymbol{t}$ and $\boldsymbol{s}$ can be inferred by solving the following MAP problem

$$\{\hat{\boldsymbol{t}}, \hat{\boldsymbol{s}}\} = \max_{\boldsymbol{t},\boldsymbol{s}} p(\boldsymbol{t}, \boldsymbol{s}|\boldsymbol{y}) \propto (\boldsymbol{y}|\boldsymbol{t}, \boldsymbol{s})p(\boldsymbol{t})p(\boldsymbol{s}). \tag{5.5}$$

In the following sections, we will discuss the prior models $p(\boldsymbol{t})$ and $p(\boldsymbol{s})$, respectively.

5.3.1 Adaptive-MRF

To reduce ambiguity due to correlation in the training samples, we can further customize the underlying graph of the MRF to disconnect the representation coefficients corresponding to the partition of the training samples. Let us first consider the group of the representation coefficients $\boldsymbol{x}_{\mathbb{N}_c} = [x_i; ...; x_{i+n_c-1}] \in \mathcal{R}^{n_c}$ that is associated with the training samples $\boldsymbol{A}_c = [\boldsymbol{a}_i^c, ..., \boldsymbol{a}_{i+n_c-1}^c] \in \mathcal{R}^{M \times n_c}$ in the c^{th}. Here, $\boldsymbol{x}_{\mathbb{N}_c} = \boldsymbol{t}_{\mathbb{N}_c} \odot \boldsymbol{s}_{\mathbb{N}_c}$. The coefficient members in $\boldsymbol{t}_{\mathbb{N}_c}$ and $\boldsymbol{s}_{\mathbb{N}_c}$ are allowed to share the MRF parameters $\{w_i^c\}_{i \in \mathbb{N}_c}, b_c$ within the c^{th} class only. The MRF Eq. (4.2.1) can be customized to model the local dependency among support coefficients as follows:

$$\begin{aligned} & p(\boldsymbol{s}|\boldsymbol{b}, \boldsymbol{w}) \\ & = \prod_{c=1}^{C} p(\boldsymbol{s}_{\mathbb{N}_c}|b_c, \boldsymbol{w}_c) = \prod_{c=1}^{C} \prod_{i \in \mathbb{N}_c} p_u^c(s_i; b_c) p_p^c(s_i; w_i^c) \\ & = \prod_{i \in \mathbb{N}_1} p_u^1(s_i; b_1) p_p^1(s_i; w_i^1) \cdot ... \cdot \prod_{i \in \mathbb{N}_c} p_u^c(s_i; b_c) p_p^c(s_i; w_i^c) \cdot ... \cdot \prod_{i \in \mathbb{N}_C} p_u^C(s_i; b_C) p_p^C(s_i; w_i^c) \end{aligned} \tag{5.6}$$

where $\forall i \in \mathbb{N}_c$,

$$\begin{aligned} p_u^c(s_i; b_c) &= \text{Bernoulli}(s_i|b_c) \quad \text{with} \quad b_c \sim \text{Beta}(\alpha, \beta); \\ p_p^c(s_i; w_i^c) &= \frac{1}{Z(w_i^c, \{s_j\}_{j \in \mathcal{E}_i})} \exp(s_i \sum_{j \in \mathcal{E}_i} w_i^c s_j). \end{aligned}$$

for all $c = 1, ..., C$. The unary term $p_u^c(\cdot)$ provides the bias toward zero that influences only the supports within the same neighbor. Meanwhile, the pairwise term $p_p^c(\cdot)$ represents the interaction between them. Thus, we restrict the pairwise edge to connect with the coefficients from the same class only, i.e., $\mathcal{E}_i \subseteq \mathbb{N}_c \setminus s_i$. Note that the cardinality of $\mathbb{N}_c$ is equal to the number of training samples in each class.

In connection with the support model, we impose a statistical model onto the signal scale coefficients corresponding to the c^{th} class label as $\boldsymbol{t}_c$ to weakly induce the structure among them:

$$p(\boldsymbol{t}_{\mathbb{N}_c}; {\sigma_t}^c) = \prod_{i \in \mathbb{N}_c} \mathcal{N}(t_i | 0, {\sigma_t}^c \boldsymbol{I}) \tag{5.7}$$

where ${\sigma_t}^c$ denotes the variance of scale coefficients in $\mathbb{N}_c$, and $\boldsymbol{I}$ is an identity matrix with proper size. To facilitate the computation, we impose the hyperprior, i.e. $p(\sigma_n; \varpi_0, \xi_0)$, for the noise variance σ_n in Eq. (4.10). We also impose the hyperprior for the unary and signal scale parameters, i.e. b_c and ${\sigma_t}^c$ from Eq. (4.6) and Eq. (4.9), i.e. $p(b_c; \alpha, \beta)$ and $p({\sigma_t}^c; \varpi, \xi)$ respectively.

5.4 Adaptive MRF-based classification

With all these corresponding hyperpriors, and the given support and signal scale models from Eq. (5.3.1) and (5.7), our objective is to reconstruct signal scale and support, i.e. $\boldsymbol{t}$ and $\boldsymbol{s}$, given the query sample $\boldsymbol{y}$ by solving the following maximum a posteriori (MAP) problem:

$$\begin{aligned} (\hat{\boldsymbol{t}}, \hat{\boldsymbol{s}}) =& \max_{\boldsymbol{s},\boldsymbol{t}} p(\boldsymbol{t}, \boldsymbol{s} | \boldsymbol{y}, \boldsymbol{\Theta}) \propto p(\boldsymbol{y} | \boldsymbol{t}, \boldsymbol{s}, \boldsymbol{\Theta}) p(\boldsymbol{t}, \boldsymbol{s} | \boldsymbol{\Theta}) p(\boldsymbol{\Theta}) \\ &= p(\boldsymbol{y}, \boldsymbol{t}, \boldsymbol{s} | \boldsymbol{\Theta}) \prod_{c=1}^{C} p({\sigma_t}^c; \varpi, \xi) p(b_c; \alpha, \beta) p(\sigma_n; \varpi_0, \xi_0). \end{aligned} \tag{5.8}$$

where

$$\begin{aligned} p(\boldsymbol{y}, \boldsymbol{t}, \boldsymbol{s} | \boldsymbol{\Theta}) &= p(\boldsymbol{y} | \boldsymbol{t}, \boldsymbol{s}, \sigma_n) \prod_{c=1}^{C} p(\boldsymbol{t}_{\mathbb{N}_c} | {\sigma_t}^c) \prod_{c=1}^{C} p(\boldsymbol{s}_{\mathbb{N}_c} | b_c, \boldsymbol{w}_{\mathbb{N}_c}) \\ &\propto \exp\left(-\frac{1}{\sigma_n} ||\boldsymbol{y} - \boldsymbol{A}(\boldsymbol{t} \odot \boldsymbol{s})||_2^2 - \sum_{c=1}^{C} \sum_{i \in \mathbb{N}_c} \frac{1}{{\sigma_t}^c} t_i^2 + \sum_{c=1}^{C} \sum_{i \in \mathbb{N}_c} (b_c s_i + s_i \sum_{j \in \mathbb{N}_c \setminus i} w_i^c s_j) \right) \end{aligned}$$

and $\boldsymbol{\Theta} = \{\sigma_n, \boldsymbol{\sigma}_t, \boldsymbol{b}, \boldsymbol{w}\}$; $\boldsymbol{\sigma}_t = [\sigma_t{}^c]$, $\boldsymbol{b} = [b_c]$, $\boldsymbol{w} = [\boldsymbol{w}_{\mathbb{N}_c}]$; $\boldsymbol{w}_{\mathbb{N}_c} = [w_i^c]_{i \in \mathbb{N}_c}$, $\boldsymbol{t}_{\mathbb{N}_c} = [t_i]_{i \in \mathbb{N}_c}$ and $\boldsymbol{s}_{\mathbb{N}_c} = [s_i]_{i \in \mathbb{N}_c}$. To extract the underlying structure of the representation vector, we estimate the model parameters $\boldsymbol{\Theta}$ directly from the query. To achieve this, we estimate the MRF parameters by solving the following maximum marginal likelihood problem, where the latent variable $\boldsymbol{s}$, $\boldsymbol{t}$ will be integrated out. Then, given $\boldsymbol{\Theta}$, the representation vector is estimated by solving MAP Eq.(5.8).

5.4.1 MRF parameter estimation with variational EM

With the hyperpriors imposed on $\sigma_n, \boldsymbol{\sigma}_t$, and $\boldsymbol{b}$ in $\boldsymbol{\Theta}$, these random variables can be considered as latent variables. $\boldsymbol{w}$ is the only unknown parameter. Thus, we group these latent variables into a set, denoted as $\boldsymbol{\Lambda}$, i.e. $\boldsymbol{\Lambda} = \{\boldsymbol{t}, \boldsymbol{s}, \sigma_n, \boldsymbol{\sigma}_t, \boldsymbol{b}\}$. The unknown parameter $\boldsymbol{w}$ can be estimated by solving a maximum marginal likelihood (MML) problem: $\max_{\boldsymbol{w}} \ln p(\boldsymbol{y}|\boldsymbol{w}) \propto \int \ln p(\boldsymbol{y}, \boldsymbol{\Lambda}|\boldsymbol{w}) \mathrm{d}\boldsymbol{\Lambda}$. To solve this MML problems, all the latent variables in $\boldsymbol{\Lambda}$ are to be integrated out. Since calculating the integral is intractable, we employ the same variational EM [116] technique provided in Section 4.3.1 to solve the MML problem. All the unknown variables in $\boldsymbol{\Lambda}$ are calculated through approximating the true posterior $p(\boldsymbol{y}, \boldsymbol{\Lambda}|\boldsymbol{w})$ in Expectation step. As $\boldsymbol{t}$ and $\boldsymbol{s}$ are estimated in this Expectation step, we do not have to solve MAP Eq. (5.8). The estimation for the unknown parameters $\boldsymbol{w}$ are calculated by maximizing the lower bound $F(\boldsymbol{q}, \boldsymbol{w})$ in Maximization step. All the updating rules in Expectation and Maximization steps are derived in a similar fashion to those of the One-step-Adaptive MRF, except that the edge set is set to include only variable nodes in its local neighborhood $\mathbb{N}_c$, i.e., $\mathcal{E}_i \subseteq \mathbb{N}_c \setminus s_i$, in Algorithm 4.9. Therefore, we refer to Section 4.3.2 for the details and derivation for each update rule.

5.4.2 Adaptive-MRF-based Classifier

After we employ the variational EM process described in Section 4.3 to recover the support $\boldsymbol{s}$ and sparse signal scale vector $\boldsymbol{t}$, the solution for the representation vector is obtained by performing piece-wise product between them, i.e., $\hat{\boldsymbol{x}} = \hat{\boldsymbol{t}} \odot \hat{\mathbf{s}}$. Given the solution representation vector $\boldsymbol{x}$, we apply a similar technique to CRC to find the

Algorithm 5.11 Adaptive-MRF-based classification

Input: Test sampling vector $\boldsymbol{y}$ and the matrix $\boldsymbol{A}$ containing the training sample sorted in order, according to the class labels.

1. Recover the representation vector $\hat{\boldsymbol{x}}$ by solving $\hat{\boldsymbol{t}}$ and $\hat{\boldsymbol{s}}$ from the following MAP problem Eq. 5.8 using One-step-Adaptive MRF algorithm 4.10:

$$(\hat{\boldsymbol{t}}, \hat{\boldsymbol{s}}) = \max_{\boldsymbol{s},\boldsymbol{t}} p(\boldsymbol{t}, \boldsymbol{s}, |\boldsymbol{y}, \boldsymbol{\Theta}) \propto p(\boldsymbol{y}, \boldsymbol{t}, \boldsymbol{s}|\boldsymbol{\Theta}) p(\sigma_n; \varpi_0, \xi_0) \prod_{c=1}^{C} p(\sigma_t{}^c; \varpi, \xi) \prod_{c=1}^{C} p(b_c; \alpha, \beta) \tag{5.10}$$

where

$$\begin{aligned} p(\boldsymbol{y}, \boldsymbol{t}, \boldsymbol{s}|\boldsymbol{\Theta}) &= p(\boldsymbol{y}|\boldsymbol{t}, \boldsymbol{s}, \sigma_n) \prod_{c=1}^{C} p(\boldsymbol{t}_{\mathbb{N}_c}|\sigma_t{}^c) \prod_{c=1}^{C} p(\boldsymbol{s}_{\mathbb{N}_c}|b_c, \boldsymbol{w}_{\mathbb{N}_c}) \\ &\propto \exp\left(-\frac{1}{\sigma_n}||\boldsymbol{y} - \boldsymbol{A}(\boldsymbol{t} \odot \boldsymbol{s})||_2^2 - \sum_{c=1}^{C} \sum_{i \in \mathbb{N}_c} \frac{1}{\sigma_t{}^c} t_i^2 + \sum_{c=1}^{C} \sum_{i \in \mathbb{N}_c} (b_c s_i + s_i \sum_{j \in \mathbb{N}_c \setminus i} w_i^c s_j) \right) \end{aligned}$$

The edge set is set to include only variable nodes in the local neighborhood $\mathbb{N}_c$, i.e., $\mathcal{E}_i \subseteq \mathbb{N}_c \setminus s_i$, in Algorithm 4.9. Then, $\hat{\boldsymbol{x}} = \hat{\boldsymbol{s}} \odot \hat{\boldsymbol{t}}$.

2. Given the solution $\hat{\boldsymbol{x}}$, the output class label of query signal $\boldsymbol{y}$ is found by

$$\hat{c} = \min_{c \in \{1,...,C\}} \{r_c\}, \quad \text{where} \quad r_c = \frac{||\boldsymbol{y} - \boldsymbol{A}_c \hat{\boldsymbol{x}}_c||_2}{||\boldsymbol{x}_c||_2}. \tag{5.11}$$

Output: The class label $\hat{c}$.

label of the query sample $\boldsymbol{y}$:

$$\hat{c} = \min_{c} ||\boldsymbol{y} - \boldsymbol{A}_c \boldsymbol{x}_c||. \tag{5.9}$$

The process of Adaptive-MRF-based classification is summarized in Algorithm 5.11.

Compared with the existing work, namely CRC, ProCRC, and SRC, our objective function Eq. 5.8 is most similar to CRC Eq. (2.53), except that it contains the additional terms associated with the MRF distribution, such as $\left(b_c s_i + s_i \sum_{j \in \mathbb{N}_c \setminus i} w_i^c s_j\right)$, to capture the underlying structure of the representation vector $\boldsymbol{x}$. These two terms can be seen as the weights assigned to emphasize each class of training samples. As all the MRF are directly estimated from the query samples, the information gained from the query sample helps to recognize the class that has the highest approximation to the query sample. Additionally, to reduce ambiguity due to the correlation between each group of training samples, we separate the MRF parameters according to the group of the training samples. Unlike the ProCRC, we do not try to minimize the residual of the approximated linear combination corresponding to each class (i.e. $||\boldsymbol{Ax} - \boldsymbol{A}_c\boldsymbol{x}_c||_2 \; \forall c \in \{1, ..., C\}$ in Eq. (2.59)) as that can impose a wrong influence, if a pair of training sample groups, $\boldsymbol{A}_i$ and $\boldsymbol{A}_j$, are correlated. Thus, the recognition can provide an ambiguous result as either the class label i or j can be recognized as the activity label. Compared with SRC, the employed MRF prior is more flexible than the sparsity l_1 norm as in SRC (Eq. (2.51)), especially when the columns in $\boldsymbol{A}$ are correlated to one another; thus, the representation vector $\boldsymbol{x}$ is not sparse.

5.5 Algorithm complexity

The algorithm complexity is dominated by the computation of the One-step-Adaptive MRF executed to reconstruct $\boldsymbol{x}$, i.e., $\mathcal{O}(N^3 + 2MN^2 + 11N^2 + 5MN + (9 + 4\mathbb{N})N + M + k_0 + k_1) \approx \mathcal{O}(N^3 + 2MN^2 + 11N^2 + 5MN)$ which can be reduced to $\mathcal{O}(2N^2 + 3MN)$ per iteration where p_0 is a constant, by setting the noise variance σ_n and the signal scale variance $\boldsymbol{\Sigma}_t^{-1}$ to some appropriate values. As a result, some of the matrices associated with the noise and the signal scale variances can be computed offline, as described below.

According to Section 4.4, it is clear that the computation is dominated by the matrix inversion and the matrix production are performed during the estimation of the sparse signal scale Eq. (4.19). The computation cost for Eq. (4.19) is $\mathcal{O}(N^3 + MN^2 + 4N^2 + 3N)$. With the appropriate value for the noise variance σ_n and the signal scale variance $\boldsymbol{\Sigma}_t^{-1}$, we approximate the matrix inversion for $\boldsymbol{C}_t$ in Eq. (4.19) as follows: $\boldsymbol{u}_t = \hat{\sigma}_n \boldsymbol{C}_t^{-1} \hat{\boldsymbol{S}} \boldsymbol{A}^T \boldsymbol{y} \approx \hat{\sigma}_n \hat{\boldsymbol{S}} \tilde{\boldsymbol{C}}_t^{-1} \hat{\boldsymbol{S}} \boldsymbol{A}^T \boldsymbol{y}$, where $\tilde{\boldsymbol{C}}_t^{-1} = \hat{\boldsymbol{\Sigma}}_t + \hat{\sigma}_n \boldsymbol{A}^T \boldsymbol{A}$, which can be calculated offline. Meanwhile, $\boldsymbol{A}^T y$ is required to be computed only once, and $\boldsymbol{A}^T \boldsymbol{A}$ can be computed offline. Therefore, the computational cost is reduced to $\mathcal{O}(N^2 + 2N)$. Given the noise and signal variance, it is unnecessary to compute Eq. (4.33) and Eq. (4.30). Meanwhile, the estimation of the support Eq. (4.24) requires only $\mathcal{O}(N^2 + 3MN + 4N\mathbb{N} + M) \approx \mathcal{O}(N^2 + 3MN)$. Hence, these two dominant computations are completely removed. Thus, the total computational complexity can be reduced to $\mathcal{O}(2N^2 + 3MN)$ per iteration. In Section 5.6.7, the runtime performance of our method is provided. It is shown that the our algorithm performs classification with an affordable runtime—it performs in under 0.002 second for a query sample in practice.

5.6 Experiment

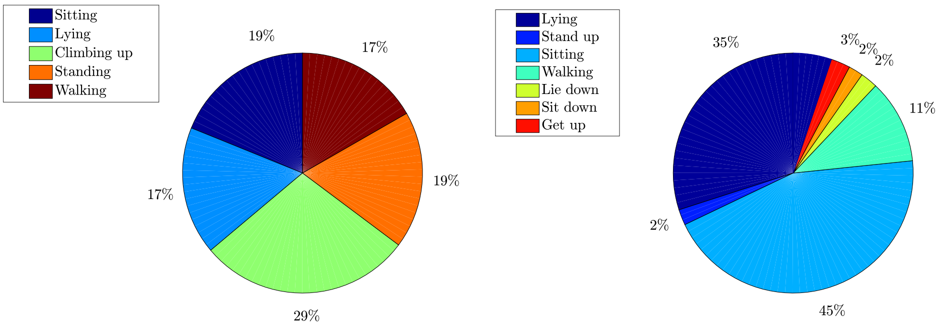
To demonstrate and verify the performance of the proposed Adaptive MRF-based classification, we conduct three different experiments: (i) to demonstrate the effectiveness of our Adaptive MRF-based classification across different numbers of training samples, we evaluate the performance using traditional sample-based classification measures in Section 5.6.5; (ii) to provide an in-depth analysis of the performance of the proposed method, we provide activity-based misalignment measures in Section 5.6.6; (iii) we investigate the runtime performance to demonstrate the expected efficiency of our method in Section 5.6.7.

The details about the datasets (e.g. the UCI-HAR and Hospital datasets), experiment settings, the comparison methods , and evaluation criterion (e.g. sample-based classification measures and activity-based misalignment) are described in the following sections.

5.6.1 Datasets

In this section, we evaluate the performance of our Adaptive MRF-based classification on the two real-world datasets: UCI-HAR [143] and Hospital dataset [136]. These datasets are different in trial setting, activity duration, class frequencies, as well as the level of correlation within the training samples across different activities.

UCI-HAR. The UCI-HAR dataset records the daily activities of thirty volunteers within the age bracket 19-48 years. Each volunteer is asked to perform five activities, i.e., walking, climbing up/down stairs, sitting, standing, and lying. A smartphone (Samsung Galaxy SII) with an embedded accelerator and gyroscope is worn at the waist of each volunteer where the tri-axial linear acceleration and the tri-axial angular velocity are acquired at the sampling rate of 50 Hz. The obtained dataset has been randomly selected into training and query sets where 70% and 30% of the volunteers were selected for generating the training and the query samples, respectively.



(A) UCI-HAR

(B) Hospital Dataset

FIGURE 5.3: The class distributions in the UCI-HAR and the Hospital dataset. The class distribution in the UCI-HAR is well balanced. Meanwhile, the class imbalance is more pronounced in the Hospital dataset.

Hospital dataset. The Hospital dataset records seven activities from twelve volunteers hospitalized older patients, i.e. lying, stand up, sitting, walking, lie down, sit down and get up. The trial in [136] employed only those volunteers who are 65 years or older, living at home, and able to consent to the study and mobilize independently. A single inertial sensor (Bosh BMI160) with an integrated accelerator and gyroscope sensor unit of 24 mm in diameter and 7 mm in thickness is used to collect the tri-axial linear acceleration and the tri-axial angular velocity at the sampling rate of 20 Hz. The data from the first eight volunteers are used in the training phase; meanwhile, the data from the last three volunteers are used in the testing phase.

Figure 5.3 shows the class distributions of the two datasets. Notably, the class imbalance problem is more pronounced in the Hospital dataset than in the UCI-HAR. However, in our experimental setting (see Section 5.6.4), we chose an equal number of training samples for each class to form a small training set, which reduces the impact of the class imbalance problem in the training phase. Therefore, the class imbalanced problem impacts more on the testing phase. We address this problem by using a weighted measure (e.g. the weighted F_1-score) to balance the score across different activities in proportion to the number of test samples. Nevertheless, the ambiguity problem due to the correlation between training samples cannot be reduced by this setting since the correlation between them is caused by the similarity in the body motions between different activities.

5.6.2 Experiment setting

To show the performance of the proposed method, we demonstrate the recognition performance across different numbers of training samples. To do so, we perform the following steps.

Feature extraction and selection. To demonstrate the classification performance, we employ a standard set of features that have been used commonly in wearable sensor-based human activity recognition. In calculating these features, we follow the direction in [91]. The window size is set up to 4×(Sampling Rate) with 50% overlapping. To extract the features from the UCI-HAR, a window of size 2.56×(Sampling Rate) with 50% overlapping is employed; meanwhile, the window size is set to 4×(Sampling Rate) with 50% overlapping for extracting the features in the Hospital dataset.

We employ the hand-craft features commonly used in human activities recognition. To extract information based on body movement in different activities, we employ the statistical features and probabilistic features that summarize the statistical information regarding the underlying activities within a window duration. The statistical features are the mean, median, variance, standard deviation, root mean square value, and interquartile range, and the probabilistic features are the correlation, entropy, skewness, and kurtosis. To capture the instantaneous changes of the body motions within a window duration, we extract transient behavior-based features, i.e. the first and the second order derivative, and zero crossing. Frequency-based features are obtained to summarize the frequency of the movements corresponding to each activity. The following list provides the features considered in this study:

- Statistical features—the mean, median, variance, standard deviation, root mean square value, interquartile range;
- Probabilistic features—the correlation, entropy, skewness, kurtosis;
- Transient behavior-based features—the first and the second order derivative, and zero crossing;
- Frequency-based features—the energy in frequency domain, spectral energy, dominant frequency;

The total number of these features is ninety-six (16 × 6). The top fifty features are selected based on the sequential forward selection, which is reported in [132], [144] as a very effective feature selection method for wearable sensor-based human activity recognition. These extracted features are used in our classification approach and all the comparison methods, except the CNN [139] which requires a special setting for the feature size.

Removing outlier in training samples and constructing the training sets. In HAR, the data acquired from sensors are often noisy. Some of the training samples cannot accurately represent the corresponding activity class labels. These outliers can be detected by a clustering technique [145] where it is assumed that training samples corresponding to the same class are organized into clusters. The outliers are those samples that depart from the cluster and are closer in proximity to the samples from the other class. This outlier can be measured by a euclidean distance between a training sample of interest and the other samples from the same class. To achieve this, we employ a classification technique similar to CRC [89], but without l_2 regularization, to remove those training samples, described as follows:

Given a samples matrix $\tilde{\boldsymbol{A}} = [\tilde{\boldsymbol{A}}_1, ..., \tilde{\boldsymbol{A}}_C] \in \mathcal{R}^M \times N$ collecting features extracted from training samples where $\tilde{\boldsymbol{a}}_i^c \in \mathcal{R}^M$ is the i^{th} training samples in the c^{th} class label; N is the total number of samples; M is the number of features used to represent each training sample. We identify a training sample in $\tilde{\boldsymbol{A}}$ as an outlier, if it is misclassified as the other class. For example, given a training sample $\tilde{\boldsymbol{a}}_i^l$ labeled as the l^{th} class, we will classify this training sample as the outlier, if its representation vector $\hat{\boldsymbol{x}}$ is obtained from

$$\hat{\boldsymbol{x}} = \min_{\boldsymbol{x} \in \mathcal{R}^N} ||\tilde{\boldsymbol{A}}\boldsymbol{x} - \tilde{\boldsymbol{a}}_i^l||_2, \tag{5.12}$$

and provides the class label $c^* \neq l$ where $c^* = \min_{c \in [C]} ||\tilde{\boldsymbol{A}}_c \hat{\boldsymbol{x}}_c - \tilde{\boldsymbol{a}}_i^l||_2$ is the class that has the closest euclidean distance to $\tilde{\boldsymbol{a}}_i^l$.

Once all the outliers are removed, we use the representative training samples to construct the multiple training sets to demonstrate the recognition performance of all the candidate algorithms. Our main objective for the experiments is to evaluate the classification performance across different numbers of training samples. To construct training sets with multiple sizes, we randomly select 10, 20, 30, 40, and 50 training samples per class from the entire set of training samples. For each selection, the training samples are randomly picked 100 times to provide a stable performance result. Nevertheless, notice that the correlation between training samples cannot be reduced by this setting, since the correlation between them is caused by the similarity between body movements in different activities. An example of how training samples are correlated is shown Figure 5.1 where 50 training samples per class are randomly selected to form a training set. Here, only approximately 4% and 12% of training samples in the UCI-HAR and the Hospital datasets are removed.

Algorithm setting. The initialization of the proposed Adaptive-MRF classification is set as follows: (i) the noise and signal parameters, σ_n and $\mathbf{\Sigma}_t$, are set to 10^{-10} and 10^{-12}; (ii) the edge set $E = \{\mathcal{E}_i\}$ is initialized as an empty set, while $\mathbb{N}$ forces each set $\mathcal{E}_i$ to include only the nodes in the same neighborhood, i.e., for $\mathcal{E}_i$ corresponds the i^{th} node in the c^{th} class label, $\mathbb{N} = \mathbb{N}_c \setminus s_i$. Here, the setting for $\mathbb{N}_c$ is $\mathbb{N}_c = \{s_{i-\frac{n_c}{2}}, ..., s_{i+\frac{n_c}{2}}\}$ is set to consider the relationship between the n_c coefficients in the (1-D) representation vector $\boldsymbol{x}$, corresponding to the training samples in the c^{th} class label $\boldsymbol{A}_c$, i.e. $\boldsymbol{a}^c_1, ..., \boldsymbol{a}^c_{n_c}$. Here, $|\mathbb{N}_c| = n_c$ is equal to the number of training samples. (iii) The algorithm will stop when the minimum update difference between two consecutive estimates, defined as $\frac{||\mathbf{x}^{prev} - \mathbf{x}^{new}||_2}{||\mathbf{x}^{prev}||_2}$, is less than 10^{-3}, or when the number of iterations reaches 200.

5.6.3 Comparison methods

The performance of our method is compared with 6 state-of-the arts competitors:

- Non-parametric approaches: (i) CRC [89], (ii) ProCRC [98] , (iii) SRC [91], (iv) kNN [134];
- Parametric approaches: (v) SVM [137] and (vi) CNN [139].

All of these comparison methods are implemented using the code of the authors with tuned parameters to the best performance. This is except for the implementation for SRC [91] that are not accessible. We implemented SRC ourselves by following the suggestions in [91], which recommends using the l_1-magic package [146] to solve for sparse signals and set the noise parameter σ_n in Eq. (2.51) to 0.03. For CNN [139], although CNN employs raw training samples, we specifically chose from the same set of training samples used in other algorithms.

5.6.4 Evaluation criterion

We mainly evaluate the proposed Adaptive MRF based on (i) the traditional sample-based classification measures and (ii) the activity-based misalignment measures. The details of each type of measure are described as follows:

Sample-based classification measures. This evaluation approach is a standard measure to compare different classification approaches. We expect our adaptive-MRF based approach to achieve high performance, especially when the training samples are small, and demonstrate better performance than the existing works. We employ two-widely used sample-based classification metrics, i.e. the weighted F-measure (F_W) and accuracy. Because of the uneven number of activities contained in the sequences of query samples, we adopted the weighted F-measure [136] where the F_1-score is weighted according to the proportion of the number of query samples corresponding to each activity:

$$F_W = \sum_{c=1}^{C} \bar{w}_c f_c \quad \text{where} \quad f_c = 2\frac{p_c r_c}{p_c + r_c} \times 100. \tag{5.13}$$

c is the class index; $\bar{w}_c = n_c^{test}/N^{test}$ with n_c^{test} the number of query samples in the cthclass, N^{test} the total number of query samples; p_i denotes precision while r_i represents recall. Notice that these sample-based classification measures (weighted F-measure (F_W) and accuracy) only measure the number of misses and hits in query samples; however, it cannot quantify the misalignment of the predicted activity sequences, e.g., the loss of duration, and the delays in predicting activities.

Activity-based misalignment measure. We employ the activity misalignment measures [147] to verify the quality of the predicted activity sequences. These measures are capable of measuring artifacts such as activity fragmentation, merging, and transitions as shown in Figure 5.4. These measures provide deeper insight into the quality of the predicted sequences. For example, *fragmentation* and *substitution* can be used to evaluate the level of miss classifications within a class. Meanwhile, the correct prediction of the class boundaries can be demonstrated by *underfill* and *overfill*.

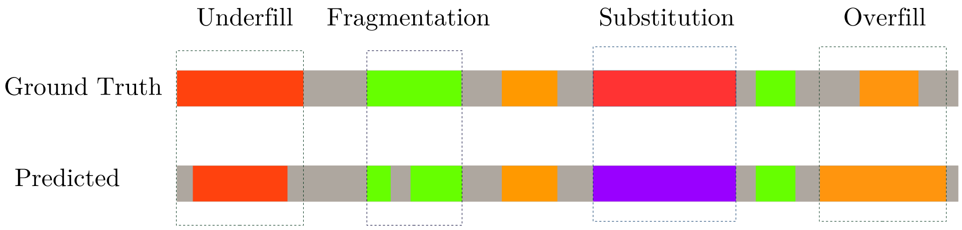


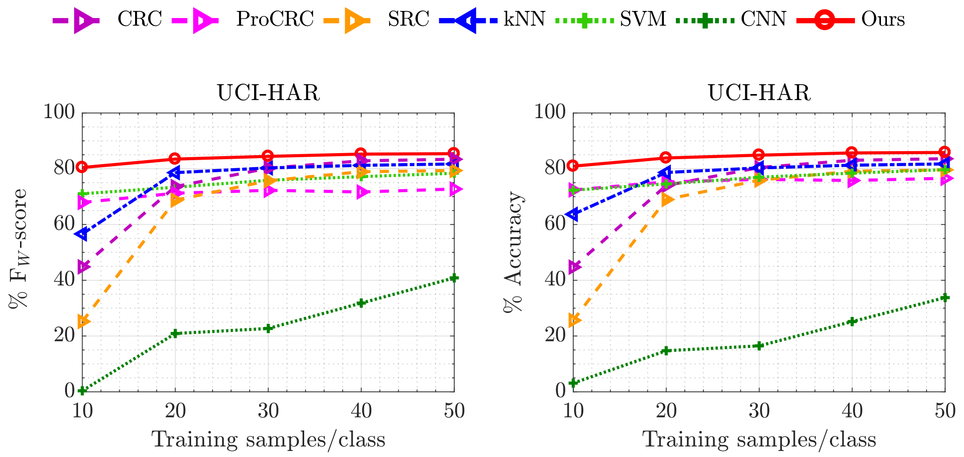
FIGURE 5.4: Demonstration of different types of misalignments. Different colors represent different class of activities. Four activity-based misalignments are illustrated here: *Fragmentation*, *Overfill*, *Underfill*, and *Substitution*. The first and the second row show the ground truth and the predicted activity labels obtained from a classifier. Gray denotes the interruptions from other classes.

Since the two datasets do not have a *Null* class, we specifically employ the following activity misalignment measures: (i) *Fragmentation* denotes the error of predicting a wrong class in between an uninterrupted activity class; (ii) *Overfill/underfill* indicates the error when the start and stop of the predicted sequence is earlier/later than the actual time; (iii) *Substitution* represents the error when an activity is misclassified as a different class. Along with these measurements, we report *True Positives* (TP) and *True Negatives* (TN) in relation to the ground truth class labels.

5.6.5 Sample-based classification performance

In this section, we evaluate the performance of the proposed method with the standard sample-based classification measures. Figure 5.5 and 5.6 provide the overall classification performance across different number of training samples of the UCI-HAR and Hospital datasets, respectively. The classification performance is shown in F_W-score and recognition accuracy. Clearly, our approach yields the best performance in most cases, which is more pronounced for the UCI-HAD dataset than the Hospital dataset. This is because the performance from all the competitors drops obviously in the UCI-HAD dataset since the training samples in the UCI-HAD dataset are more correlated, than those of the Hospital dataset. Furthermore, our approach consistently offers an F_W-score of over 80%, although the training samples are as small as 10 samples per class. This indicates that our approach is more robust than the other algorithms.

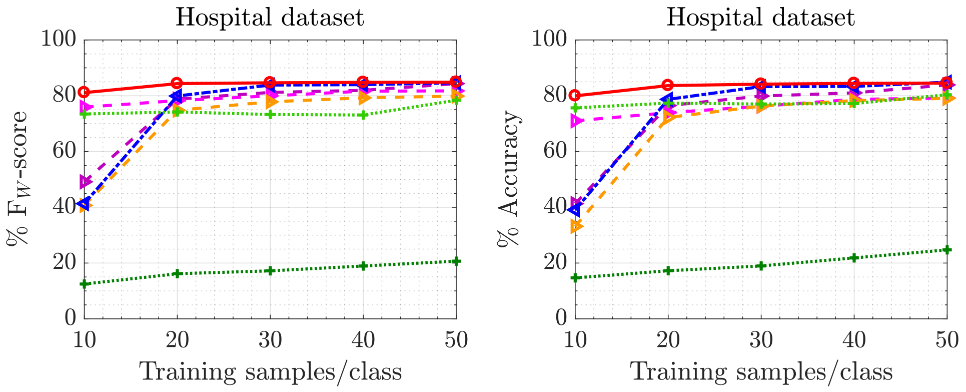
Figure 5.7 and 5.8 demonstrates class-specific activity recognition results for the UCI-HAD and the Hospital datasets. For each dataset, we provide the result at



(A) Weighted F_1-score

(B) Recognition accuracy

FIGURE 5.5: Sample-based classification performance, i.e. weighted F1-score and recognition accuracy, across different numbers of training samples on UCI-HAR.

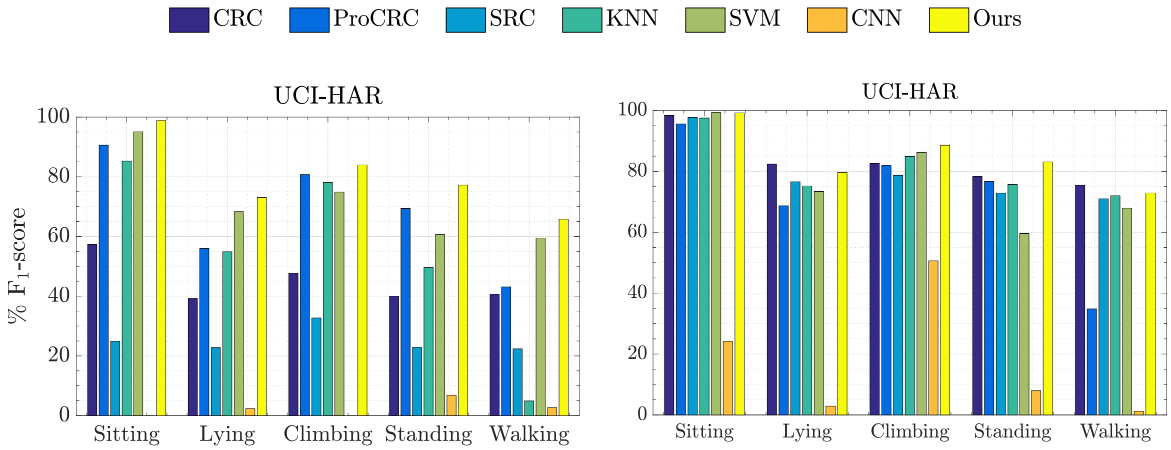


(A) Weighted F_1-score

(B) Recognition accuracy

FIGURE 5.6: Sample-based classification performance, i.e. weighted F1-score and recognition accuracy, across different numbers of training samples on Hospital dataset.

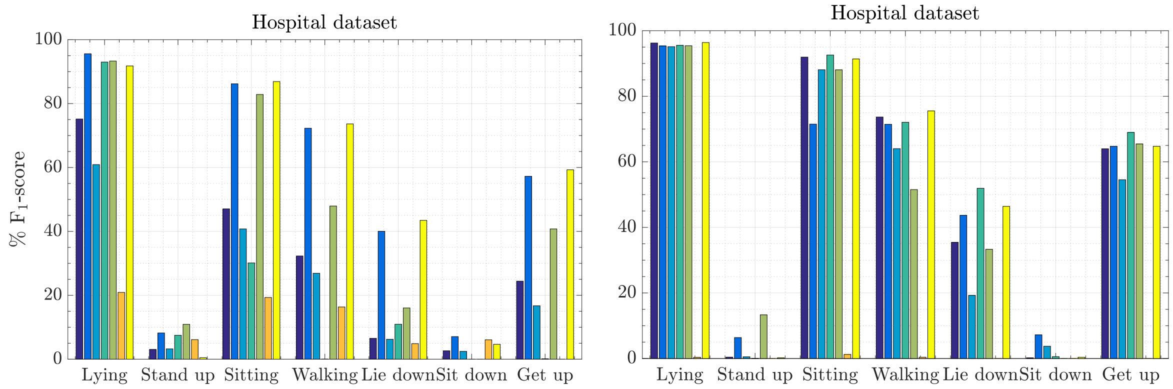
the two extremes : (A) when there are 10 and (B) 50 training samples per class. In most cases, almost every activity benefits from our adaptive-MRF prior that improve discriminative power in the representation vector recovery. When the number of training samples per class is small (ten samples per class in particular), the adaptive-MRF prior offers a clear performance improvement over CRC, as well as other methods in recognizing activity classes whose training samples are highly correlated to one another (e.g. *Sitting*, *Lying*, and *Standing* in the UCI-HAR) . In such case, our method produces the largest information gains. For example, the F_1-score is 98%, 73%, and 77% for *Sitting*, *Lying*, and *Standing* when only 10×5 training samples



(A) 10 Training samples per class

(B) 50 Training samples per class

FIGURE 5.7: UCI-HAR dataset. Class-specific recognition performance by F_1-score across different classes.



(A) 10 Training samples per class

(B) 50 Training samples per class

FIGURE 5.8: Hospital dataset. Class-specific recognition performance measured by F_1-score across different classes.

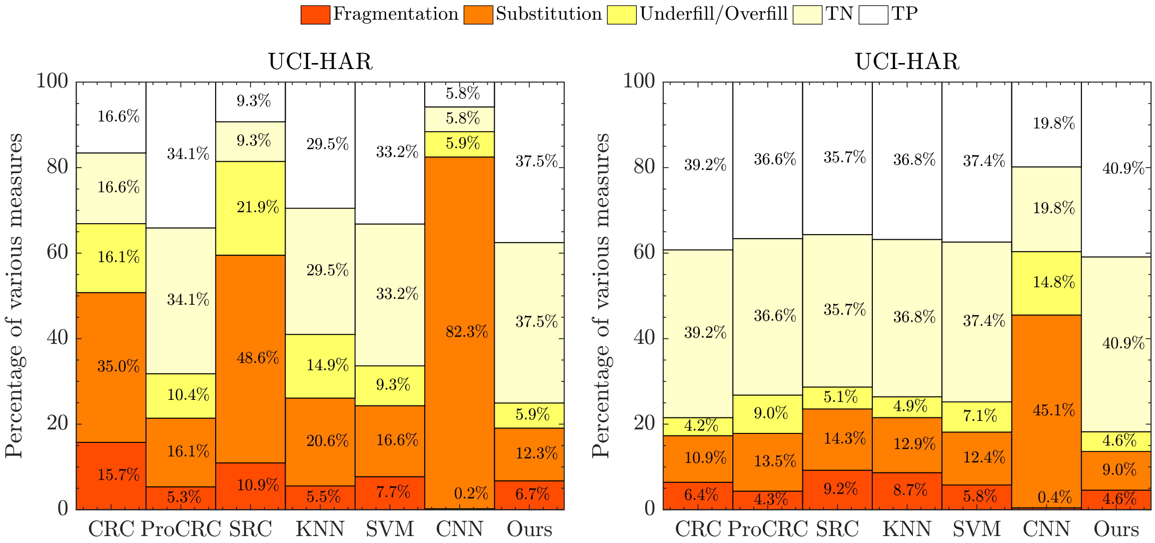
from UCI-HAR are used. This suggests that the adaptive-MRF prior is crucial for improving discriminative power against ambiguity due to the correlation in training samples. Finally, this improves the ultimate classification performance when the number of training samples are small. We observe that adaptive-MRF prior helps improve the overall classification performance consistently across different number of training samples (see Figure 5.5 and 5.6). This suggests that the adaptive-MRF prior helps improve the overall classification performance especially when the number of training samples is small. When the training samples per class increases to fifty, many algorithms improve their performance to a level that is comparable (though still inferior) to our adaptive-MRF prior.

5.6.6 Activity-based misalignment performance

In this section, we demonstrate the performance in the predicted sequences of human activities by using the activity-based misalignment measures. Figure 5.9 and 5.10 demonstrates the activity-based misalignment performance on UCI-HAR and Hospital dataset, respectively. For each dataset, we provide the result at the two extreme settings—(A) when there are 10 training samples per class and (B) 50 training samples per class are used. Clearly, our approach achieves the highest percentage of TN and TP, while providing the lowest percentage of artifacts—*Fragmentation*, *Substitution*, and *Overfill/underfill* in both datasets and with all cases of the number of training samples. Its high performance is more obvious when the number of training samples is extremely small—10 samples per class—and when the training samples are highly correlated. In such cases, for example, our approach is 6.9% better in terms of artifacts (*Fragmentation*, *Substitution*, and *Overfill/underfill*), and 3.5%$\times$2 better in $TN + TP$ than ProCRC which is the second best performing CRC-based approach in the UCI-HAR. Our approach also yield the best results in the Hospital dataset. Our approach offers 0.7% better in terms of artifacts (*Fragmentation*, *Substitution*, and *Overfill/underfill*), and 0.3%$\times$2 better in $TN + TP$ than SVM which is the second best performing classifier in the Hospital dataset. This consistent misalignment performance verifies that our proposed classification can provide high quality in predicting sequences of human activities when the amount of training samples is limited.

5.6.7 Runtime performance

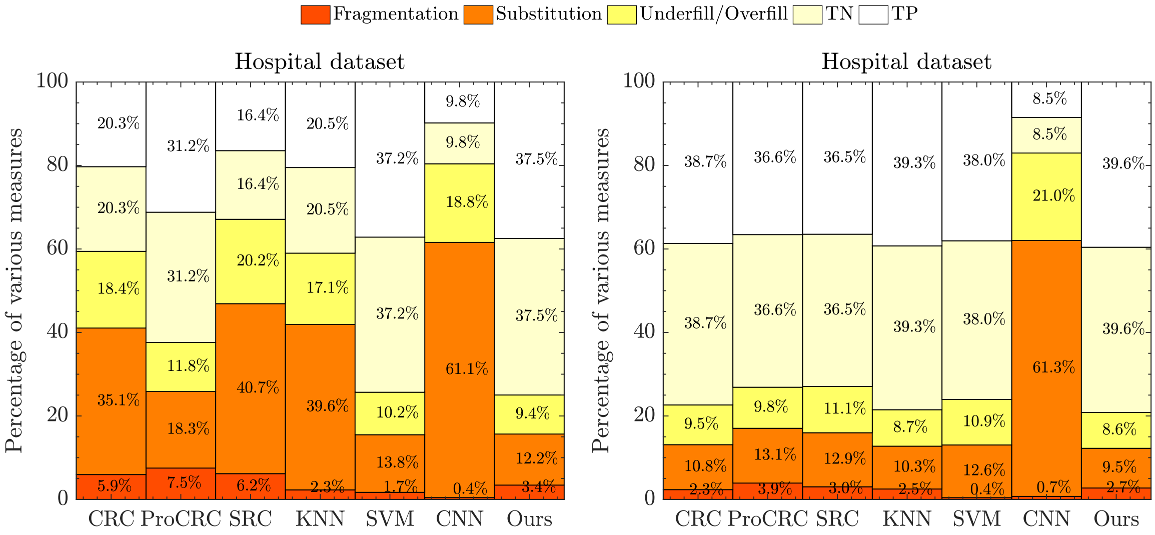
To show the efficiency of the proposed method, we report the total runtime across different numbers of training samples. The number of training samples is associated with N, which contributes to most of the computational cost (see Section 5.5 for the algorithm complexity). Figure 5.11 demonstrates the average runtime for each query sample where the runtime of all the candidate classification approach is compared across different numbers of training samples. It can be seen that the curve of our



(A) 10 training samples per class

(B) 50 training samples per class

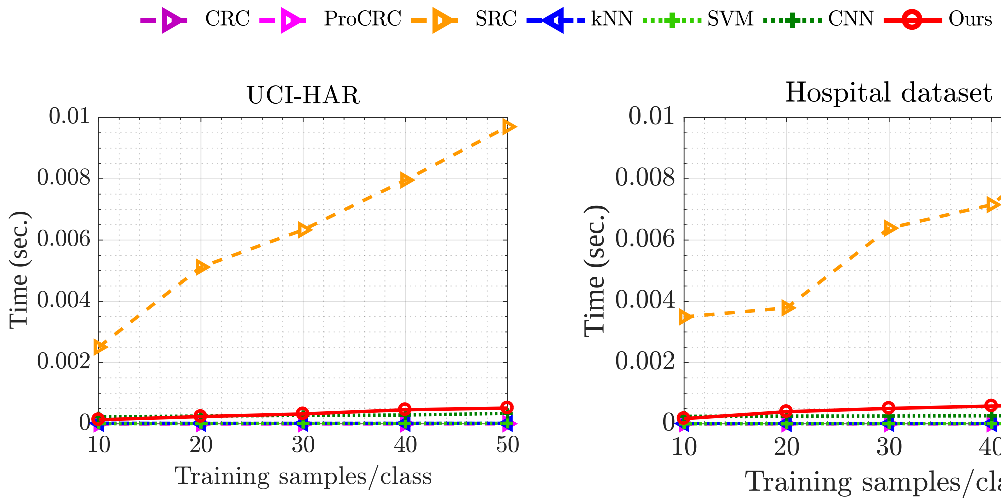
FIGURE 5.9: Activity-based misalignment performance on the UCI-HAR dataset at two extreme numbers of training samples, i.e. 10 and 50 training samples per class.



(A) 10 Training samples per class

(B) 50 Training samples per class

FIGURE 5.10: Activity-based misalignment performance on the Hospital dataset at two extreme numbers of training samples, i.e. 10 and 50 training samples per class.



(A) UCL-HAR dataset

(B) Hospital dataset

FIGURE 5.11: Runtime performance across different numbers of training samples on the UCI-HAR dataset and the Hospital dataset. The size of features is set to 50.

total runtime is moderate and aligned with the other algorithm. Moreover, it is stable across different numbers of training samples (under 0.0002 seconds). The runtime of our proposed approach is much less (one tenth of SRC's) than the one of SRC that is over ≥ 0.002 seconds. Therefore, the runtime performance result suggests that the runtime of our algorithm can be affordable in practice.

5.7 Summary

In this section, we have presented a new graphical-based classification that improves the robustness of the collaborative representation-based classification. We propose to employ the adaptive MRF to capture the underlying structure of the representation vectors from a query sample. The underlying structure offers the additional information that is related to the class of the corresponding query sample, which helps improve discriminative power. The adaptive MRF can be customized according to the partition of the training samples to further reduce the ambiguity due to correlated training samples. We apply the One-step-Adaptive MRF to efficiently estimate the MRF parameters from the given query. Extensive experiments on the two real-world datasets demonstrates the promising classification performance of the proposed classification method.

Chapter 6

Conclusion

6.1 Summary

Compressive sensing (CS) is an advanced signal sensing technique that acquires high dimensional signals from a few measurements. Recent research in CS attempts to further reduce the number of measurements by employing signal structures. In this thesis, we propose a novel structured sparsity model, *adaptive Markov random field (MRF)*, that has two desirable properties: (i) flexibility—the ability to represent a wide range of signal structures and (ii)adaptability—being able to adapt for any signal structures. However, most of the existing methods are only able to achieve one of these two properties. The existing MRF-based methods inherit flexibility from a learned MRF, but cannot adapt for a new signal structure. Meanwhile, the data-adaptive models are able to adapt their model parameters, but they assume limited signal structures. Hence, the main contribution of this thesis is the novel and efficient recovery methods for CS.

In Chapter 3, we proposed an adaptive MRF and developed a Two-step-Adaptive MRF that leverages the adaptability of the MRF by adjusting the parameters and the underlying graph of the MRF according to the given measurements. To realize adaptability, the MRF parameters are estimated based on the point estimate of the latent sparse signals. Then, the sparse signal is estimated using the resulting MRF as the prior. To maximize adaptability, we also propose a new sparse signal estimation method to jointly and recursively estimate the sparse signal, support, and noise parameters. Extensive experiments on three real-world datasets demonstrate the promising results of the proposed adaptive-MRF-based method. The point estimation of sparse signals underpins the performance of the MRF parameters estimation for the Two-step-Adaptive MRF. However, the point estimation cannot depict the statistical uncertainty of the latent sparse signals.

In Chapter 4, we proposed a One-step-Adaptive MRF that considers the statistical uncertainty of the latent sparse signals. We reformulate the MRF parameter estimation into a maximum marginal likelihood problem that solves for the MRF parameters directly from the measurements. The marginal likelihood is obtained from averaging all over the latent sparse signal population; thus, it offers better generalization over the Two-step-Adaptive MRF. Experiments on three real-world image datasets demonstrate the superior performance of the One-step-Adaptive MRF than the Two-step-Adaptive MRF and the state-of-the-art methods.

The adaptive MRF, as well as the One-step-Adaptive MRF, have significantly improved the performance of sparse signal recovery. They can be applied to many applications related to sparse signal recovery to exploit the underlying structure of the latent sparse representation as a prior in sparse signal recovery. Among many applications, collaborative-representation based classifications (CRCs) can directly benefit from the One-step-Adaptive MRF to extract the underlying structure directly from the query sample which can be a good indicator of the class label.

In Chapter 5, we apply adaptive MRF to improve the robustness of the CRCs in wearable sensors-based human activity recognition, when the training samples are limited. Most of the existing methods are based on the shortest Euclidean distance from a query sample to the training samples, which, however, can be susceptible to noise and correlation in the training samples. To address this problem, we proposed to extract the underlying structure of representation vector from the query sample which can be a good indicative of the class label; thus, this helps improve the discriminative power of the classifier. To reduce the ambiguity due to the correlated training samples, the adaptive MRF can be customized according to the training samples to reduce the correlation between different classes. The proposed One-step-Adaptive MRF is applied to extract the underlying structure and capture it with the adaptive MRF. Experiments on two real-world datasets demonstrate the promising performance of the Adaptive-MRF-based classification.

6.2 Future research directions

Based on our contributions and motivation, we see the following potential directions to develop new research fields:

6.2.1 Adaptive and deep structured sparsity model

Our thesis has demonstrated the benefits of employing adaptive Markov random fields (MRF) as the structured sparsity model to improve the performance of sparse signal recovery in compressive sensing (CS). Due to the flexibility and adaptability of the proposed adaptive MRF, our adaptive MRF can capture and adapt to various types of signal structures. Deep neural networks [148] have been a powerful tool to capture the underlying structure in various types of data with deep representation in many applications such as image denoising [149], image classification [148], and image super-resolution [150]. The deep neural networks such as stacking denoising auto-encoder [60], deep residual network [61], and convolutional neural network [62] have been applied to compressive sensing. However, the performance of these works are still constrained by the information in the training data. Therefore, we suggest a new research direction, which is to integrate the deep neural network architectures into the adaptive structured CS to create a new structured sparsity model that is adaptive and flexible with higher order representation of the neural network. To achieve this goal, one may have to address two important questions: (i) how to adaptively updating the structure and parameters of a deep neural network given a few compressed measurements and (ii) how to keep the computational cost low. One may consider updating only some parts of the neural network structure and parameters according to the given measurements. However, many research questions still open for further investigation; for example, how to decide at what condition the neural network should be updated for the new signal structure? Also, to adaptively estimate only some parts of the neural network, what parts or layers of the deep neural network should be fixed and what parts should be updated? Alternatively, to keep the computational cost low at the testing phase, one may consider using transfer learning [151] to adapt the deep structured sparsity model offline.

6.2.2 Adaptive-MRF based CS for large size images

Although the computational complexity and runtime of the proposed Two-step and One-step Adaptive-MRF in Chapter 3 and 4 are lower than many existing MRF-based approaches, their computational costs are still deemed too high for recovering a large-size image. To improve the proposed Two-step-Adaptive MRF, new techniques may be developed to improve the MRF parameter estimation and the sparse signal estimation process which involves the support estimation as well as the sparse signal recovery. To improve MRF parameter estimation and graphical model inference, one may consider techniques that are developed for large scale problem such as [152] and [153], respectively. For the sparse signal recovery, one can resort to employing a generalized approximate message passing (GAMP) [154] that is able to adaptively and recursively estimate the sparse signal, support, noise and signal parameters with low computational cost, which has shown to be very efficient for large-scale problems. The method in [154] can also be employed to improve the scalability of the proposed One-step Adaptive-MRF. However, the approach in [154] employs the Bernoulli-Gaussian model as the structured sparsity prior in signal recovery. The Bernoulli-Gaussian model is a special case of our MRF. To employ this signal recovery framework and use it with an adaptive MRF, a new approximation for the posterior distribution of support given measurements and new equations for updating the MRF parameters need to be derived. With lower complexity, our proposed methods can be useful for many CS applications involved with high dimensional signals.

6.2.3 Volumetric (3D) structure

Most of the existing MRF methods and the proposed adaptive MRF can be exploited to capture the structure of 1D signals and 2D images because the highest order of the MRFs is pairwise. One may consider extending our work to higher order MRFs [155], [156] can be more flexible to model the relationship between voxels which can be useful for many high level 3D imaging applications, such as integral imaging, holography, 3D scene reconstruction and 3D object recognition. This direction can be an alternative to realize an adaptive and deep structured sparsity model, instead of using the deep neural networks.

The main contribution of this thesis is the proposed adaptive MRF and the adaptive-MRF-based structured CS algorithms. Our research effort demonstrates the potential to achieve high sparse signal recovery performance by extracting the underlying structure of the latent sparse signal structures from the measurements. This opens up a new direction to exploit signal structure and to further improve the adaptability of many existing powerful models for the ultimate performance.

Appendix A

Additional Visual results

A.1 Additional Results for Two-step-Adaptive MRF

In this section, we provide full visual results of the proposed Two-step-Adaptive MRF (TA-MRF) on three real-world datasets, i.e. MNIST, CMU-IDB, and CIFAR-10 images at the sampling rates of 0.3 and the noise level (SNR) of 30 dB. Here we provide the visual results from the reconstruction of sparse representation in handwritten digits in Figure A.1; the reconstruction of sparse representation in PCA, wavelet and DCT domains for CMU-IDB images are provided in Figure A.2, A.3, and A.4; and, the reconstruction of sparse representation in wavelet and DCT domains of CIFAR-10 images are provided in Figure A.5 and A.6. Our Two-step-Adaptive MRF can yield competitive reconstruction results over different sparse representation. Our Two-step-Adaptive MRF can provide the highest reconstruction quality across in most cases:

- *MNIST images.* The proposed One-step-Adaptive MRF provides the highest PSNR improvement of 2.99 dB over the second most competitive method on digit no. 2 in Figure A.1.
- *CMU-IDB images.* In the PCA domain, the proposed One-step-Adaptive MRF yields the highest PSNR improvement of 1.47 dB over the second most competitive method in the 2^{nd} and 8^{th} row from the top of Figure A.2. In the wavelet domain, the proposed method yields the highest PSNR improvement of 1.28 dB over the second most competitive method in the 1^{st} row from the top of Figure A.3. In the DCT domain, the proposed method yields the highest PSNR

improvement of 0.71 dB over the second most competitive method in the 9^{th} row from the top of Figure A.4.

- *CIFAR-10 images.* In the wavelet domain, the proposed One-step-Adaptive MRF yields the highest PSNR improvement of 2.91 dB over the second most competitive method in the 3^{rd} row from the top of Figure A.5. In the DCT domain, the proposed method provides the highest PSNR improvement of 1.62 dB over the second most competitive method in the 4^{th} row from the top of Figure A.6. For completeness, the reconstruction results of CIFAR-10 images in PCA domain are provided in Figure A.7 where all the algorithms provide poor results due to the lack of sparsity in the PCA signal representation.

In conclusion, these visual results (Figure A.1- Figure A.6) are consistent with the compressibility performance in Figure 3.11 in Chapter 3, where our proposed method offers the highest numerical result in most cases.

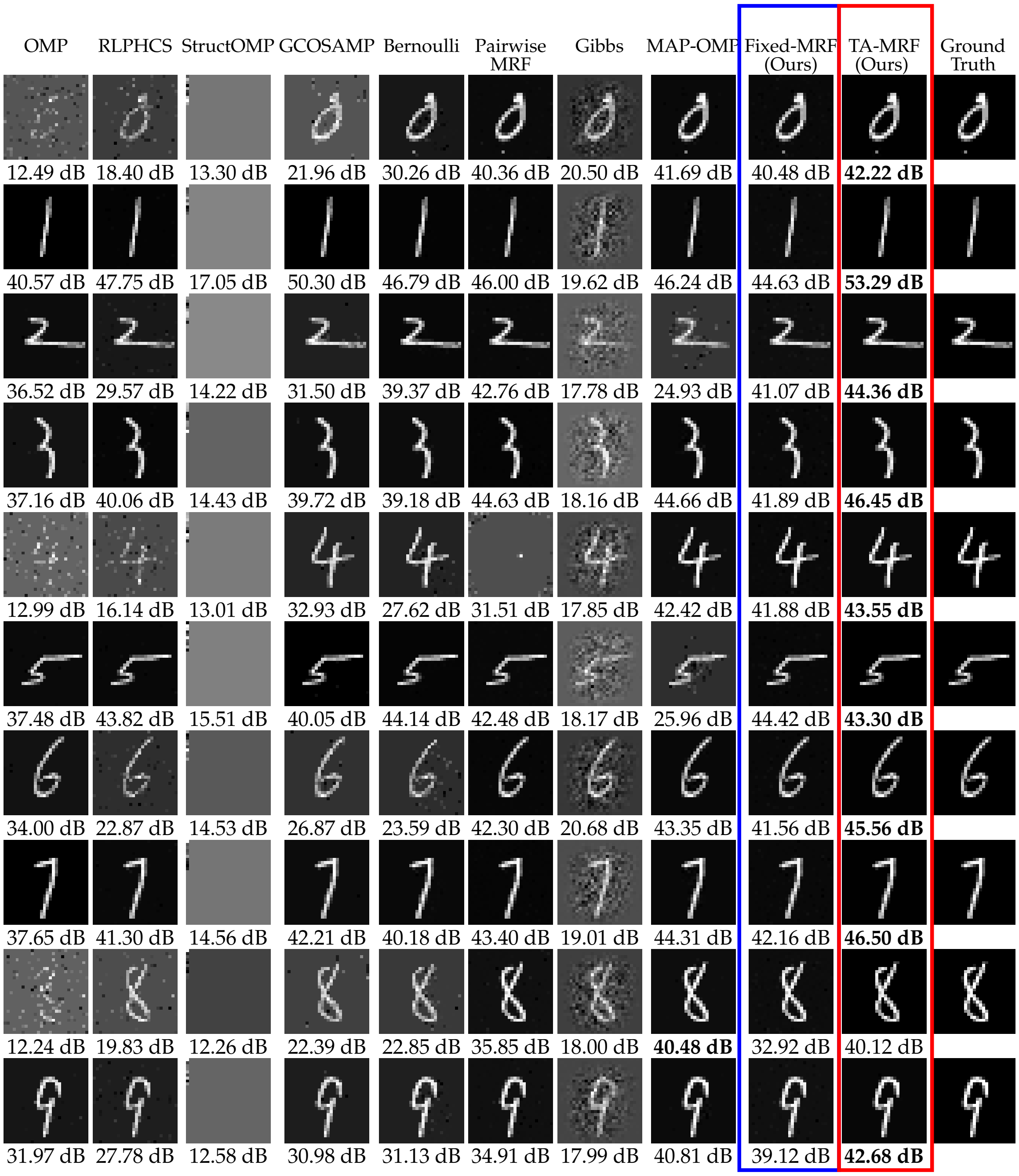


FIGURE A.1: Visual results of MNIST handwritten digit images (at $M/N = 0.3$, SNR = 30 dB).

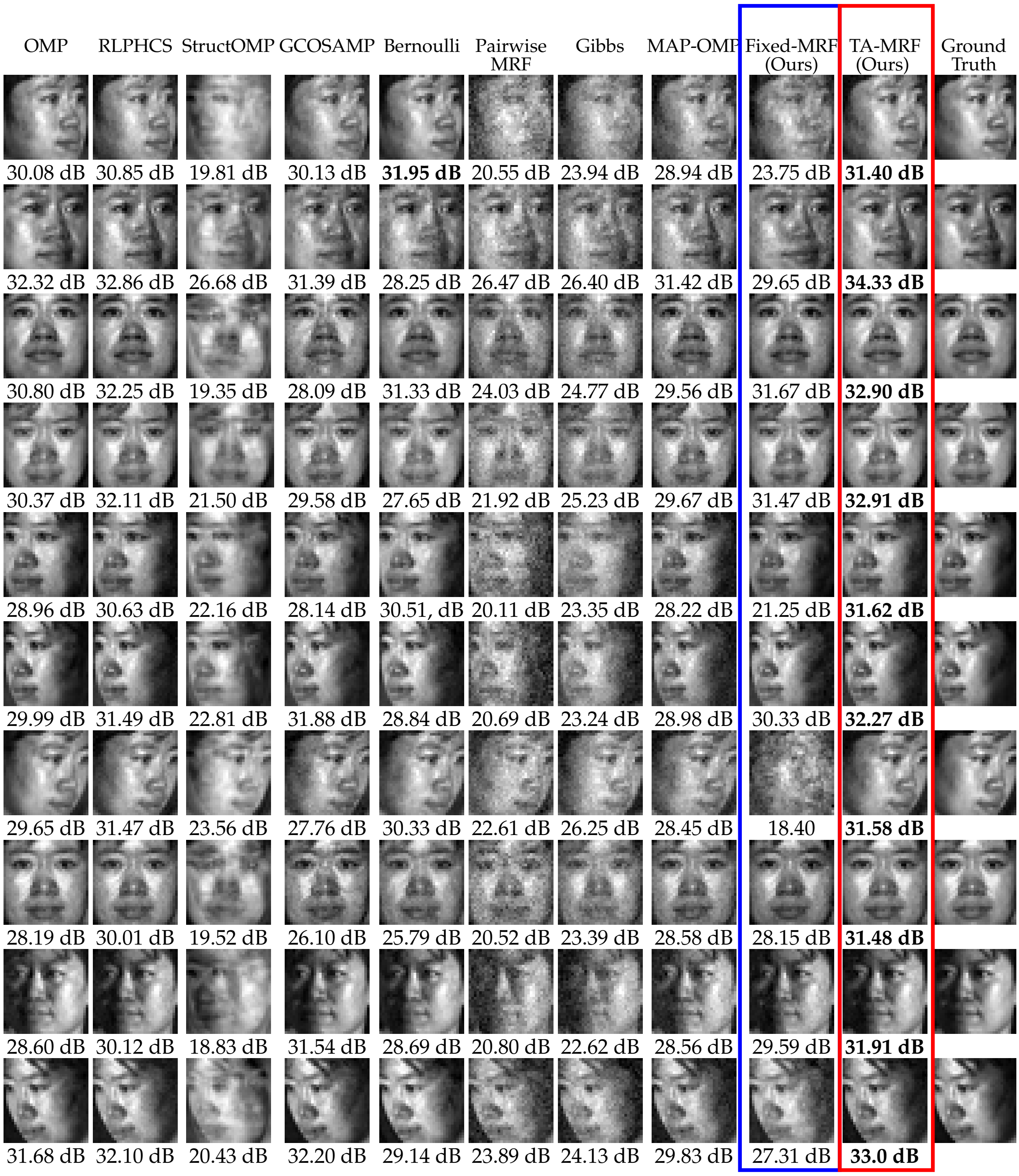

OMP
RLPHCS
StructOMP
GCOSAMP
Bernoulli
Pairwise MRF
Gibbs
MAP-OMP
Fixed-MRF (Ours)
TA-MRF (Ours)
Ground Truth
30.08 dB 30.85 dB 19.81 dB 30.13 dB **31.95 dB** 20.55 dB 23.94 dB 28.94 dB 23.75 dB **31.40 dB**
32.32 dB 32.86 dB 26.68 dB 31.39 dB 28.25 dB 26.47 dB 26.40 dB 31.42 dB 29.65 dB **34.33 dB**
30.80 dB 32.25 dB 19.35 dB 28.09 dB 31.33 dB 24.03 dB 24.77 dB 29.56 dB 31.67 dB **32.90 dB**
30.37 dB 32.11 dB 21.50 dB 29.58 dB 27.65 dB 21.92 dB 25.23 dB 29.67 dB 31.47 dB **32.91 dB**
28.96 dB 30.63 dB 22.16 dB 28.14 dB 30.51, dB 20.11 dB 23.35 dB 28.22 dB 21.25 dB **31.62 dB**
29.99 dB 31.49 dB 22.81 dB 31.88 dB 28.84 dB 20.69 dB 23.24 dB 28.98 dB 30.33 dB **32.27 dB**
29.65 dB 31.47 dB 23.56 dB 27.76 dB 30.33 dB 22.61 dB 26.25 dB 28.45 dB 18.40 **31.58 dB**
28.19 dB 30.01 dB 19.52 dB 26.10 dB 25.79 dB 20.52 dB 23.39 dB 28.58 dB 28.15 dB **31.48 dB**
28.60 dB 30.12 dB 18.83 dB 31.54 dB 28.69 dB 20.80 dB 22.62 dB 28.56 dB 29.59 dB **31.91 dB**
31.68 dB 32.10 dB 20.43 dB 32.20 dB 29.14 dB 23.89 dB 24.13 dB 29.83 dB 27.31 dB **33.0 dB**

FIGURE A.2: Visual results of CMU-IDB face images from PCA sparse signal reconstruction (at $M/N = 0.3$, SNR = 30 dB).

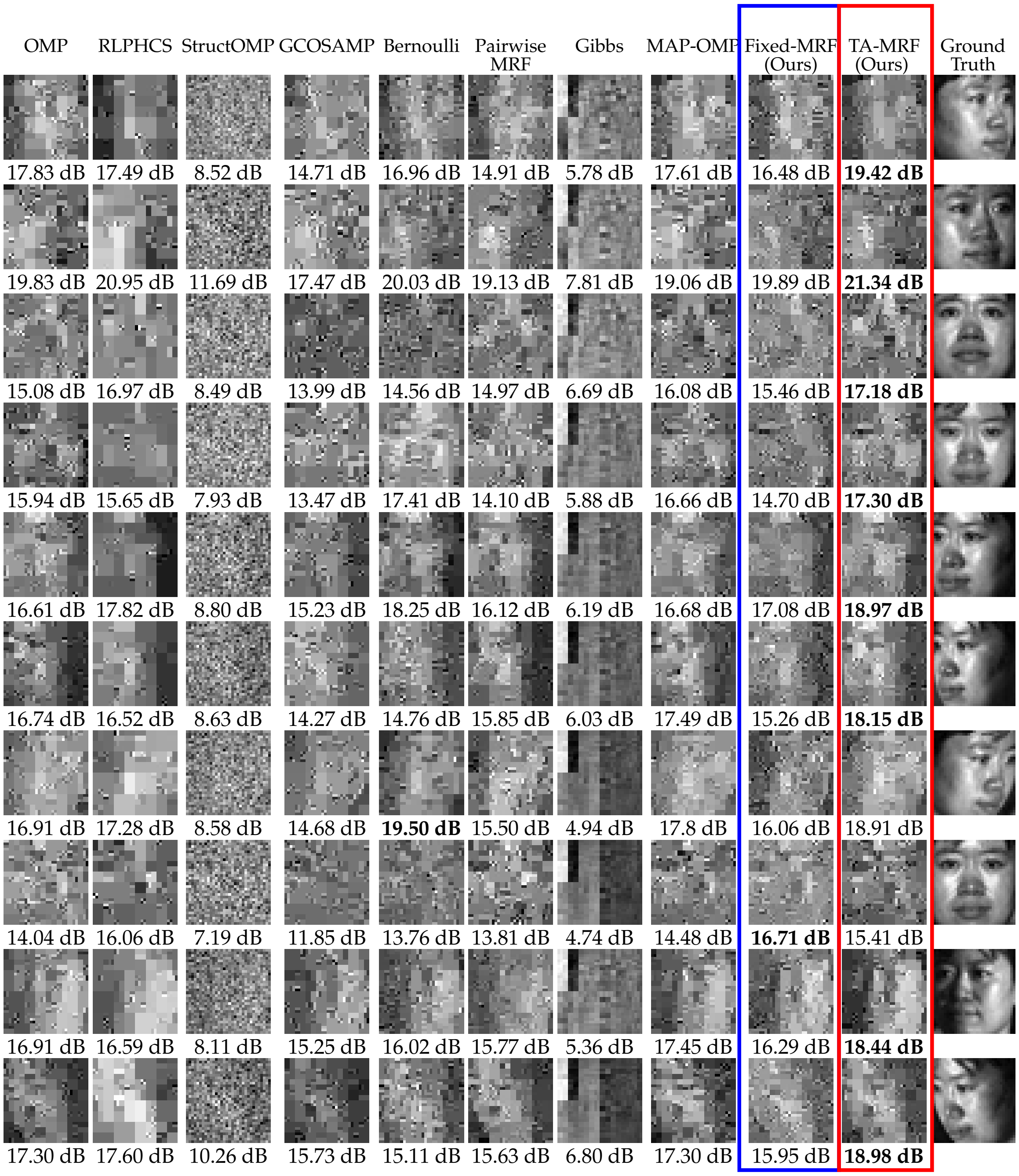

OMP
RLPHCS
StructOMP
GCOSAMP
Bernoulli
Pairwise MRF
Gibbs
MAP-OMP
Fixed-MRF (Ours)
TA-MRF (Ours)
Ground Truth
17.83 dB 17.49 dB 8.52 dB 14.71 dB 16.96 dB 14.91 dB 5.78 dB 17.61 dB 16.48 dB **19.42 dB**
19.83 dB 20.95 dB 11.69 dB 17.47 dB 20.03 dB 19.13 dB 7.81 dB 19.06 dB 19.89 dB **21.34 dB**
15.08 dB 16.97 dB 8.49 dB 13.99 dB 14.56 dB 14.97 dB 6.69 dB 16.08 dB 15.46 dB **17.18 dB**
15.94 dB 15.65 dB 7.93 dB 13.47 dB 17.41 dB 14.10 dB 5.88 dB 16.66 dB 14.70 dB **17.30 dB**
16.61 dB 17.82 dB 8.80 dB 15.23 dB 18.25 dB 16.12 dB 6.19 dB 16.68 dB 17.08 dB **18.97 dB**
16.74 dB 16.52 dB 8.63 dB 14.27 dB 14.76 dB 15.85 dB 6.03 dB 17.49 dB 15.26 dB **18.15 dB**
16.91 dB 17.28 dB 8.58 dB 14.68 dB **19.50 dB** 15.50 dB 4.94 dB 17.8 dB 16.06 dB 18.91 dB
14.04 dB 16.06 dB 7.19 dB 11.85 dB 13.76 dB 13.81 dB 4.74 dB 14.48 dB **16.71 dB** 15.41 dB
16.91 dB 16.59 dB 8.11 dB 15.25 dB 16.02 dB 15.77 dB 5.36 dB 17.45 dB 16.29 dB **18.44 dB**
17.30 dB 17.60 dB 10.26 dB 15.73 dB 15.11 dB 15.63 dB 6.80 dB 17.30 dB 15.95 dB **18.98 dB**

FIGURE A.3: Visual results of CMU-IDB face images from wavelet sparse signal reconstruction (at $M/N = 0.3$, SNR = 30 dB).

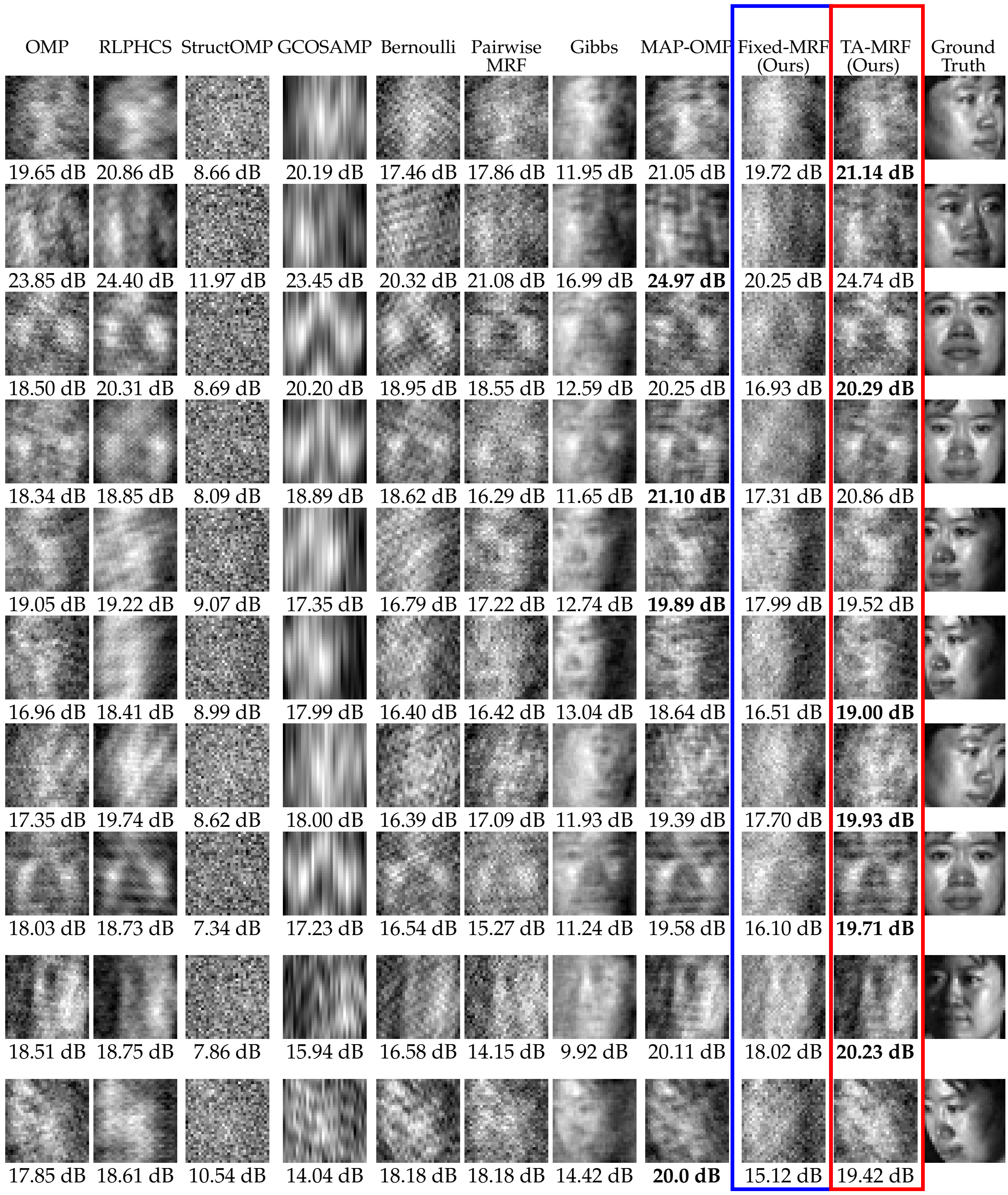

OMP
RLPHCS
StructOMP
GCOSAMP
Bernoulli
Pairwise MRF
Gibbs
MAP-OMP
Fixed-MRF (Ours)
TA-MRF (Ours)
Ground Truth
19.65 dB 20.86 dB 8.66 dB 20.19 dB 17.46 dB 17.86 dB 11.95 dB 21.05 dB 19.72 dB **21.14 dB**
23.85 dB 24.40 dB 11.97 dB 23.45 dB 20.32 dB 21.08 dB 16.99 dB **24.97 dB** 20.25 dB 24.74 dB
18.50 dB 20.31 dB 8.69 dB 20.20 dB 18.95 dB 18.55 dB 12.59 dB 20.25 dB 16.93 dB **20.29 dB**
18.34 dB 18.85 dB 8.09 dB 18.89 dB 18.62 dB 16.29 dB 11.65 dB **21.10 dB** 17.31 dB 20.86 dB
19.05 dB 19.22 dB 9.07 dB 17.35 dB 16.79 dB 17.22 dB 12.74 dB **19.89 dB** 17.99 dB 19.52 dB
16.96 dB 18.41 dB 8.99 dB 17.99 dB 16.40 dB 16.42 dB 13.04 dB 18.64 dB 16.51 dB **19.00 dB**
17.35 dB 19.74 dB 8.62 dB 18.00 dB 16.39 dB 17.09 dB 11.93 dB 19.39 dB 17.70 dB **19.93 dB**
18.03 dB 18.73 dB 7.34 dB 17.23 dB 16.54 dB 15.27 dB 11.24 dB 19.58 dB 16.10 dB **19.71 dB**
18.51 dB 18.75 dB 7.86 dB 15.94 dB 16.58 dB 14.15 dB 9.92 dB 20.11 dB 18.02 dB **20.23 dB**
17.85 dB 18.61 dB 10.54 dB 14.04 dB 18.18 dB 18.18 dB 14.42 dB **20.0 dB** 15.12 dB 19.42 dB

FIGURE A.4: Visual results of CMU-IDB face images from DCT sparse signal reconstruction (at $M/N = 0.3$, SNR = 30 dB).

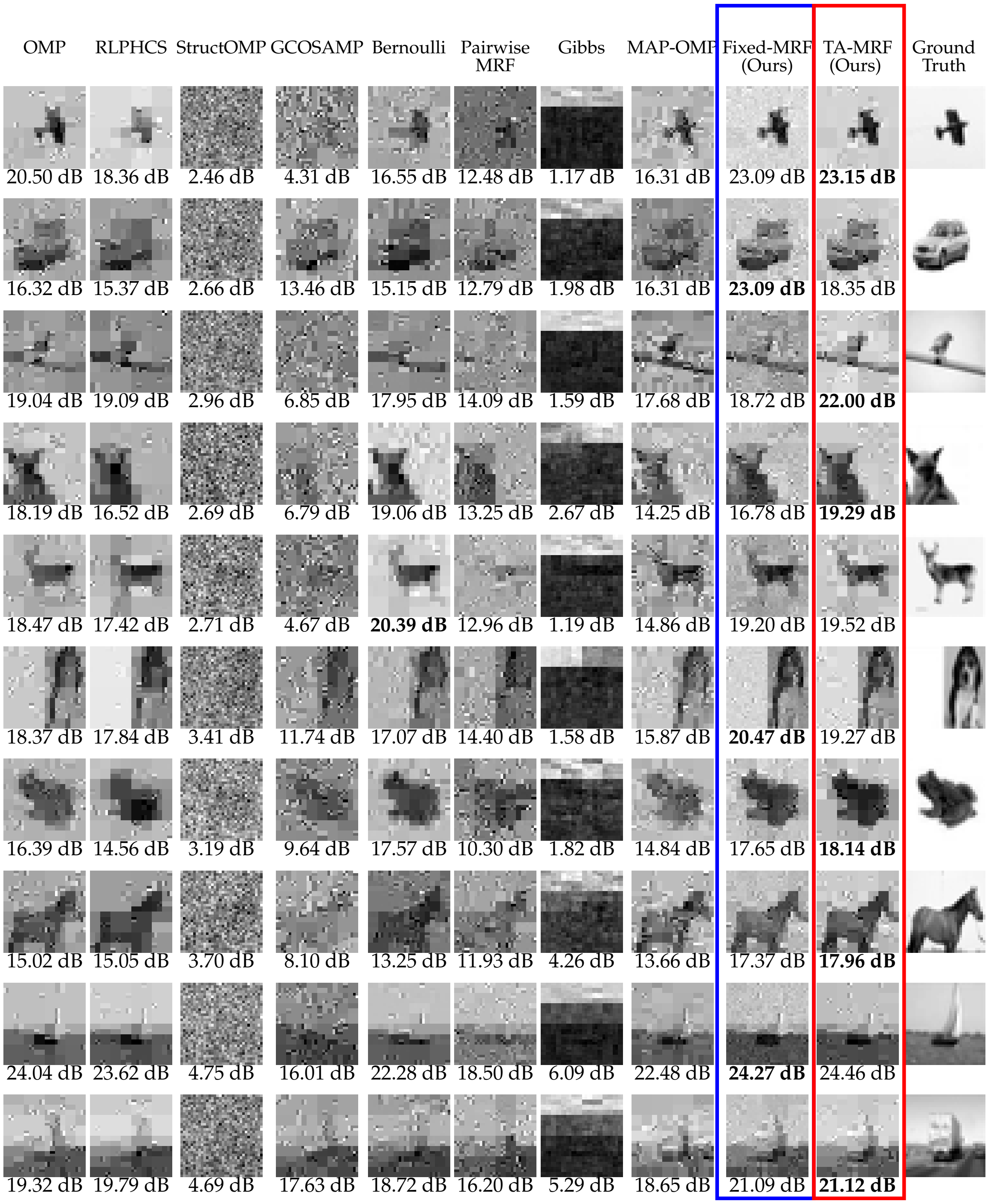

OMP
RLPHCS
StructOMP
GCOSAMP
Bernoulli
Pairwise MRF
Gibbs
MAP-OMP
Fixed-MRF (Ours)
TA-MRF (Ours)
Ground Truth
20.50 dB 18.36 dB 2.46 dB 4.31 dB 16.55 dB 12.48 dB 1.17 dB 16.31 dB 23.09 dB 23.15 dB
16.32 dB 15.37 dB 2.66 dB 13.46 dB 15.15 dB 12.79 dB 1.98 dB 16.31 dB 23.09 dB 18.35 dB
19.04 dB 19.09 dB 2.96 dB 6.85 dB 17.95 dB 14.09 dB 1.59 dB 17.68 dB 18.72 dB 22.00 dB
18.19 dB 16.52 dB 2.69 dB 6.79 dB 19.06 dB 13.25 dB 2.67 dB 14.25 dB 16.78 dB 19.29 dB
18.47 dB 17.42 dB 2.71 dB 4.67 dB 20.39 dB 12.96 dB 1.19 dB 14.86 dB 19.20 dB 19.52 dB
18.37 dB 17.84 dB 3.41 dB 11.74 dB 17.07 dB 14.40 dB 1.58 dB 15.87 dB 20.47 dB 19.27 dB
16.39 dB 14.56 dB 3.19 dB 9.64 dB 17.57 dB 10.30 dB 1.82 dB 14.84 dB 17.65 dB 18.14 dB
15.02 dB 15.05 dB 3.70 dB 8.10 dB 13.25 dB 11.93 dB 4.26 dB 13.66 dB 17.37 dB 17.96 dB
24.04 dB 23.62 dB 4.75 dB 16.01 dB 22.28 dB 18.50 dB 6.09 dB 22.48 dB 24.27 dB 24.46 dB
19.32 dB 19.79 dB 4.69 dB 17.63 dB 18.72 dB 16.20 dB 5.29 dB 18.65 dB 21.09 dB 21.12 dB

FIGURE A.5: Visual results of CIFAR-10 natural images from wavelet sparse signal reconstruction (at $M/N = 0.3$, SNR = 30 dB).

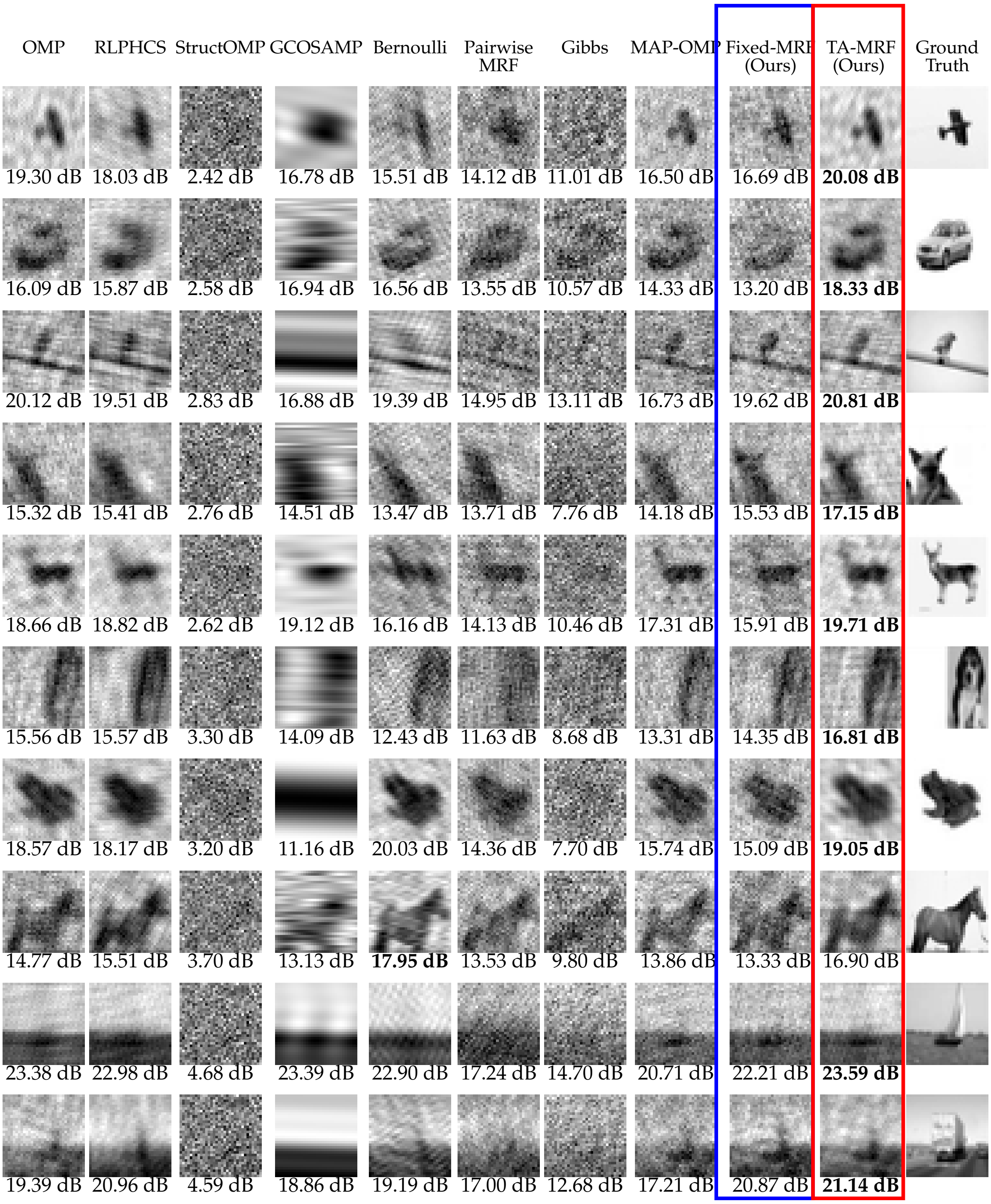


FIGURE A.6: Visual results of CIFAR-10 natural images from DCT sparse signal reconstruction (at $M/N = 0.3$, SNR = 30 dB).

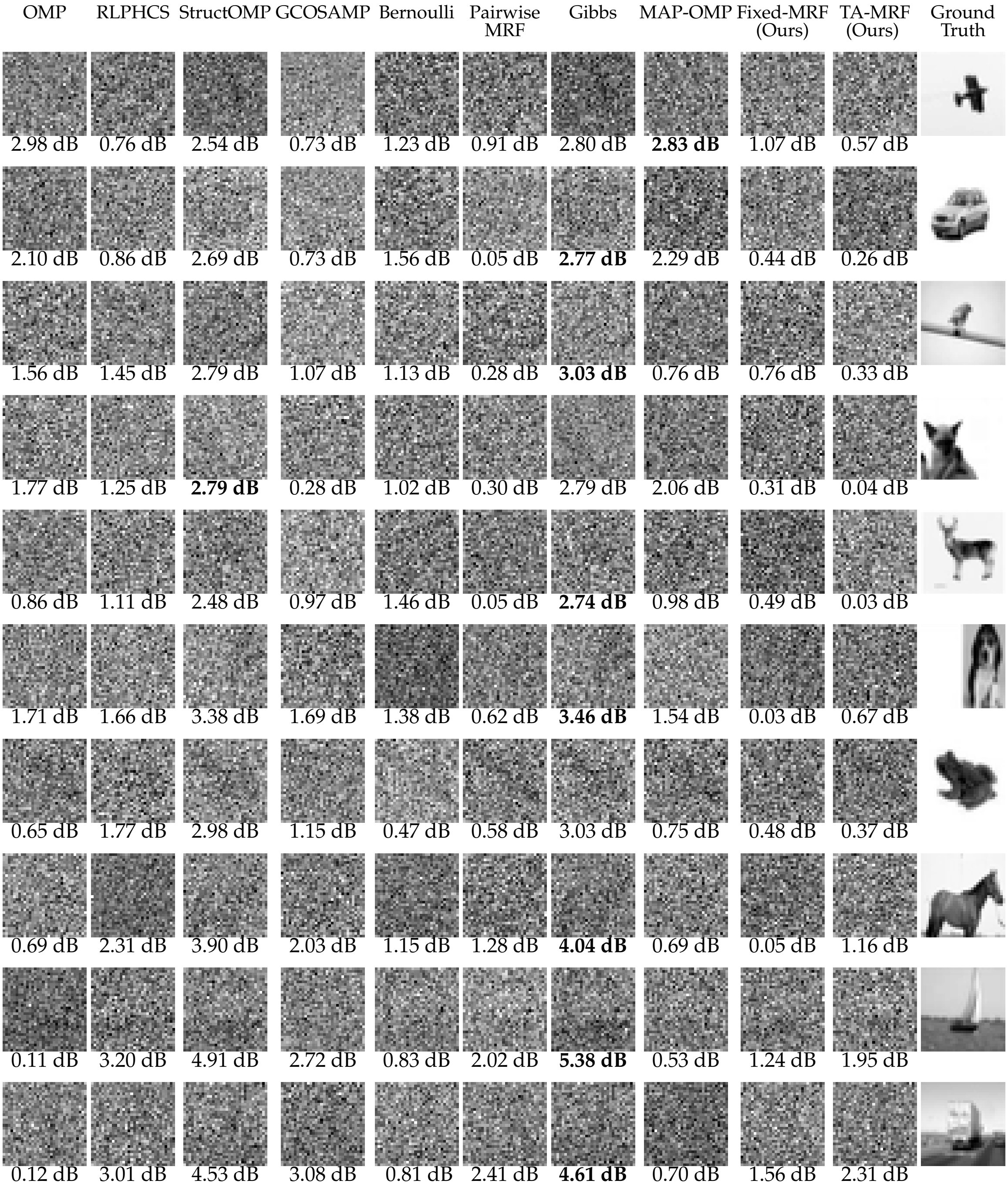


FIGURE A.7: Every algorithm fails in reconstruction of the CIFAR-10 natural images in the PCA domain (at $M/N = 0.3$, SNR = 30 dB).

A.2 Additional visual results for One-step-Adaptive MRF

In this section, we provide full visual results of the proposed One-step-Adaptive MRF (OA-MRF) on three real-world datasets, i.e. MNIST, CMU-IDB, and CIFAR-10 images at the sampling rate of 0.3 and the noise level (SNR) of 30 dB. Here, the visual results from the reconstruction of sparse representation of MNIST handwritten images are provided in Figure A.8; the visual results from the reconstruction of sparse representation in the PCA, wavelet, and DCT domains for CMU-IDB images are provided in Figure A.9, A.10, and A.11. The visual results from the reconstruction of sparse representation in the wavelet and DCT domains of CIFAR-10 images are provided in Figure A.12 and A.13. Our One-step-Adaptive MRF can yield competitive reconstruction results over different sparse representation. It can provide the highest reconstruction quality across in most cases:

- *MNIST images.* The proposed One-step-Adaptive MRF provides the highest PSNR improvement of 5.13 dB over the second most competitive method on digit no. 5 in Figure A.8.

- *CMU-IDB images.* In the PCA domain, the proposed One-step-Adaptive MRF yields the highest PSNR improvement of 2.4 dB over the second most competitive method in the 7^{th} row from the top of Figure A.9. In the wavelet domain, the proposed method yields the highest PSNR improvement of 2.54 dB over the second most competitive method in the 2^{nd} row from the top of Figure A.10. In the DCT domain, the proposed method yields the highest PSNR improvement of 2.15 dB over the second most competitive method in the 8^{th} row from the top of Figure A.11.

- *CIFAR-10 images.* In the wavelet domain, the proposed One-step-Adaptive MRF yields the highest PSNR improvement of 1.73 dB over the second most competitive method in the 10^{th} row from the top of Figure A.12. In the DCT domain, the proposed method provides the highest PSNR improvement of 2.50 dB over the second most competitive method in the 3^{rd} row from the top of Figure A.13. For completeness reasons, the reconstruction results of CIFAR-10 images in the PCA domain are provided in Figure A.14 where all the algorithms provide poor results due to the lack of sparsity in the PCA signal representation.

Therefore, these visual results (Figure A.8-A.13) are consistent with the compressibility performance in Figure 4.6 in Chapter 4, where the proposed One-step-Adaptive MRF offers the highest numerical result in most cases.

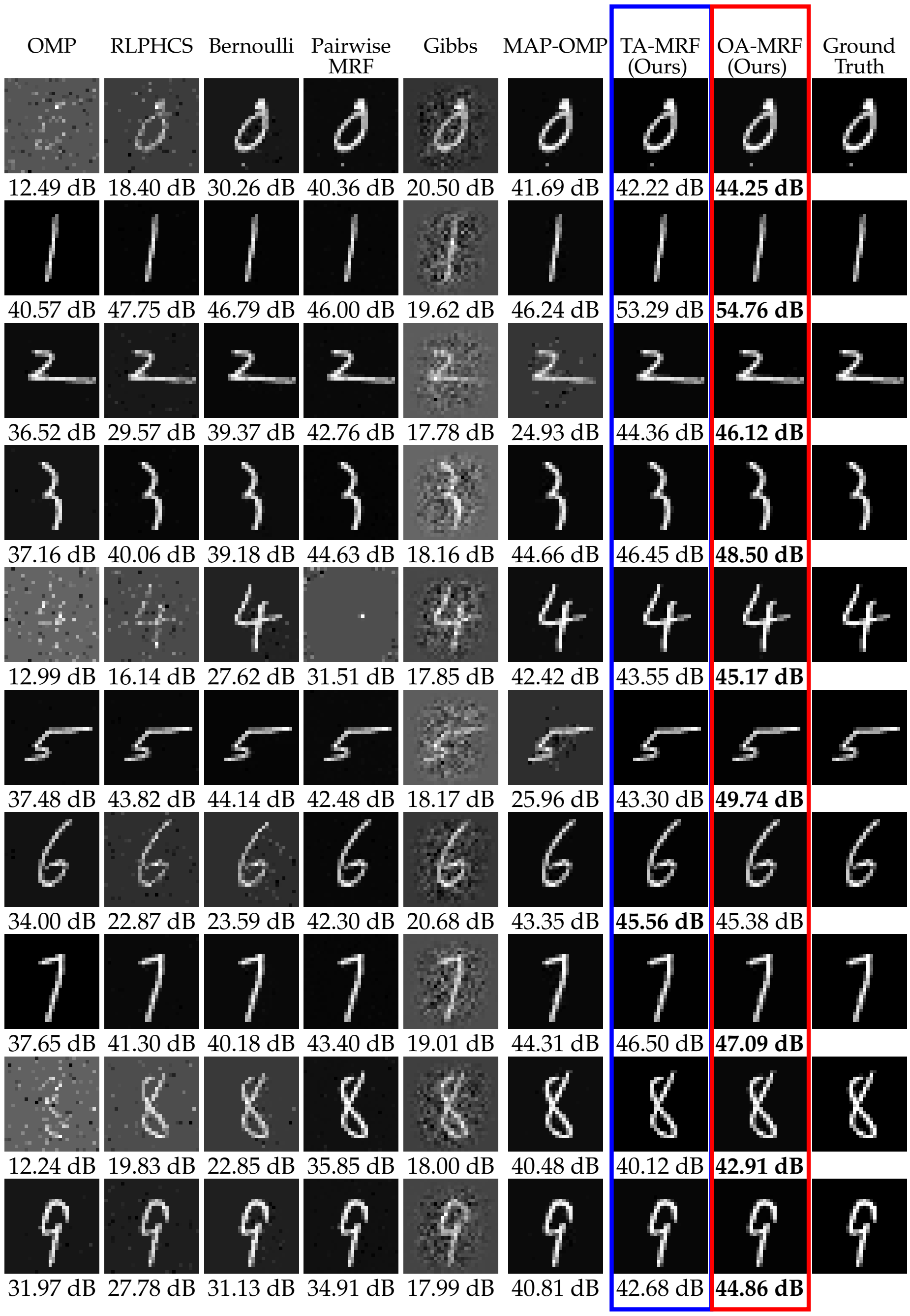


FIGURE A.8: Visual results of MNIST handwritten digit images (at $M/N = 0.3$, SNR = 30 dB).

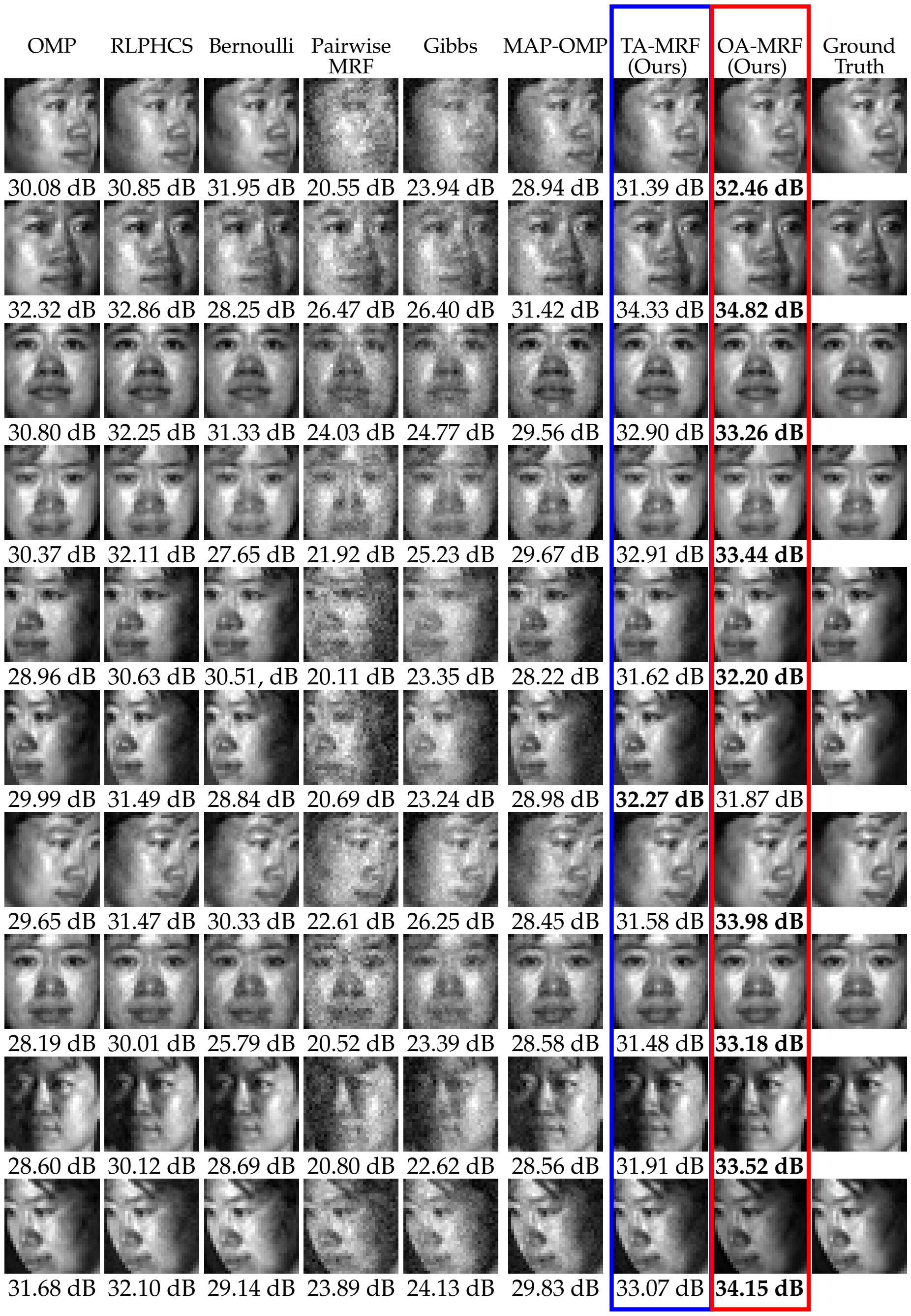

OMP
RLPHCS
Bernoulli
Pairwise MRF
Gibbs
MAP-OMP
TA-MRF (Ours)
OA-MRF (Ours)
Ground Truth
30.08 dB 30.85 dB 31.95 dB 20.55 dB 23.94 dB 28.94 dB 31.39 dB **32.46 dB**
32.32 dB 32.86 dB 28.25 dB 26.47 dB 26.40 dB 31.42 dB 34.33 dB **34.82 dB**
30.80 dB 32.25 dB 31.33 dB 24.03 dB 24.77 dB 29.56 dB 32.90 dB **33.26 dB**
30.37 dB 32.11 dB 27.65 dB 21.92 dB 25.23 dB 29.67 dB 32.91 dB **33.44 dB**
28.96 dB 30.63 dB 30.51, dB 20.11 dB 23.35 dB 28.22 dB 31.62 dB **32.20 dB**
29.99 dB 31.49 dB 28.84 dB 20.69 dB 23.24 dB 28.98 dB **32.27 dB** 31.87 dB
29.65 dB 31.47 dB 30.33 dB 22.61 dB 26.25 dB 28.45 dB 31.58 dB **33.98 dB**
28.19 dB 30.01 dB 25.79 dB 20.52 dB 23.39 dB 28.58 dB 31.48 dB **33.18 dB**
28.60 dB 30.12 dB 28.69 dB 20.80 dB 22.62 dB 28.56 dB 31.91 dB **33.52 dB**
31.68 dB 32.10 dB 29.14 dB 23.89 dB 24.13 dB 29.83 dB 33.07 dB **34.15 dB**

FIGURE A.9: Visual results of CMU-IDB face images from PCA sparse signal reconstruction (at $M/N = 0.3$, SNR = 30 dB).

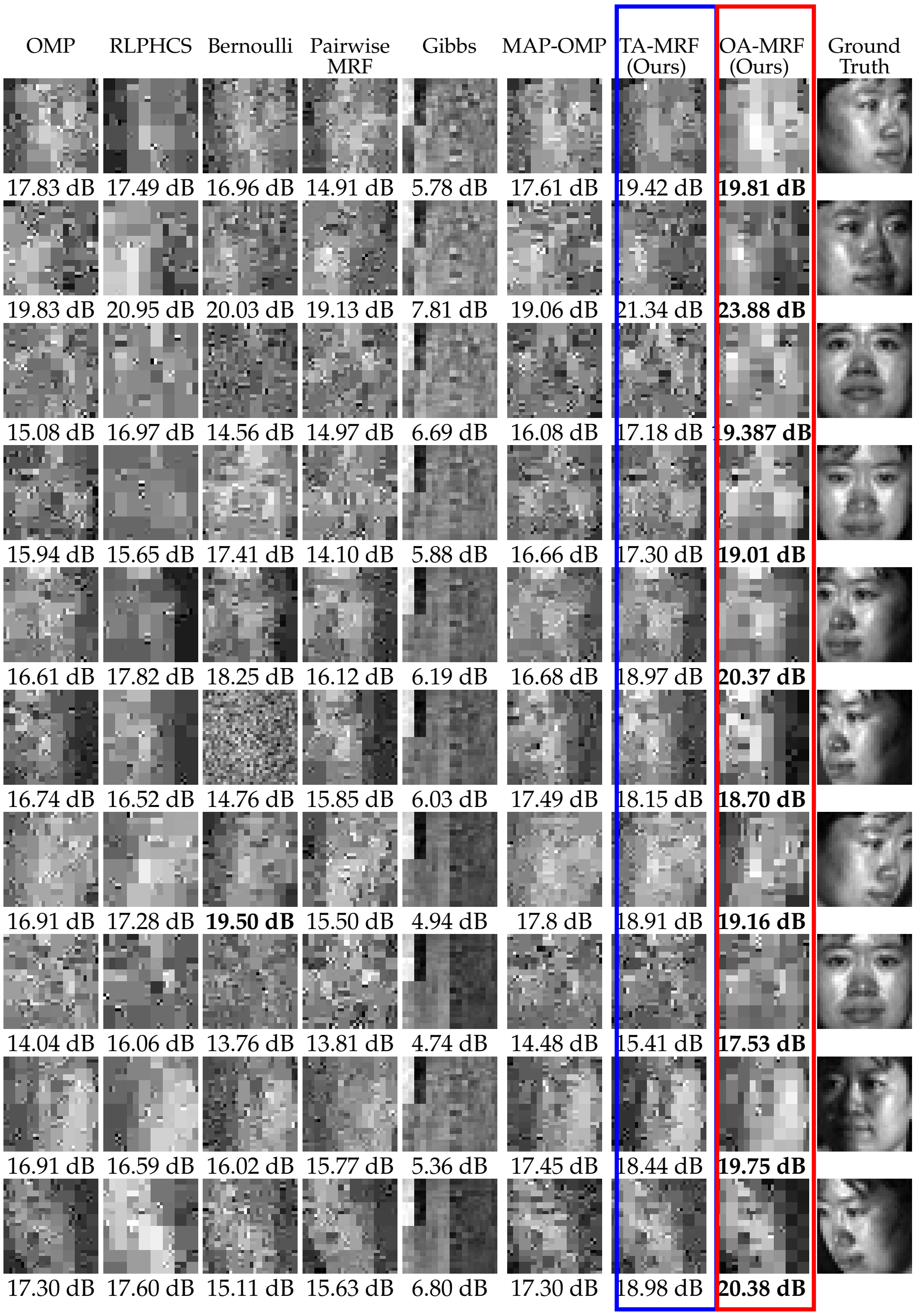

OMP
RLPHCS
Bernoulli
Pairwise MRF
Gibbs
MAP-OMP
TA-MRF (Ours)
OA-MRF (Ours)
Ground Truth
17.83 dB 17.49 dB 16.96 dB 14.91 dB 5.78 dB 17.61 dB 19.42 dB **19.81 dB**
19.83 dB 20.95 dB 20.03 dB 19.13 dB 7.81 dB 19.06 dB 21.34 dB **23.88 dB**
15.08 dB 16.97 dB 14.56 dB 14.97 dB 6.69 dB 16.08 dB 17.18 dB **19.387 dB**
15.94 dB 15.65 dB 17.41 dB 14.10 dB 5.88 dB 16.66 dB 17.30 dB **19.01 dB**
16.61 dB 17.82 dB 18.25 dB 16.12 dB 6.19 dB 16.68 dB 18.97 dB **20.37 dB**
16.74 dB 16.52 dB 14.76 dB 15.85 dB 6.03 dB 17.49 dB 18.15 dB **18.70 dB**
16.91 dB 17.28 dB **19.50 dB** 15.50 dB 4.94 dB 17.8 dB 18.91 dB **19.16 dB**
14.04 dB 16.06 dB 13.76 dB 13.81 dB 4.74 dB 14.48 dB 15.41 dB **17.53 dB**
16.91 dB 16.59 dB 16.02 dB 15.77 dB 5.36 dB 17.45 dB 18.44 dB **19.75 dB**
17.30 dB 17.60 dB 15.11 dB 15.63 dB 6.80 dB 17.30 dB 18.98 dB **20.38 dB**

FIGURE A.10: Visual results of CMU-IDB face images from wavelet sparse signal reconstruction (at $M/N = 0.3$, SNR = 30 dB).

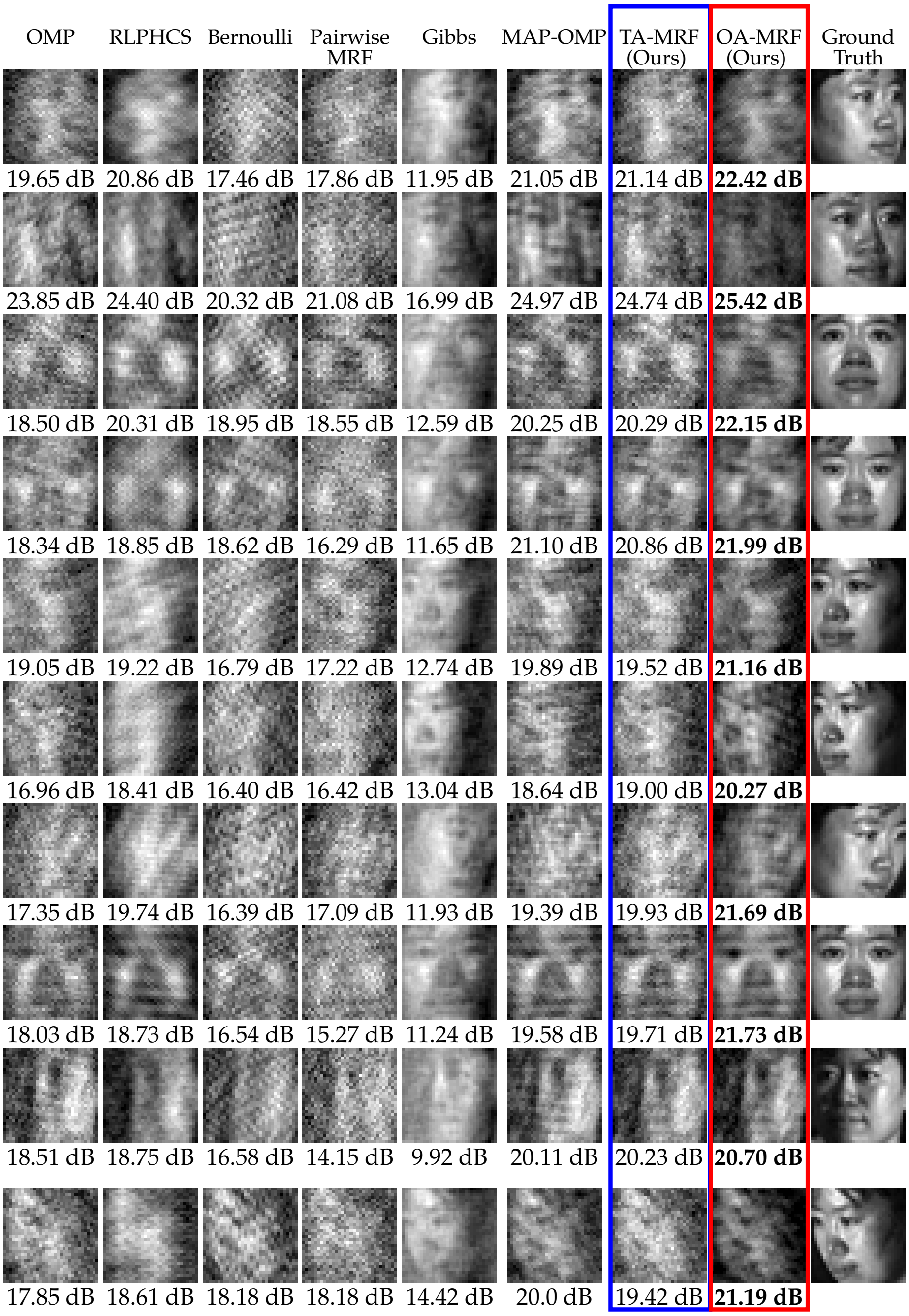

OMP
RLPHCS
Bernoulli
Pairwise MRF
Gibbs
MAP-OMP
TA-MRF (Ours)
OA-MRF (Ours)
Ground Truth
19.65 dB 20.86 dB 17.46 dB 17.86 dB 11.95 dB 21.05 dB 21.14 dB **22.42 dB**
23.85 dB 24.40 dB 20.32 dB 21.08 dB 16.99 dB 24.97 dB 24.74 dB **25.42 dB**
18.50 dB 20.31 dB 18.95 dB 18.55 dB 12.59 dB 20.25 dB 20.29 dB **22.15 dB**
18.34 dB 18.85 dB 18.62 dB 16.29 dB 11.65 dB 21.10 dB 20.86 dB **21.99 dB**
19.05 dB 19.22 dB 16.79 dB 17.22 dB 12.74 dB 19.89 dB 19.52 dB **21.16 dB**
16.96 dB 18.41 dB 16.40 dB 16.42 dB 13.04 dB 18.64 dB 19.00 dB **20.27 dB**
17.35 dB 19.74 dB 16.39 dB 17.09 dB 11.93 dB 19.39 dB 19.93 dB **21.69 dB**
18.03 dB 18.73 dB 16.54 dB 15.27 dB 11.24 dB 19.58 dB 19.71 dB **21.73 dB**
18.51 dB 18.75 dB 16.58 dB 14.15 dB 9.92 dB 20.11 dB 20.23 dB **20.70 dB**
17.85 dB 18.61 dB 18.18 dB 18.18 dB 14.42 dB 20.0 dB 19.42 dB **21.19 dB**

FIGURE A.11: Visual results of CMU-IDB face images from DCT sparse signal reconstruction (at $M/N = 0.3$, SNR = 30 dB).

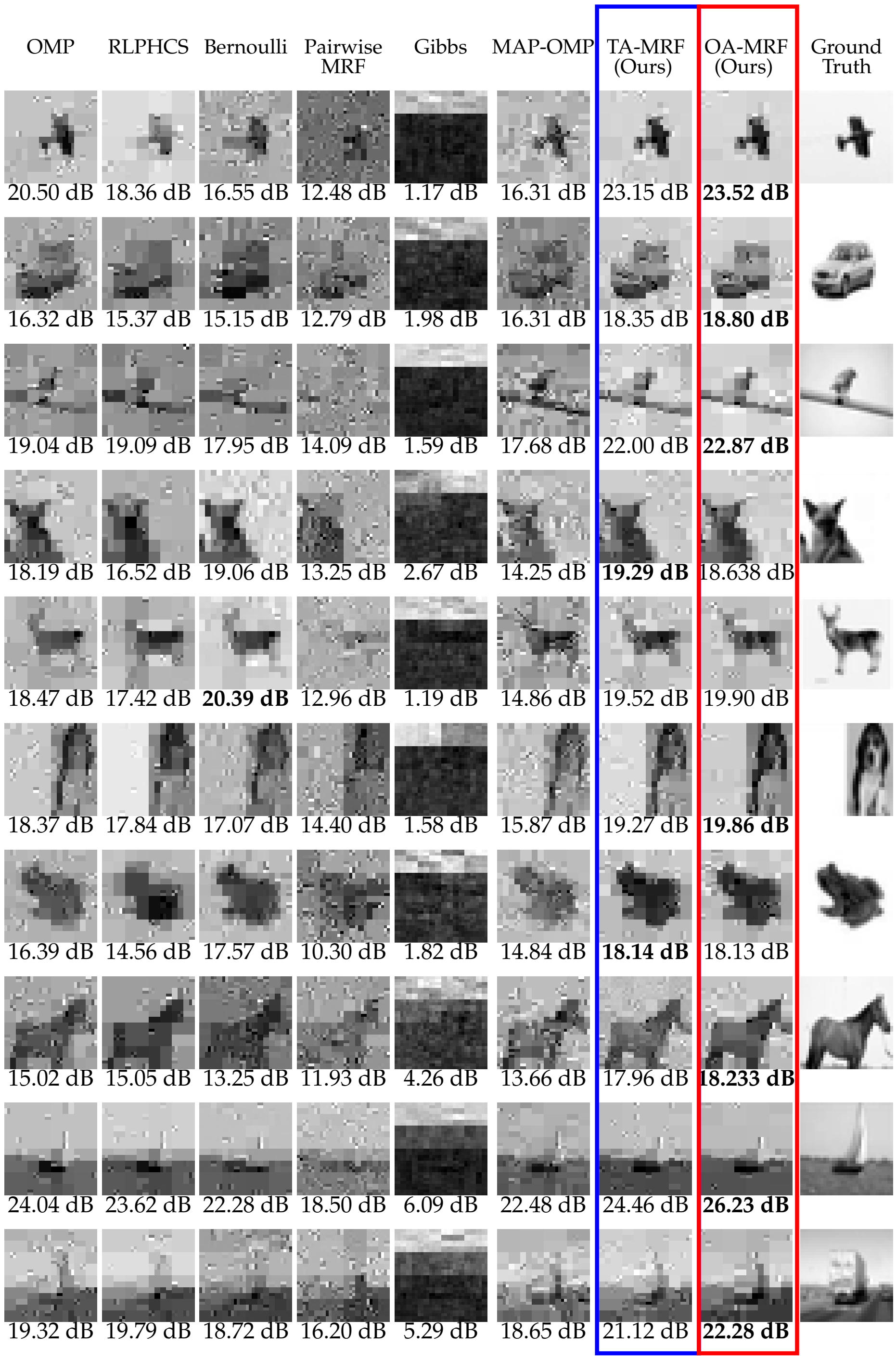

OMP
RLPHCS
Bernoulli
Pairwise MRF
Gibbs
MAP-OMP
TA-MRF (Ours)
OA-MRF (Ours)
Ground Truth
20.50 dB 18.36 dB 16.55 dB 12.48 dB 1.17 dB 16.31 dB 23.15 dB 23.52 dB
16.32 dB 15.37 dB 15.15 dB 12.79 dB 1.98 dB 16.31 dB 18.35 dB 18.80 dB
19.04 dB 19.09 dB 17.95 dB 14.09 dB 1.59 dB 17.68 dB 22.00 dB 22.87 dB
18.19 dB 16.52 dB 19.06 dB 13.25 dB 2.67 dB 14.25 dB 19.29 dB 18.638 dB
18.47 dB 17.42 dB 20.39 dB 12.96 dB 1.19 dB 14.86 dB 19.52 dB 19.90 dB
18.37 dB 17.84 dB 17.07 dB 14.40 dB 1.58 dB 15.87 dB 19.27 dB 19.86 dB
16.39 dB 14.56 dB 17.57 dB 10.30 dB 1.82 dB 14.84 dB 18.14 dB 18.13 dB
15.02 dB 15.05 dB 13.25 dB 11.93 dB 4.26 dB 13.66 dB 17.96 dB 18.233 dB
24.04 dB 23.62 dB 22.28 dB 18.50 dB 6.09 dB 22.48 dB 24.46 dB 26.23 dB
19.32 dB 19.79 dB 18.72 dB 16.20 dB 5.29 dB 18.65 dB 21.12 dB 22.28 dB

FIGURE A.12: Visual results of CIFAR-10 from wavelet sparse signal reconstruction (at $M/N = 0.3$, SNR = 30 dB).

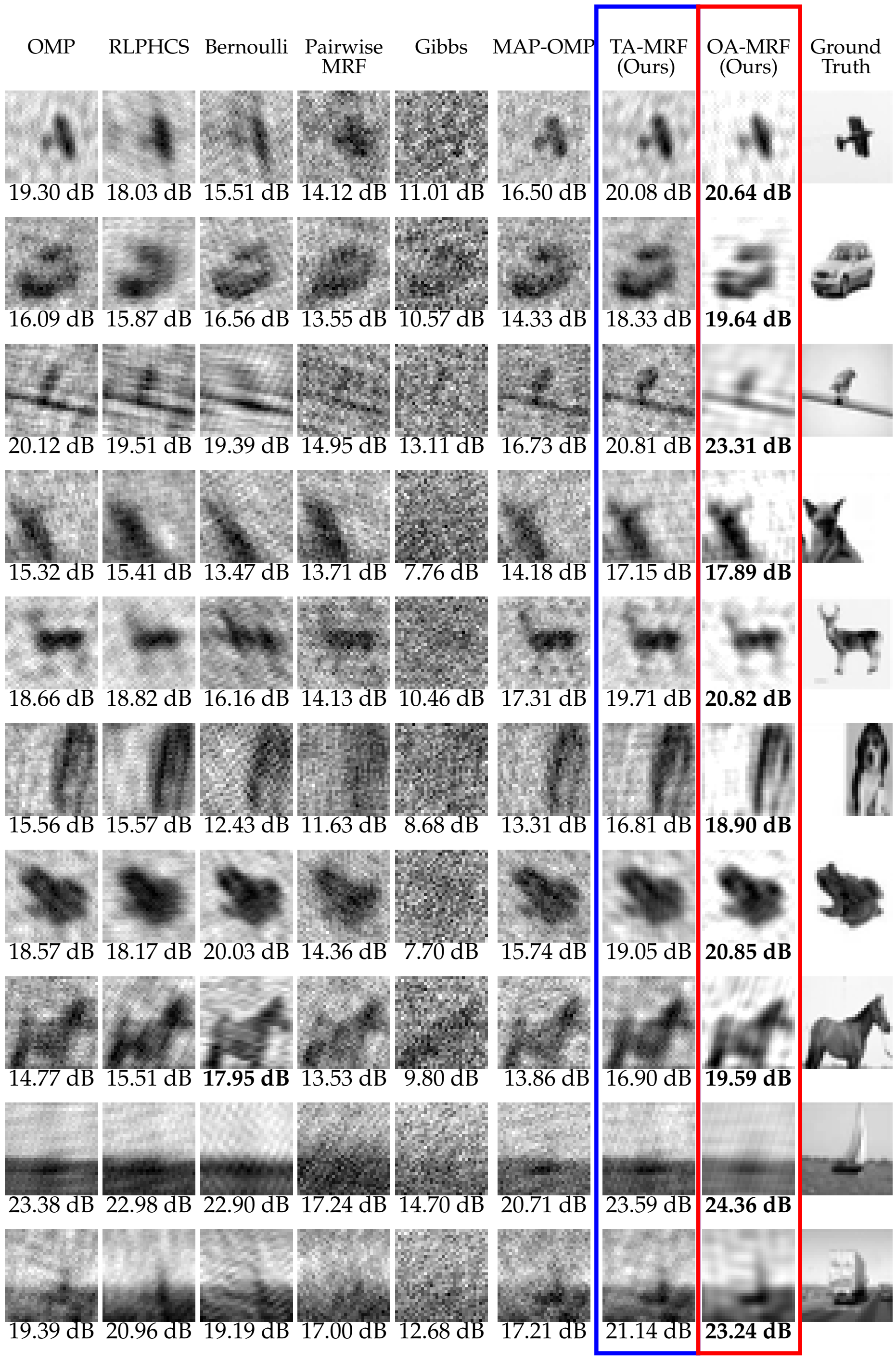

OMP
RLPHCS
Bernoulli
Pairwise MRF
Gibbs
MAP-OMP
TA-MRF (Ours)
OA-MRF (Ours)
Ground Truth
19.30 dB 18.03 dB 15.51 dB 14.12 dB 11.01 dB 16.50 dB 20.08 dB **20.64 dB**
16.09 dB 15.87 dB 16.56 dB 13.55 dB 10.57 dB 14.33 dB 18.33 dB **19.64 dB**
20.12 dB 19.51 dB 19.39 dB 14.95 dB 13.11 dB 16.73 dB 20.81 dB **23.31 dB**
15.32 dB 15.41 dB 13.47 dB 13.71 dB 7.76 dB 14.18 dB 17.15 dB **17.89 dB**
18.66 dB 18.82 dB 16.16 dB 14.13 dB 10.46 dB 17.31 dB 19.71 dB **20.82 dB**
15.56 dB 15.57 dB 12.43 dB 11.63 dB 8.68 dB 13.31 dB 16.81 dB **18.90 dB**
18.57 dB 18.17 dB 20.03 dB 14.36 dB 7.70 dB 15.74 dB 19.05 dB **20.85 dB**
14.77 dB 15.51 dB **17.95 dB** 13.53 dB 9.80 dB 13.86 dB 16.90 dB **19.59 dB**
23.38 dB 22.98 dB 22.90 dB 17.24 dB 14.70 dB 20.71 dB 23.59 dB **24.36 dB**
19.39 dB 20.96 dB 19.19 dB 17.00 dB 12.68 dB 17.21 dB 21.14 dB **23.24 dB**

FIGURE A.13: Visual results of CIFAR-10 natural images from DCT sparse signal reconstruction (at $M/N = 0.3$, SNR = 30 dB).

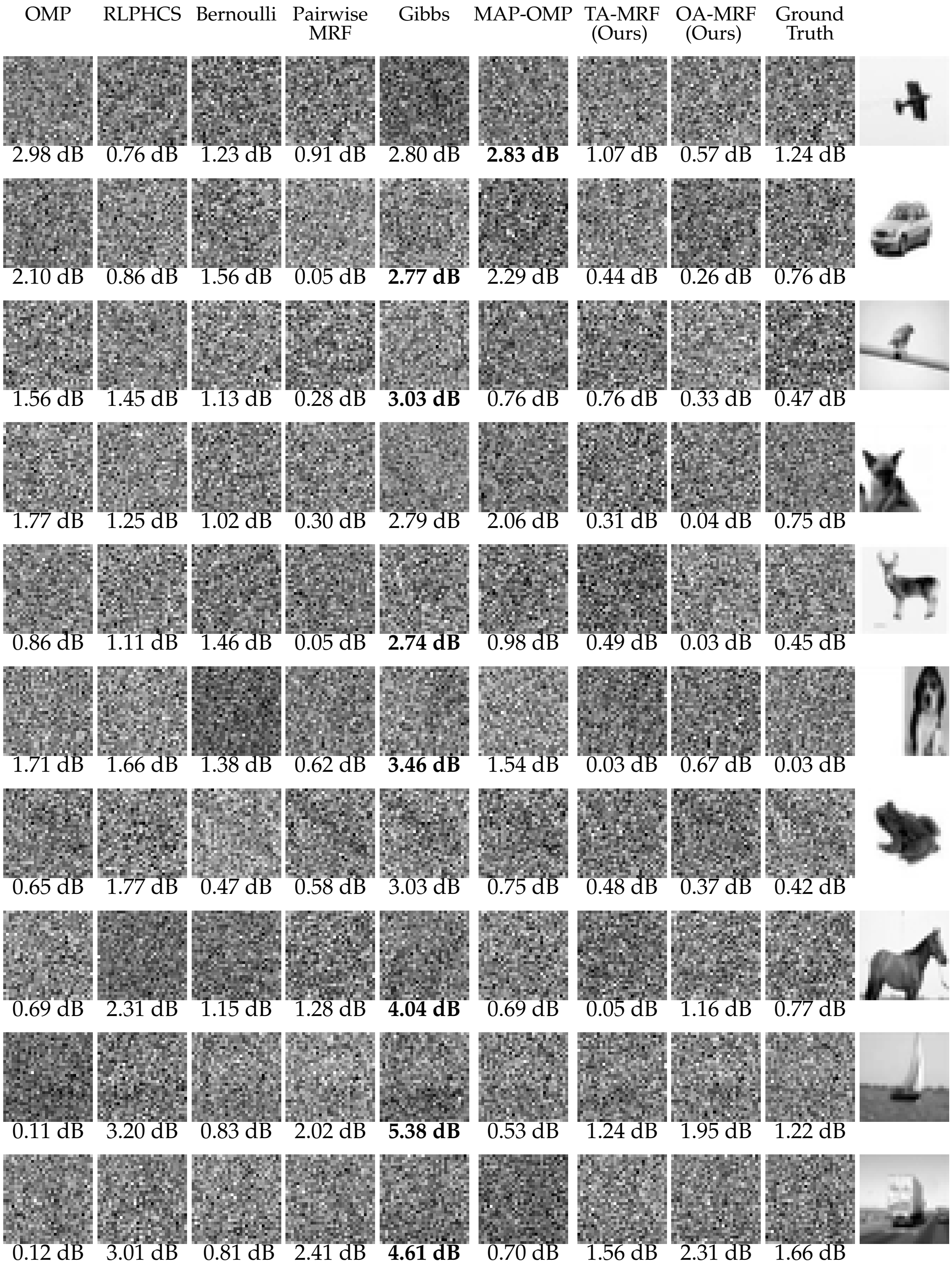

OMP
RLPHCS
Bernoulli
Pairwise MRF
Gibbs
MAP-OMP
TA-MRF (Ours)
OA-MRF (Ours)
Ground Truth
2.98 dB 0.76 dB 1.23 dB 0.91 dB 2.80 dB **2.83 dB** 1.07 dB 0.57 dB 1.24 dB
2.10 dB 0.86 dB 1.56 dB 0.05 dB **2.77 dB** 2.29 dB 0.44 dB 0.26 dB 0.76 dB
1.56 dB 1.45 dB 1.13 dB 0.28 dB **3.03 dB** 0.76 dB 0.76 dB 0.33 dB 0.47 dB
1.77 dB 1.25 dB 1.02 dB 0.30 dB 2.79 dB 2.06 dB 0.31 dB 0.04 dB 0.75 dB
0.86 dB 1.11 dB 1.46 dB 0.05 dB **2.74 dB** 0.98 dB 0.49 dB 0.03 dB 0.45 dB
1.71 dB 1.66 dB 1.38 dB 0.62 dB **3.46 dB** 1.54 dB 0.03 dB 0.67 dB 0.03 dB
0.65 dB 1.77 dB 0.47 dB 0.58 dB 3.03 dB 0.75 dB 0.48 dB 0.37 dB 0.42 dB
0.69 dB 2.31 dB 1.15 dB 1.28 dB **4.04 dB** 0.69 dB 0.05 dB 1.16 dB 0.77 dB
0.11 dB 3.20 dB 0.83 dB 2.02 dB **5.38 dB** 0.53 dB 1.24 dB 1.95 dB 1.22 dB
0.12 dB 3.01 dB 0.81 dB 2.41 dB **4.61 dB** 0.70 dB 1.56 dB 2.31 dB 1.66 dB

FIGURE A.14: Every algorithm fails in reconstruction of the CIFAR-10 natural images in the PCA domain (at $M/N = 0.3$, SNR = 30 dB).

Bibliography

[1] E. J. Candes and M. B. Wakin, “An introduction to compressive sampling”, *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 21–30, 2008, ISSN: 1053-5888. DOI: 10.1109/MSP.2007.914731.

[2] D. L. Donoho, “Compressed sensing”, *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006, ISSN: 0018-9448. DOI: 10.1109/TIT.2006.871582.

[3] J. H. Ender, “On compressive sensing applied to radar”, *Signal Processing*, vol. 90, no. 5, pp. 1402 –1414, 2010, ISSN: 0165-1684. DOI: https://doi.org/10.1016/j.sigpro.2009.11.009. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0165168409004721.

[4] X. X. Zhu and R. Bamler, “Tomographic sar inversion by l1 -norm regularization;the compressive sensing approach”, *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, no. 10, pp. 3839–3846, 2010, ISSN: 0196-2892. DOI: 10.1109/TGRS.2010.2048117.

[5] M. T. Alonso, P. Lopez-Dekker, and J. J. Mallorqui, “A novel strategy for radar imaging based on compressive sensing”, *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, no. 12, pp. 4285–4295, 2010, ISSN: 0196-2892. DOI: 10.1109/TGRS.2010.2051231.

[6] S. Becker, J. Bobin, and E. J. Candès, “NESTA: a fast and accurate first-order method for sparse recovery”, *SIAM Journal on Imaging Sciences*, vol. 4, no. 1, pp. 1–39, 2011. DOI: 10.1137/090756855. [Online]. Available: https://doi.org/10.1137/090756855.

[7] M. Lustig, D. Donoho, and J. M. Pauly, “Sparse MRI: The application of compressed sensing for rapid MR imaging”, *Magnetic Resonance in Medicine*,

vol. 58, no. 6, pp. 1182–1195, DOI: 10.1002/mrm.21391. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/mrm.21391.

[8] M. Lustig, D. L. Donoho, J. M. Santos, and J. M. Pauly, "Compressed sensing MRI", *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 72–82, 2008, ISSN: 1053-5888. DOI: 10.1109/MSP.2007.914728.

[9] M. Lustig and J. Pauly, "SPIRiT: iterative self-consistent parallel imaging reconstruction from arbitrary k-space", vol. 64, pp. 457–71, Aug. 2010.

[10] C. R. Berger, Z. Wang, J. Huang, and S. Zhou, "Application of compressive sensing to sparse channel estimation", *IEEE Communications Magazine*, vol. 48, no. 11, pp. 164–174, 2010, ISSN: 0163-6804. DOI: 10.1109/MCOM.2010.5621984.

[11] L. Xiang, J. Luo, and A. Vasilakos, "Compressed data aggregation for energy efficient wireless sensor networks", in *The 8th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, 2011, pp. 46–54. DOI: 10.1109/SAHCN.2011.5984932.

[12] J. W. Choi, B. Shim, Y. Ding, B. Rao, and D. I. Kim, "Compressed sensing for wireless communications: useful tips and tricks", *IEEE Communications Surveys Tutorials*, vol. 19, no. 3, pp. 1527–1550, 2017. DOI: 10.1109/COMST.2017.2664421.

[13] R. G. Baraniuk, V. Cevher, M. F. Duarte, and C. Hegde, "Model-based compressive sensing", *IEEE Transactions on Information Theory*, vol. 56, no. 4, pp. 1982–2001, 2010, ISSN: 0018-9448. DOI: 10.1109/TIT.2010.2040894.

[14] R. G. Baraniuk, V. Cevher, and M. B. Wakin, "Low-dimensional models for dimensionality reduction and signal recovery: a geometric perspective", *Proceedings of the IEEE*, vol. 98, no. 6, pp. 959–971, 2010.

[15] M. Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables", *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 68,

[16] M. Mishali and Y. C. Eldar, "Reduce and boost: recovering arbitrary sets of jointly sparse vectors", *IEEE Transactions on Signal Processing*, vol. 56, no. 10, pp. 4692–4702, 2008, ISSN: 1053-587X. DOI: 10.1109/TSP.2008.927802.

[17] Y. C. Eldar and M. Mishali, "Robust recovery of signals from a structured union of subspaces", *IEEE Transactions on Inform. Theory*, vol. 55, no. 11, pp. 5302–5316, 2009, ISSN: 0018-9448. DOI: 10.1109/TIT.2009.2030471.

[18] M. Kowalski and B. Torrésani, "Structured sparsity: from mixed norms to structured shrinkage", in *SPARS'09-Signal Processing with Adaptive Sparse Structured Representations*, 2009.

[19] P. Zhao, G. Rocha, and B. Yu, "The composite absolute penalties family for grouped and hierarchical variable selection", *The Annals of Statistics*, pp. 3468–3497, 2009.

[20] L. Jacob, G. Obozinski, and J.-P. Vert, "Group lasso with overlap and graph lasso", in *The 26th Annual International Conference on Machine Learning*, ACM, 2009, pp. 433–440.

[21] M. S. Crouse, R. D. Nowak, and R. G. Baraniuk, "Wavelet-based statistical signal processing using hidden markov models", *IEEE Transactions on Signal Processing*, vol. 46, no. 4, pp. 886–902, 1998, ISSN: 1053-587X. DOI: 10.1109/78.668544.

[22] L. He and L. Carin, "Exploiting structure in wavelet-based bayesian compressive sensing", *IEEE Transactions on Signal Processing*, vol. 57, no. 9, pp. 3488–3497, 2009, ISSN: 1053-587X. DOI: 10.1109/TSP.2009.2022003.

[23] M. F. Duarte, M. B. Wakin, and R. G. Baraniuk, "Wavelet-domain compressive signal reconstruction using a hidden markov tree model", in *2008 ICASSP*, IEEE, 2008, pp. 5137–5140.

[24] C. La and M. N. Do, "Tree-based orthogonal matching pursuit algorithm for signal reconstruction", in *2006 ICIP*, IEEE, 2006, pp. 1277–1280.

[25] J. Huang, T. Zhang, and D. Metaxas, "Learning with structured sparsity", *Journal of Machine Learning Research*, vol. 12, no. Nov, pp. 3371–3412, 2011.

[26] F. Bach, R. Jenatton, J. Mairal, and G. Obozinski, "Optimization with sparsity-inducing penalties", *Foundations and Trends® in Machine Learning*, vol. 4, no. 1, pp. 1–106, 2012.

[27] C. Hegde, P. Indyk, and L. Schmidt, "A nearly-linear time framework for graph-structured sparsity", in *International Conference on Machine Learning*, 2015, pp. 928–937.

[28] B. Zhou and F. Chen, "Graph-structured sparse optimization for connected subgraph detection", in *IEEE 16th International Conference on Data Mining*, 2016, pp. 709–718. DOI: 10.1109/ICDM.2016.0082.

[29] V. Cevher, M. F. Duarte, C. Hegde, and R. Baraniuk, "Sparse signal recovery using markov random fields", in *Advances in Neural Information Processing Systems*, 2009, pp. 257–264.

[30] P. J. Wolfe, S. J. Godsill, and W. Ng, "Bayesian variable selection and regularization for time frequency surface estimation", *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 66, no. 3, pp. 575–589, 2004, ISSN: 1467-9868.

[31] P. Garrigues and B. A Olshausen, "Learning horizontal connections in a sparse coding model of natural images", in *Advances in Neural Information Processing Systems*, 2008, pp. 505–512.

[32] T. Peleg, Y. C. Eldar, and M. Elad, "Exploiting statistical dependencies in sparse representations for signal recovery", *IEEE Transactions on Signal Processing*, vol. 60, no. 5, pp. 2286–2303, 2012, ISSN: 1053-587X.

[33] A. Drémeau, C. Herzet, and L. Daudet, "Boltzmann machine and mean-field approximation for structured sparse decompositions", *IEEE Transactions on Signal Processing*, vol. 60, no. 7, pp. 3425–3438, 2012, ISSN: 1053-587X.

[34] J. Ren, J. Liu, and Z. Guo, "Context-aware sparse decomposition for image denoising and super-resolution", *IEEE Transactions on Image Processing*, vol. 22, no. 4, pp. 1456–1469, 2013, ISSN: 1057-7149. DOI: 10.1109/TIP.2012.2231690.

[35] M. Panić, J. Aelterman, V. Crnojević, and A. Pižurica, "Compressed sensing in mri with a markov random field prior for spatial clustering of subband coefficients", in *The 24th European Signal Processing Conference*, 2016, pp. 562–566. DOI: 10.1109/EUSIPCO.2016.7760311.

[36] R. Torkamani and R. A. Sadeghzadeh, "Bayesian compressive sensing using wavelet based markov random fields", *Signal Processing: Image Communication*, vol. 58, no. Supplement C, pp. 65 –72, 2017, ISSN: 0923-5965. DOI: https://doi.org/10.1016/j.image.2017.06.004.

[37] L. Yu, C. Wei, J. Jia, and H. Sun, "Compressive sensing for cluster structured sparse signals: variational bayes approach", *IET Signal Processing*, vol. 10, no. 7, pp. 770–779, 2016.

[38] L. Yu, H. Sun, J.-P. Barbot, and G. Zheng, "Bayesian compressive sensing for cluster structured sparse signals", *Signal Processing*, vol. 92, no. 1, pp. 259–269, 2012.

[39] L. Yu, H. Sun, G. Zheng, and J. P. Barbot, "Model based bayesian compressive sensing via local beta process", *Signal Processing*, vol. 108, pp. 259–271, 2015.

[40] J. Fang, Y. Shen, H. Li, and P. Wang, "Pattern-coupled sparse bayesian learning for recovery of block-sparse signals", *IEEE Transactions on Signal Processing*, vol. 63, no. 2, pp. 360–372, 2015, ISSN: 1053-587X. DOI: 10.1109/TSP.2014.2375133.

[41] J. Fang, L. Zhang, and H. Li, "Two-dimensional pattern-coupled sparse bayesian learning via generalized approximate message passing", *IEEE Transactions on Image Processing*, vol. 25, no. 6, pp. 2920–2930, 2016, ISSN: 1057-7149. DOI: 10.1109/TIP.2016.2556582.

[42] L. Wang, L. Zhao, G. Bi, and C. Wan, "Sparse representation-based isar imaging using markov random fields", *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 8, no. 8, pp. 3941–3953, 2015, ISSN: 1939-1404. DOI: 10.1109/JSTARS.2014.2359250.

[43] Y. Altmann, M. Pereyra, and J. Bioucas-Dias, "Collaborative sparse regression using spatially correlated supports - application to hyperspectral unmixing", *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 5800–5811, 2015, ISSN: 1057-7149. DOI: 10.1109/TIP.2015.2487862.

[44] O. Eches, J. A. Benediktsson, N. Dobigeon, and J. Y. Tourneret, "Adaptive markov random fields for joint unmixing and segmentation of hyperspectral

images", *IEEE Transactions on Image Processing*, vol. 22, no. 1, pp. 5–16, 2013, ISSN: 1057-7149. DOI: 10.1109/TIP.2012.2204270.

[45] S. Suwanwimolkul, L. Zhang, D. Gong, Z. Zhang, C. Chen, D. C. Ranasinghe, and Q. Shi, "An adaptive markov random field for structured compressive sensing", 2018, Accepted. IEEE Transactions on Image Processing. [Online]. Available: https://arxiv.org/abs/1802.05395.

[46] S. Suwanwimolkul, L. Zhang, D. C. Ranasinghe, and Q. Shi, "One-step adaptive markov random field for structured compressive sensing", 2018, Revised. Signal Processing.

[47] S. Nam, M. Davies, M. Elad, and R. Gribonval, "The cosparse analysis model and algorithms", *Applied and Computational Harmonic Analysis*, vol. 34, no. 1, pp. 30 –56, 2013, ISSN: 1063-5203. DOI: https://doi.org/10.1016/j.acha.2012.03.006. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1063520312000450.

[48] C. Hegde, P. Indyk, and L. Schmidt, "Approximation algorithms for model-based compressive sensing", *IEEE Transactions on Information Theory*, vol. 61, no. 9, pp. 5129–5147, 2015.

[49] T. Blumensath and M. E. Davies, "Sampling theorems for signals from the union of finite-dimensional linear subspaces", *IEEE Transactions on Information Theory*, vol. 55, no. 4, pp. 1872–1882, 2009, ISSN: 0018-9448. DOI: 10.1109/TIT.2009.2013003.

[50] D. Needell and J. Tropp, "Cosamp: iterative signal recovery from incomplete and inaccurate samples", *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 301 –321, 2009, ISSN: 1063-5203. DOI: https://doi.org/10.1016/j.acha.2008.07.002. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1063520308000638.

[51] T. Blumensath and M. E. Davies, "Iterative hard thresholding for compressed sensing", *Applied and Computational Harmonic Analysis*, vol. 27, no. 3, pp. 265 –274, 2009, ISSN: 1063-5203. DOI: https://doi.org/10.1016/j.acha.2009.04.

002. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1063520309000384.

[52] L. Meier, S. Van De Geer, and P. Bühlmann, "The group lasso for logistic regression", *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 70, no. 1, pp. 53–71, 2008, ISSN: 1467-9868. DOI: 10.1111/j.1467-9868.2007.00627.x. [Online]. Available: http://dx.doi.org/10.1111/j.1467-9868.2007.00627.x.

[53] M. F. Duarte and Y. C. Eldar, "Structured compressed sensing: from theory to applications", *IEEE Transactions on Signal Processing*, vol. 59, no. 9, pp. 4053–4085, 2011, ISSN: 1053-587X. DOI: 10.1109/TSP.2011.2161982.

[54] M. Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables", *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 68, no. 1, pp. 49–67, 2006, ISSN: 1467-9868. DOI: 10.1111/j.1467-9868.2005.00532.x. [Online]. Available: http://dx.doi.org/10.1111/j.1467-9868.2005.00532.x.

[55] N. Simon, J. Friedman, T. Hastie, and R. Tibshirani, "A sparse-group lasso", *Journal of Computational and Graphical Statistics*, vol. 22, no. 2, pp. 231–245, 2013. DOI: 10.1080/10618600.2012.681250. eprint: https://doi.org/10.1080/10618600.2012.681250. [Online]. Available: https://doi.org/10.1080/10618600.2012.681250.

[56] R. Baraniuk, R. DeVore, G. Kyriazis, and X. Yu, "Near best tree approximation", *Advances in Computational Mathematics*, vol. 16, no. 4, pp. 357–373, 2002, ISSN: 1572-9044. DOI: 10.1023/A:1014554317692. [Online]. Available: https://doi.org/10.1023/A:1014554317692.

[57] E. W. Tramel, A. Manoel, F. Caltagirone, M. Gabrié, and F. Krzakala, "Inferring sparsity: compressed sensing using generalized restricted boltzmann machines", in *IEEE Information Theory Workshop*, 2016, pp. 265–269. DOI: 10.1109/ITW.2016.7606837.

[58] E. W. Tramel, A. Drémeau, and F. Krzakala, "Approximate message passing with restricted boltzmann machine priors", *Journal of Statistical Mechanics:*

Theory and Experiment, vol. 2016, no. 7, p. 073 401, 2016. [Online]. Available: http://stacks.iop.org/1742-5468/2016/i=7/a=073401.

[59] L. F. Polanía and K. E. Barner, “Exploiting restricted boltzmann machines and deep belief networks in compressed sensing”, *IEEE Transactions on Signal Processing*, vol. 65, no. 17, pp. 4538–4550, 2017, ISSN: 1053-587X. DOI: 10.1109/TSP.2017.2712128.

[60] A. Mousavi, A. B. Patel, and R. G. Baraniuk, “A deep learning approach to structured signal recovery”, in *2015 53rd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, 2015, pp. 1336–1343. DOI: 10.1109/ALLERTON.2015.7447163.

[61] H. Yao, F. Dai, D. Zhang, Y. Ma, S. Zhang, and Y. Zhang, “Deep residual reconstruction network for image compressive sensing”, *CoRR*, vol. abs/1702.05743, 2017. arXiv: 1702.05743. [Online]. Available: http://arxiv.org/abs/1702.05743.

[62] K. Kulkarni, S. Lohit, P. Turaga, R. Kerviche, and A. Ashok, “Reconnet: non-iterative reconstruction of images from compressively sensed measurements”, in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[63] M. J. Beal, “Variational algorithms for approximate bayesian inference”, Tech. Rep., 2003.

[64] V. Cevher, P. Indyk, L. Carin, and R. G. Baraniuk, “Sparse signal recovery and acquisition with graphical models”, *IEEE Signal Processing Magazine*, vol. 27, no. 6, pp. 92–103, 2010, ISSN: 1053-5888. DOI: 10.1109/MSP.2010.938029.

[65] P. J. M. Laarhoven and E. H. L. Aarts, Eds., *Simulated Annealing: Theory and Applications*. Norwell, MA, USA: Kluwer Academic Publishers, 1987, ISBN: 9-027-72513-6. DOI: https://doi.org/10.1007/978-94-015-7744-1.

[66] W. Gilks, S. Richardson, and D. Spiegelhalter, *Markov Chain Monte Carlo in Practice*, ser. Chapman & Hall/CRC Interdisciplinary Statistics. Taylor & Francis, 1995, ISBN: 9780412055515. [Online]. Available: http://books.google.com/books?id=TRXrMWY_i2IC.

[67] A. E. Raftery and S. Lewis, "How many iterations in the gibbs sampler?", in *In Bayesian Statistics 4*, Oxford University Press, 1992, pp. 763–773.

[68] R. Baraniuk, M. Davenport, R. DeVore, and M. Wakin, "A simple proof of the restricted isometry property for random matrices", *Constructive Approximation*, vol. 28, no. 3, pp. 253–263, 2008.

[69] J. Neyman, "Outline of a theory of statistical estimation based on the classical theory of probability", *Philosophical Transactions of the Royal Society of London. Series A, Mathematical and Physical Sciences*, vol. 236, no. 767, pp. 333–380, 1937, ISSN: 00804614. [Online]. Available: http://www.jstor.org/stable/91337.

[70] S. Z. Li, *Markov Random Field Modeling in Image Analysis*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2001, ISBN: 4-431-70309-8.

[71] M. J. Wainwright and M. I. Jordan, "Graphical models, exponential families, and variational inference", *Foundations and Trends® in Machine Learning*, vol. 1, no. 1–2, pp. 1–305, 2008, ISSN: 1935-8237. DOI: 10.1561/2200000001. [Online]. Available: http://dx.doi.org/10.1561/2200000001.

[72] D. Koller and N. Friedman, *Probabilistic graphical models: principles and techniques*. The MIT Press, 2009, ISBN: 0262013193, 9780262013192.

[73] J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli, "Image denoising using scale mixtures of gaussians in the wavelet domain", *IEEE Transactions on Image Processing*, vol. 12, no. 11, pp. 1338–1351, 2003, ISSN: 1057-7149. DOI: 10.1109/TIP.2003.818640.

[74] Y. Y. Boykov and M. P. Jolly, "Interactive graph cuts for optimal boundary amp; region segmentation of objects in n-d images", in *IEEE International Conference on Computer Vision*, vol. 1, 2001, 105–112 vol.1. DOI: 10.1109/ICCV.2001.937505.

[75] W. T. Freeman, E. C. Pasztor, and O. T. Carmichael, "Learning low-level vision", *International Journal of Computer Vision*, vol. 40, no. 1, pp. 25–47, 2000, ISSN: 1573-1405. DOI: 10.1023/A:1026501619075. [Online]. Available: https://doi.org/10.1023/A:1026501619075.

[76] M. Malfait and D. Roose, “Wavelet-based image denoising using a markov random field a priori model”, *IEEE Transactions on Image Processing*, vol. 6, no. 4, pp. 549–565, 1997, ISSN: 1057-7149. DOI: 10.1109/83.563320.

[77] J. Moussouris, “Gibbs and markov random systems with constraints”, *Journal of Statistical Physics*, vol. 10, no. 1, pp. 11–33, 1974, ISSN: 1572-9613. DOI: 10.1007/BF01011714. [Online]. Available: https://doi.org/10.1007/BF01011714.

[78] M. I. Jordan and Y. Weiss, “Graphical models: probabilistic inference”, in *The Handbook of Brain Theory and Neural Networks*, 2nd. Cambridge, MA, USA: MIT Press, 2002, ISBN: 0262011972.

[79] V. Kolmogorov and R. Zabin, “What energy functions can be minimized via graph cuts?”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 2, pp. 147–159, 2004, ISSN: 0162-8828. DOI: 10.1109/TPAMI.2004.1262177.

[80] R. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, and C. Rother, “A comparative study of energy minimization methods for markov random fields”, in *European Conference on Computer Vision*, Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 16–29, ISBN: 978-3-540-33835-2.

[81] S. Geman and D. Geman, “Stochastic relaxation, gibbs distributions, and the bayesian restoration of images”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-6, no. 6, pp. 721–741, 1984, ISSN: 0162-8828. DOI: 10.1109/TPAMI.1984.4767596.

[82] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul, “An introduction to variational methods for graphical models”, *Machine Learning*, vol. 37, no. 2, pp. 183–233, 1999, ISSN: 1573-0565. DOI: 10.1023/A:1007665907178. [Online]. Available: https://doi.org/10.1023/A:1007665907178.

[83] Z. Zhang, Q. Shi, J. McAuley, W. Wei, Y. Zhang, and A. v. d. Hengel, “Pairwise matching through max-weight bipartite belief propagation”, in *IEEE*

Conference on Computer Vision and Pattern Recognition, 2016, pp. 1202–1210. DOI: 10.1109/CVPR.2016.135.

[84] J. Besag, "Spatial interaction and the statistical analysis of lattice systems", *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 36, no. 2, pp. 192–236, 1974, ISSN: 00359246. [Online]. Available: http://www.jstor.org/stable/2984812.

[85] G. E. Hinton, "Training products of experts by minimizing contrastive divergence", *Neural Computation*, vol. 14, no. 8, pp. 1771–1800, 2002. DOI: 10.1162/089976602760128018. [Online]. Available: https://doi.org/10.1162/089976602760128018.

[86] Y. Cun and F. Huang, "Loss functions for discriminative training of energy-based models", in *Proceedings of the 10th International Workshop on Artificial Intelligence and Statistics*. 2005, pp. 206–213, ISBN: 097273581X.

[87] A. Hyvärinen, "Estimation of non-normalized statistical models by score matching", *Journal of Machine Learning Research*, vol. 6, pp. 695–709, 2005, ISSN: 1532-4435. [Online]. Available: http://dl.acm.org/citation.cfm?id=1046920.1088696.

[88] D. J. C. MacKay, *Information Theory, Inference & Learning Algorithms*. New York, NY, USA: Cambridge University Press, 2002, ISBN: 0521642981.

[89] L. Zhang, M. Yang, and X. Feng, "Sparse representation or collaborative representation: which helps face recognition?", in *IEEE International Conference on Computer Vision*, 2011, pp. 471–478. DOI: 10.1109/ICCV.2011.6126277.

[90] 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*, vol. 31, no. 2, pp. 210–227, 2009, ISSN: 0162-8828. DOI: 10.1109/TPAMI.2008.79.

[91] M. Zhang and A. A. Sawchuk, "Human daily activity recognition with sparse representation using wearable sensors", *IEEE Journal of Biomedical and Health Informatics*, vol. 17, no. 3, pp. 553–560, 2013, ISSN: 2168-2194. DOI: 10.1109/JBHI.2013.2253613.

[92] W. Deng, J. Hu, and J. Guo, "In defense of sparsity based face recognition", in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 399–406. DOI: 10.1109/CVPR.2013.58.

[93] Y. Chi and F. Porikli, "Connecting the dots in multi-class classification: from nearest subspace to collaborative representation", in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 3602–3609. DOI: 10.1109/CVPR.2012.6248105.

[94] ——, "Classification and boosting with multiple collaborative representations", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 8, pp. 1519–1531, 2014, ISSN: 0162-8828. DOI: 10.1109/TPAMI.2013.236.

[95] E. Elhamifar and R. Vidal, "Robust classification using structured sparse representation", in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011, pp. 1873–1879. DOI: 10.1109/CVPR.2011.5995664.

[96] ——, "Sparse subspace clustering: algorithm, theory, and applications", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 11, pp. 2765–2781, 2013, ISSN: 0162-8828. DOI: 10.1109/TPAMI.2013.57.

[97] M. Soltanolkotabi, E. Elhamifar, and E. J. Candès, "Robust subspace clustering", *Ann. Statist.*, vol. 42, no. 2, pp. 669–699, Apr. 2014. DOI: 10.1214/13-AOS1199. [Online]. Available: https://doi.org/10.1214/13-AOS1199.

[98] S. Cai, L. Zhang, W. Zuo, and X. Feng, "A probabilistic collaborative representation based approach for pattern classification", in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2950–2959. DOI: 10.1109/CVPR.2016.322.

[99] W. Xu, M. Zhang, A. A. Sawchuk, and M. Sarrafzadeh, "Co-recognition of human activity and sensor location via compressed sensing in wearable body sensor networks", in *2012 Ninth International Conference on Wearable and Implantable Body Sensor Networks*, 2012, pp. 124–129. DOI: 10.1109/BSN.2012.14.

[100] C. Chen, R. Jafari, and N. Kehtarnavaz, "Improving human action recognition using fusion of depth camera and inertial sensors", *IEEE Transactions on*

Human-Machine Systems, vol. 45, no. 1, pp. 51–61, 2015, ISSN: 2168-2291. DOI: 10.1109/THMS.2014.2362520.

[101] C. Wang, B. Zhang, K. Ren, J. M. Roveda, C. W. Chen, and Z. Xu, "A privacy-aware cloud-assisted healthcare monitoring system via compressive sensing", in *IEEE INFOCOM 2014 - IEEE Conference on Computer Communications*, 2014, pp. 2130–2138. DOI: 10.1109/INFOCOM.2014.6848155.

[102] D. P. Wipf, B. D. Rao, and S. Nagarajan, "Latent variable bayesian models for promoting sparsity", *IEEE Transactions on Information Theory*, vol. 57, no. 9, pp. 6236–6255, 2011, ISSN: 0018-9448. DOI: 10.1109/TIT.2011.2162174.

[103] L. Zhang, W. Wei, C. Tian, F. Li, and Y. Zhang, "Exploring structured sparsity by a reweighted laplace prior for hyperspectral compressive sensing", *IEEE Transactions on Image Processing*, vol. 25, no. 10, pp. 4974–4988, 2016, ISSN: 1057-7149. DOI: 10.1109/TIP.2016.2598652.

[104] M. J. Wainwright and M. I. Jordan, "Graphical models, exponential families, and variational inference", *Foundations and Trends® in Machine Learning*, vol. 1, no. 1-2, pp. 1–305, 2008.

[105] C. Wang, N. Komodakis, and N. Paragios, "Markov random field modeling, inference and learning in computer vision and image understanding: a survey", *Computer Vision and Image Understanding*, vol. 117, no. 11, pp. 1610 –1627, 2013, ISSN: 1077-3142. DOI: https://doi.org/10.1016/j.cviu.2013.07.004. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1077314213001343.

[106] J. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit", *IEEE Transactions on Inform. Theory*, vol. 53, no. 12, pp. 4655–4666, 2007.

[107] A. Drémeau, C. Herzet, and L. Daudet, "Soft bayesian pursuit algorithm for sparse representations", in *2011 IEEE Statistical Signal Processing Workshop (SSP)*, 2011, pp. 341–344. DOI: 10.1109/SSP.2011.5967699.

[108] S. Foucart and H. Rauhut, *A Mathematical Introduction to Compressive Sensing*. Birkhuser Basel, 2013, ISBN: 0817649476, 9780817649470.

[109] G. B. Dantzig and P. Wolfe, “Decomposition principle for linear programs”, *Operational Research*, vol. 8, no. 1, pp. 101–111, Feb. 1960, ISSN: 0030-364X. DOI: 10.1287/opre.8.1.101. [Online]. Available: http://dx.doi.org/10.1287/opre.8.1.101.

[110] V. Kolmogorov, “Convergent tree-reweighted message passing for energy minimization”, *IEEE Transactions Pattern Analysis and Machine Intelligence*, vol. 28, no. 10, pp. 1568–1583, 2006, ISSN: 0162-8828. DOI: 10.1109/TPAMI.2006.200.

[111] O. Meshi and A. Globerson, “An alternating direction method for dual map lp relaxation”, in *Machine Learning and Knowledge Discovery in Databases*, Springer, 2011, pp. 470–483.

[112] S. Parise and M. Welling, “Learning in markov random fields: an empirical study”, in *Proceedings of the Joint Statistical Meeting*, 2005.

[113] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition”, *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[114] T. Sim, S. Baker, and M. Bsat, “The cmu pose, illumination, and expression database”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 12, pp. 1615 –1618, 2003.

[115] A. Krizhevsky, “Learning multiple layers of features from tiny images”, University of Toronto, Tech. Rep., 2009.

[116] D. G. Tzikas, A. C. Likas, and N. P. Galatsanos, “The variational approximation for bayesian inference”, *IEEE Signal Processing Magazine*, vol. 25, no. 6, pp. 131–146, 2008, ISSN: 1053-5888. DOI: 10.1109/MSP.2008.929620.

[117] J. Besag, “Statistical analysis of non-lattice data”, *Journal of the Royal Statistical Society. Series D (The Statistician)*, vol. 24, no. 3, pp. 179–195, 1975, ISSN: 00390526, 14679884. [Online]. Available: http://www.jstor.org/stable/2987782.

[118] D. P. Wipf and B. D. Rao, "An empirical bayesian strategy for solving the simultaneous sparse approximation problem", *IEEE Transactions on Signal Processing*, vol. 55, no. 7, pp. 3704–3716, 2007, ISSN: 1053-587X. DOI: 10.1109/TSP.2007.894265.

[119] A. Pantelopoulos and N. G. Bourbakis, "A survey on wearable sensor-based systems for health monitoring and prognosis", *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 40, no. 1, pp. 1–12, 2010, ISSN: 1094-6977. DOI: 10.1109/TSMCC.2009.2032660.

[120] O. D. Lara and M. A. Labrador, "A survey on human activity recognition using wearable sensors", *IEEE Communications Surveys Tutorials*, vol. 15, no. 3, pp. 1192–1209, 2013, ISSN: 1553-877X. DOI: 10.1109/SURV.2012.110112.00192.

[121] G.-Z. Yang, *Body Sensor Networks*, 2nd. Springer Publishing Company, Incorporated, 2014, ISBN: 1447163737, 9781447163732.

[122] S. M. R. Islam, D. Kwak, M. H. Kabir, M. Hossain, and K. S. Kwak, "The internet of things for health care: a comprehensive survey", *IEEE Access*, vol. 3, pp. 678–708, 2015, ISSN: 2169-3536. DOI: 10.1109/ACCESS.2015.2437951.

[123] M. Mubashir, L. Shao, and L. Seed, "A survey on fall detection: principles and approaches", *Neurocomputing*, vol. 100, pp. 144 –152, 2013, Special issue: Behaviours in video, ISSN: 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2011.09.037. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0925231212003153.

[124] R. L. Shinmoto Torres, R. Visvanathan, D. Abbott, K. D. Hill, and D. C. Ranasinghe, "A battery-less and wireless wearable sensor system for identifying bed and chair exits in a pilot trial in hospitalized older people", *PLOS ONE*, vol. 12, no. 10, pp. 1–25, Oct. 2017. DOI: 10.1371/journal.pone.0185670. [Online]. Available: https://doi.org/10.1371/journal.pone.0185670.

[125] F. Cincotti, D. Mattia, F. Aloise, S. Bufalari, G. Schalk, G. Oriolo, A. Cherubini, M. G. Marciani, and F. Babiloni, "Non-invasive brain–computer interface system: towards its application as assistive technology", *Brain Research Bulletin*,

vol. 75, no. 6, pp. 796 –803, 2008, Special Issue: Robotics and Neuroscience, ISSN: 0361-9230. DOI: https://doi.org/10.1016/j.brainresbull.2008.01.007. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0361923008000142.

[126] D. J. Cook, J. C. Augusto, and V. R. Jakkula, "Ambient intelligence: technologies, applications, and opportunities", *Pervasive and Mobile Computing*, vol. 5, no. 4, pp. 277 –298, 2009, ISSN: 1574-1192. DOI: https://doi.org/10.1016/j.pmcj.2009.04.001. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S157411920900025X.

[127] J. D. Nahrgang, F. P. Morgeson, and D. A. Hofmann, "Safety at work: a meta-analytic investigation of the link between job demands, job resources, burnout, engagement, and safety outcomes", *Journal of Applied Psychology*, vol. 96, no. 1, pp. 71–94, 2011.

[128] M. Dağdeviren and İhsan Yüksel, "Developing a fuzzy analytic hierarchy process (AHP) model for behavior-based safety management", *Information Sciences*, vol. 178, no. 6, pp. 1717 –1733, 2008, ISSN: 0020-0255. DOI: https://doi.org/10.1016/j.ins.2007.10.016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0020025507005130.

[129] A. Jain and B. Chandrasekaran, "39 dimensionality and sample size considerations in pattern recognition practice", in *Classification Pattern Recognition and Reduction of Dimensionality*, ser. Handbook of Statistics, vol. 2, Elsevier, 1982, pp. 835 –855. DOI: https://doi.org/10.1016/S0169-7161(82)02042-2. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0169716182020422.

[130] J. Fan and R. Li, "Statistical challenges with high dimensionality: feature selection in knowledge discovery", in *Proceedings of the International Congress of Mathematicians*, vol. 3, Madrid, Mar. 2006.

[131] S. Khalifa, M. Hassan, and A. Seneviratne, "Pervasive self-powered human activity recognition without the accelerometer", in *IEEE International Conference on Pervasive Computing and Communications*, 2015, pp. 79–86. DOI: 10.1109/PERCOM.2015.7146512.

[132] A. Bulling, U. Blanke, and B. Schiele, "A tutorial on human activity recognition using body-worn inertial sensors", *ACM Computing Surveys*, vol. 46, no. 3, 33:1–33:33, 2014, ISSN: 0360-0300. DOI: 10.1145/2499621. [Online]. Available: http://doi.acm.org/10.1145/2499621.

[133] S. B. Kotsiantis, "Supervised machine learning: a review of classification techniques", in *Emerging artificial intelligence applications in computer engineering real word AI systems with applications in eHealth, HCI, information retrieval and pervasive technologies*, Amsterdam, The Netherlands, The Netherlands: IOS Press, 2007, pp. 3–24, ISBN: 978-1-58603-780-2. [Online]. Available: http://dl.acm.org/citation.cfm?id=1566770.1566773.

[134] T. Cover and P. Hart, "Nearest neighbor pattern classification", *IEEE Transactions on Information Theory*, vol. 13, no. 1, pp. 21–27, Sep. 2006, ISSN: 0018-9448. DOI: 10.1109/TIT.1967.1053964. [Online]. Available: http://dx.doi.org/10.1109/TIT.1967.1053964.

[135] D. Wettschereck, D. W. Aha, and T. Mohri, "A review and empirical evaluation of feature weighting methods for a class of lazy learning algorithms", *Artificial Intelligence Review*, vol. 11, no. 1, pp. 273–314, 1997, ISSN: 1573-7462. DOI: 10.1023/A:1006593614256. [Online]. Available: https://doi.org/10.1023/A:1006593614256.

[136] R. Yao, G. Lin, Q. Shi, and D. C. Ranasinghe, "Efficient dense labelling of human activity sequences from wearables using fully convolutional networks", *Pattern Recognition*, vol. 78, pp. 252 –266, 2018, ISSN: 0031-3203. DOI: https://doi.org/10.1016/j.patcog.2017.12.024. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0031320317305204.

[137] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf, "Support vector machines", *IEEE Intelligent Systems and their Applications*, vol. 13, no. 4, pp. 18–28, 1998, ISSN: 1094-7167. DOI: 10.1109/5254.708428.

[138] C.-C. Chang and C.-J. Lin, "Libsvm: a library for support vector machines", *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, 27:1–27:27, May 2011, ISSN: 2157-6904. DOI: 10.1145/1961189.1961199. [Online]. Available: http://doi.acm.org/10.1145/1961189.1961199.

[139] J. B. Yang, M. N. Nguyen, P. P. San, X. L. Li, and S. Krishnaswamy, "Deep convolutional neural networks on multichannel time series for human activity recognition", in *Proceedings of the 24th International Conference on Artificial Intelligence*, Buenos Aires, Argentina: AAAI Press, 2015, pp. 3995–4001, ISBN: 978-1-57735-738-4. [Online]. Available: http://dl.acm.org/citation.cfm?id=2832747.2832806.

[140] N. Y. Hammerla, S. Halloran, and T. Plötz, "Deep, convolutional, and recurrent models for human activity recognition using wearables", in *Proceedings of the 25th International Joint Conference on Artificial Intelligence*, New York, New York, USA: AAAI Press, 2016, pp. 1533–1540, ISBN: 978-1-57735-770-4. [Online]. Available: http://dl.acm.org/citation.cfm?id=3060832.3060835.

[141] O. Boiman, E. Shechtman, and M. Irani, "In defense of nearest-neighbor based image classification", in *IEEE Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–8. DOI: 10.1109/CVPR.2008.4587598.

[142] H. Zhang, A. C. Berg, M. Maire, and J. Malik, "SVM-KNN: discriminative nearest neighbor classification for visual category recognition", in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, 2006, pp. 2126–2136. DOI: 10.1109/CVPR.2006.301.

[143] J.-L. Reyes-Ortiz, L. Oneto, A. Samà, X. Parra, and D. Anguita, "Transition-aware human activity recognition using smartphones", *Neurocomputing*, vol. 171, no. C, pp. 754–767, Jan. 2016, ISSN: 0925-2312. DOI: 10.1016/j.neucom.2015.07.085. [Online]. Available: https://doi.org/10.1016/j.neucom.2015.07.085.

[144] M. Zhang and A. A. Sawchuk, "A feature selection-based framework for human activity recognition using wearable multimodal sensors", in *Proceedings of the 6th International Conference on Body Area Networks*, Beijing, China: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2011, pp. 92–98, ISBN: 978-1-936968-29-9. [Online]. Available: http://dl.acm.org/citation.cfm?id=2318776.2318798.

[145] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*, 3rd. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2011, ISBN: 0123814790, 9780123814791.

[146] E. Candes, "L1-Magic: Recovery of Sparse Signals", Standford University, Tech. Rep. [Online]. Available: http://www.acm.caltech.edu/l1magic/.

[147] J. A. Ward, P. Lukowicz, and H. W. Gellersen, "Performance metrics for activity recognition", *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 1, 6:1–6:23, Jan. 2011, ISSN: 2157-6904. DOI: 10.1145/1889681.1889687. [Online]. Available: http://doi.acm.org/10.1145/1889681.1889687.

[148] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks", in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds., Curran Associates, Inc., 2012, pp. 1097–1105. [Online]. Available: http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf.

[149] J. Xie, L. Xu, and E. Chen, "Image denoising and inpainting with deep neural networks", in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds., Curran Associates, Inc., 2012, pp. 341–349. [Online]. Available: http://papers.nips.cc/paper/4686-image-denoising-and-inpainting-with-deep-neural-networks.pdf.

[150] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution", in *Computer Vision – ECCV 2014*, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds., Cham: Springer International Publishing, 2014, pp. 184–199, ISBN: 978-3-319-10593-2.

[151] S. J. Pan and Q. Yang, "A survey on transfer learning", *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010, ISSN: 1041-4347. DOI: 10.1109/TKDE.2009.191.

[152] Y. Jernite, A. M. Rush, and D. Sontag, "A fast variational approach for learning markov random field language models", in *Proceedings of the 32Nd International Conference on International Conference on Machine Learning - Volume 37*, ser.

ICML'15, Lille, France: JMLR.org, 2015, pp. 2209–2216. [Online]. Available: http://dl.acm.org/citation.cfm?id=3045118.3045353.

[153] A. Schwing, T. Hazan, M. Pollefeys, and R. Urtasun, "Distributed message passing for large scale graphical models", in *CVPR 2011*, 2011, pp. 1833–1840. DOI: 10.1109/CVPR.2011.5995642.

[154] S. Rangan, "Generalized approximate message passing for estimation with random linear mixing", in *IEEE International Symposium on Information Theory Proceedings*, 2011, pp. 2168–2172. DOI: 10.1109/ISIT.2011.6033942.

[155] P. Kohli and C. Rother, "Higher-order models in computer vision", in *Image Processing and Analysing Graphs: Theory and Practice*. CRC Press, 2012, ch. 3, ISBN: 9781439855072. [Online]. Available: https://www.microsoft.com/en-us/research/publication/higher-order-models-computer-vision/.

[156] Y. Chen, W. Feng, R. Ranftl, H. Qiao, and T. Pock, "A higher-order MRF based variational model for multiplicative noise reduction", *IEEE Signal Processing Letters*, vol. 21, no. 11, pp. 1370–1374, 2014, ISSN: 1070-9908. DOI: 10.1109/LSP.2014.2337274.