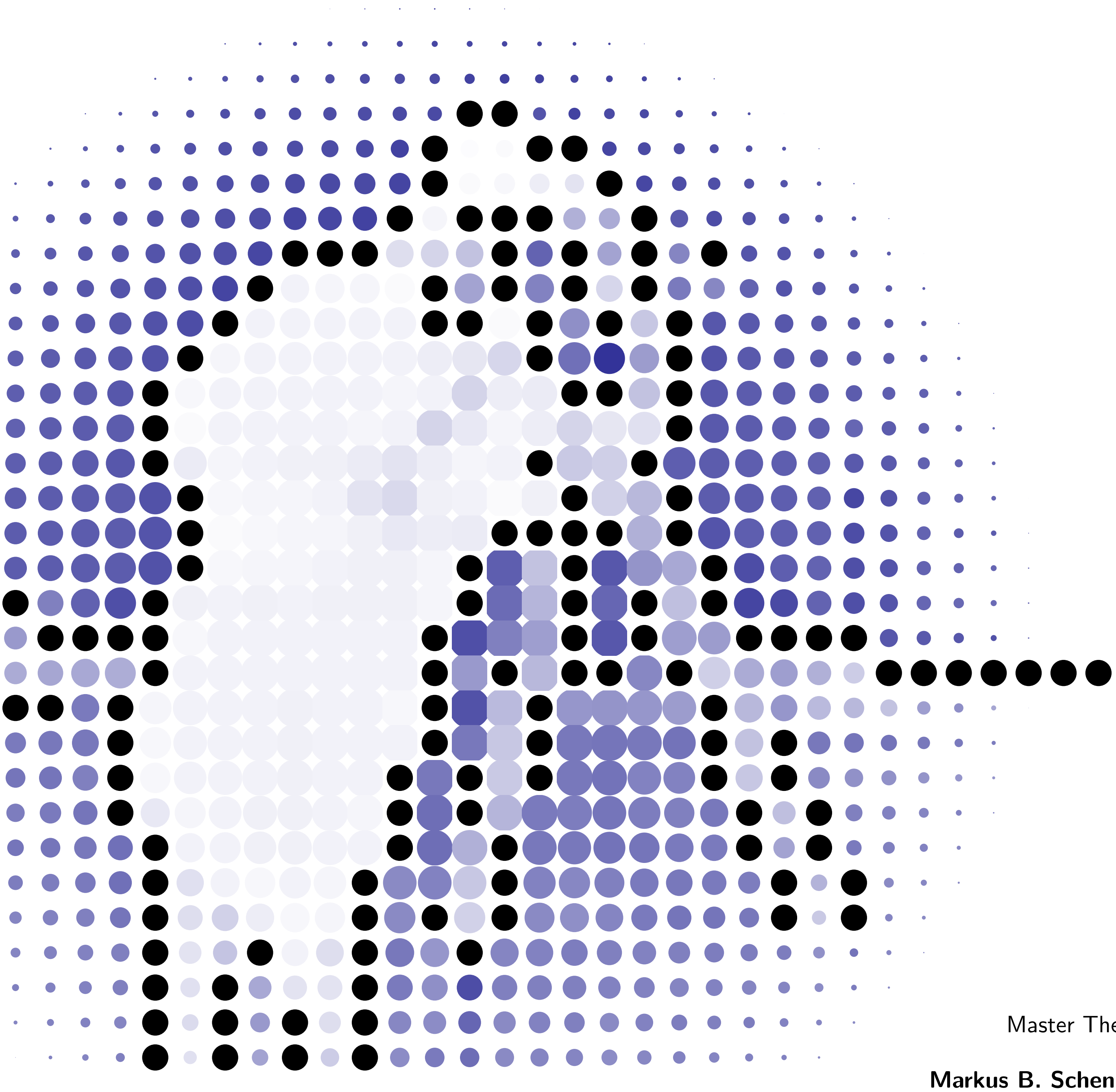
Sparsity Based Correlation Models for Joint Reconstruction of Compressed Images



Master Thesis

Markus B. Schenkel

Signal Processing Laboratory 4
Swiss Federal Institute of Technology
Lausanne, Switzerland

February 2010

markus.schenkel@epfl.ch

Cover: A parsimonious view of the *Photographer*.

Sparsity Based Correlation Models for Joint Reconstruction of Compressed Images

Markus B. Schenkel

Master thesis project supervised by

Prof. Dr. Pascal Frossard		Dr. Feng Wu
Signal Processing Laboratory 4	and	Internet Media Group
Swiss Federal Institute of Technology		Microsoft Research Asia
Lausanne, Switzerland		Beijing, China

2010 – 02 – 26

Preface

This master's thesis concludes my research on applications of sparse image representations and related inverse problems to the joint decoding of compressed images. It was initially motivated by the fast growing body of work around *Compressed Sensing* that has emerged during the recent years and offers some surprising results. This project allowed me to deepen my understanding of image processing and communication – a field that has caught my interest because it combines the beauty and richness of images with the elegance of mathematics to enable the multimedia applications that are undoubtedly an important aspect of the digital era we are living in today.

The entire work presented in the following was conducted while I was with the *Internet Media Group* at *Microsoft Research Asia* (MSRA) in Beijing. At this point I would like to thank Dr. Feng Wu for giving me the opportunity to work with his group and his advices, Dr. Chong Luo for the enlightening discussions, Prof. Pascal Frossard for his encouragement and valuable inputs from far away and Hao Cui for helping me out in countless ways.

During my time in Beijing I could not only work under ideal conditions at a leading research institution and among many bright people with exciting ideas, but was also emerged into the fascinating world of Chinese culture and learned a lot about this multifaceted country thanks to my colleagues and friends. I could never have made all these enriching experiences at home.

On a different note, Igor Carron's blog *Nuit Blanche* [12] has helped me a lot to get quickly acquainted with the whole field of compressed sensing and related topics.

Last but not least I would like to thank my parents and family for their continuous support throughout the many years of my education. *Vielen Dank!*

Markus B. Schenkel
Beijing, February 26, 2010

Summary

Recently there has been a big and growing interest in sparse representations of signals. A signal can always be represented as a linear combination of basis vectors because a basis is by definition complete. If the basis is well chosen by exploiting the correlation structure of a given class of signals, only a small number of basis vectors will be required, leading to a sparse representation. If we further use more vectors than necessary to form an overcomplete dictionary, sparse representations become possible for an even larger range of signals. This can be applied to compression, feature extraction or to regularize a number of inverse problems. A particular body of work that is motivated by the fact that such sparse representations exist for a wide range of signals is known as *compressed sensing*. It states that the dimensionality of signals that are known to be sparse in some basis can be reduced and that it is possible to accurately recover the original signal from the compressed data with high probability and efficient algorithms. After an introduction to the relevant concepts, this thesis presents two applications of this framework to image communication.

In a first part we propose a new scheme for wireless video multicast based on compressed sensing. It has the property of graceful degradation and, unlike systems adhering to traditional separate coding, it does not suffer from a cliff effect. Compressed sensing is applied to generate measurements of equal importance from a video such that a receiver with a better channel will naturally have more information at hands to reconstruct the content without penalizing others. We experimentally compare different random matrices at the encoder side in terms of their performance for video transmission. We further investigate how properties of natural images can be exploited to improve the reconstruction performance by transmitting a small amount of side information. And we propose a way of exploiting inter-frame correlation by extending only the decoder. Finally, we compare our results with a different scheme targeting the same problem with simulations and find competitive results for some channel configurations.

In a second part we address the problem of joint decoding of JPEG encoded stereo image pairs. Stereo images typically contain a high degree of redundancy. But cameras would have to implement proprietary encoding solutions for predictive coding, because no standard technology is available. Furthermore the limited processing power of portable cameras encourages a distributed scheme. We propose to rather use the ubiquitous JPEG compression tools, and focus on the joint decoding problem for quality enhancement. We formulate this as a constrained optimization problem and show how appropriate regularization leads to more consistent results. This scheme is similar to a distributed source coding framework, where the exploitation of the correlation at the decoder permits to save on the overall bandwidth. Experiments on natural stereo images show an improvement in both visual quality and PSNR when compared to separate decoding.

Contents

Key to notation

Throughout this text matrices and operators are denoted with upper case, upright bold symbols $\mathbf{A}$, vectors in lowercase, bold italic $\boldsymbol{a}$ and scalars as italic a. Elements of matrices and vectors are indexed as $A_{i,j}$ and a_k respectively. A hat $\hat{a}$ denotes the reconstruction result. If not otherwise noted images are implicitly concatenated into column vectors. Because all theory presented is going to be applied to images, only real valued signals in a discrete space are studied. The words *signal* and *image* are often used synonym.

A superscript $a^{(i)}$ is used for the value of a during the i^{th} iteration of an algorithm. For a set $\Gamma \subseteq \{1, \ldots, N\}$, $\mathbf{A}_\Gamma$ is the reduction of $\mathbf{A} \in \mathbb{R}^{M\times N}$ to the columns indexed by Γ.

The following table summarizes the most commonly used symbols and operations:

Symbol	**Dimension**	**Description**
N	$\mathbb{N}$	Signal dimension in the pixel domain
M	$\mathbb{N}$	Signal dimension in the compressed domain
D	$\mathbb{N}$	Number of elements in a dictionary
K	$\mathbb{N}$	Number of non-zero coefficients (Sparsity) of a signal
L	$\mathbb{N}$	Number of blocks in an image
$\mathbf{\Phi}$	$\mathbb{R}^{M\times N}$	Measurement matrix
$\mathbf{\Psi}$	$\mathbb{R}^{N\times D}$	Transform matrix or dictionary
$\mathbf{D}$	$\mathbb{R}^{N\times N}$	Two-dimensional DCT transform matrix
$\boldsymbol{b}$	$\mathbb{R}^{N}$	An image block of size $N = n \times n$ in column form
$\boldsymbol{s}$	$\mathbb{R}^{D}$	Coefficients of a signal representation over a dictionary
$\mathbf{A}^\dagger$		Pseudo inverse of $\mathbf{A}$
$\lvert\Gamma\rvert$	$\mathbb{N}$	Cardinality of a set Γ
$\lVert\cdot\rVert_0$	$\mathbb{R}$	ℓ_0 pseudo norm
$\lVert\cdot\rVert_p$	$\mathbb{R}$	ℓ_p norm for $p \geq 1$
$\lceil\cdot\rceil$	$\mathbb{N}$	Next higher integer
$[\,\cdot\,]$	$\mathbb{Z}$	Closest integer
$\lfloor\cdot\rfloor$	$\mathbb{N}$	Next lower integer

Chapter 1

Introduction

This chapter presents the relevant background information and theory related to this thesis. We introduce sparse image coding methods and common inverse problems, followed by an overview of the compressed sensing framework and the important algorithms.

1.1 Sparse Representations

1.1.1 Signal model

A discrete signal $\boldsymbol{x} \in \mathbb{R}^N$ can be written as a linear combination of a set of D vectors $\{\boldsymbol{\psi}_i\}$ weighted by the corresponding coefficients $\{s_i\}$. Those vectors are often referred to as atoms. A sufficient condition on the *dictionary* $\boldsymbol{\Psi} = [\boldsymbol{\psi}_1 \cdots \boldsymbol{\psi}_D] \in \mathbb{R}^{N \times D}$ to represent any possible signal $\boldsymbol{x} \in \mathbb{R}^N$ is that it spans $\mathbb{R}^N$. Thus a complete basis with $D = N$ is the smallest set of vectors that satisfies this condition, but so called *overcomplete* dictionaries where the number of elements exceeds the dimension of the signal space (i.e. $D > N$) are possible. In general we can then decompose $\boldsymbol{x}$ as

$$\boldsymbol{x} = \sum_{i=1}^{D} \boldsymbol{\psi}_i s_i = \boldsymbol{\psi} \boldsymbol{s}. \tag{1.1}$$

A sparse representation of a signal $\boldsymbol{x}$ is one that concentrates most of its energy in only a small number of the coefficients in $\boldsymbol{s}$. It will therefore be based on only a small number of atoms $\boldsymbol{\psi}_i$. A signal representation that involves only a small number of components and achieves the same accuracy as one with more components, is simpler and can be considered as a better explanation of the signal. If $\boldsymbol{\Psi}$ is overcomplete, a unique best representation exists only under some conditions and finding the most compact $\boldsymbol{s}$ can be a challenging task. However, the number of components is an objective we can optimize for and, as we will see, sufficiently good representations can be found easier than one might expect.

On a side note, it is also suggested that the efficiency of the human visual system and the related learning processes are to a big extent based on sparsity, for instance in the excitation of neurons [32]. All this motivates the concept of parsimony or sparsity for signal representations and indeed it has been successfully applied to a large number of relevant problems. For instance to image denoising [29], where thresholding in a sparse representation keeps only the vectors that best explain a signal but discards the noise. The concentration in a small number of coefficients also benefits the learning in support- and relevance-vector machines (SVM and RVM) [49].

There often arises a trade-off between efficient coding and an accurate approximation. To quantify this more precisely we introduce the *best K-term approximation* of a signal as the representation with at most K non-zero coefficients that minimizes the error in the mean-square sense.

1.1.2 Choice of dictionary

To some extend we can say that every interesting signal is sparse – if only we can find the right way to define it. But how to choose a good dictionary? We first study the use of full-rank bases followed by more general dictionaries to find good approximations of that kind.

Adaptive bases

If we depart from a known class of signals defined by a large set of given examples, we would like to find the basis that leads to the sparsest representation. If the class of signals in question are natural images we could for instance take a high number of randomly selected patches as an input.

The typical solution to this problem is the *Karhunen-Loève transform* (KLT). It departs from the assumption that the elements $\{\boldsymbol{a}_i\}$ of the multivariate dataset $\mathbf{A} \in \mathbb{R}^{N \times M}$ are correlated and defines an orthonormal basis such that its spanning vectors are oriented in the directions of highest variance. These directions are referred to as principal components and the procedure is also known as *principal component analysis* (PCA) in the context of machine learning. The first principal component is chosen to point in the direction of highest variability. The second component does the same within the subspace orthogonal to the first component and so on.

The KLT can be computed as either a singular value decomposition (SVD) of the data matrix if available or as an eigen-decomposition of its covariance matrix if the signal model is given by its distribution.

The SVD of a matrix $\mathbf{A} = [\boldsymbol{a}_1 \ldots \boldsymbol{a}_N]$ centered around the origin (i.e. $\sum_i \boldsymbol{a}_i = \mathbf{0}$) is given by

$$\mathbf{A} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^{\mathsf{T}}$$

such that $\boldsymbol{\Sigma} \in \mathbb{R}^{N \times M}$ is a diagonal matrix with non-negative elements, $\mathbf{U} \in \mathbb{R}^{N \times N}$ and $\mathbf{V} \in \mathbb{R}^{M \times M}$ are orthogonal matrices. The diagonal elements of $\boldsymbol{\Sigma}$ are called singular values of $\mathbf{A}$ and arranged in decreasing order. The KLT is then given by

$$\mathrm{KLT}\{\mathbf{A}\} = \mathbf{T} = \boldsymbol{\Sigma}\mathbf{V}^{\mathsf{T}}.$$

If we reduce $\mathbf{T}$ by taking only the first K components and denote this as

$$\boldsymbol{y}_K = \mathbf{T}_K^{\mathsf{T}} \boldsymbol{x} \ \in \ \mathbb{R}^K,$$

then the K-term approximation of a random vector $\boldsymbol{x}$ following the signal distribution is given by the projection

$$\hat{\boldsymbol{x}} = \mathbf{T}_K \, \boldsymbol{y}_K = \mathbf{T}_K \mathbf{T}_K^{\mathsf{T}} \boldsymbol{x}$$

It has minimal distortion in the mean square sense (i.e. the expectation of $\|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2$ is minimal) if $\mathbf{T}$ is a KLT basis. The proof is based on the fact that the KLT diagonalizes the covariance matrix of $\mathbf{A}$ and is provided in [29]. Hence the best K-term approximation in a KLT basis is given by the first K terms because the components are arranged in the order of decreasing eigenvalues.

Fixed bases

The covariance matrix of a class of signals is not always known a priori, and even if it is, calculating the KLT is of high computational complexity. Furthermore applying the transform requires the multiplication with a dense matrix. For these reasons we are sometimes also interested in finding better and fixed transforms that come close enough to a KLT and its optimality properties.

For image patches the discrete cosine transform (DCT) is heuristically a good approximation of the KLT in this sense. It is further a separable transform and can be implemented in an efficient way based on a fast Fourier transform (FFT).

Another popular family of orthonormal bases are wavelets. They have the property of well approximating images that are dominated by piecewise regular areas and their scale based design is well adapted to the structure of images. Furthermore they can be implemented with fast algorithms and the geometric interpretation of the coefficients comes in handy for an efficient coding. This has lead to successful applications in compression, notably the practical schemes JPEG2000 and SPIHT [38].

Overcomplete dictionaries

The bases discussed so far are not invariant to geometric transformations, in particular affine transformations such as translation or rotation. If a signal is composed by a set of basis vectors, a shifted version of that signal for instance can not simply be represented by shifted versions of the basis vectors because the latter are not part of the dictionary. If this was the case, we could greatly simplify the estimation of the transform of the signals because they could simply be represented by manipulating the coefficient indices. This has led to so called *structured dictionaries* [31]. However, it is still a partially open problem how they can be learned.

A different category of dictionaries based on overcomplete discrete cosine or wavelet transforms has successfully been applied to image denoising problems [16]. Another approach is to combine a K-means (or Lloyd-Max) like algorithm with an SVD to adaptively learn good dictionaries and is known as K-SVD [2].

Finally, if a signal is a superposition of multiple components where each is sparse in a different basis, the corresponding bases can be carefully combined to build a dictionary over which the superposition can still be sparsely represented.

1.1.3 Coherence

In the limit case, an infinite dictionary could contain all possible signals and any signal could be represented by just a single non-zero coefficient. Because this obviously does not lead to a practical scheme, we can expect an optimal dictionary size somewhere in between.

A definition commonly introduced to study the properties of a dictionary is the *coherence parameter* μ. It is defined as the cosine of the angle between the two closest atoms

$$\mu(\boldsymbol{\Psi}) = \max_{i \neq j} |\langle \boldsymbol{\psi}_i, \boldsymbol{\psi}_j \rangle|$$

of a dictionary $\boldsymbol{\Psi}$ with normalized atoms $\|\boldsymbol{\psi}_i\| = 1 \; \forall i$. For an orthonormal basis μ is zero.

We can describe a dictionary with a comparably small value of μ as *incoherent* and use this as a heuristic for sparse approximations to be easy to find. If a signal has a K-term approximation with respect to the dictionary $\boldsymbol{\Psi}$ that satisfies

$$K < \frac{1}{2}\left(1 + \frac{1}{\mu(\boldsymbol{\Psi})}\right),$$

then this representation is the unique, sparsest representation in this dictionary [10].

1.2 Inverse Problems

In a very general sense we can define an *inverse problem* as a problem where the effect is known and we are asked to conclude on what caused it. Before solving such a problem we will need to, first, model the *forward process* and second, *parameterize* the system with a minimal set of values that completely describe its state [47].

Usually a system can be parameterized in different ways. The possible parameterizations are equivalent if they can be related by a bijective transform. All possible and distinct sets of parameter values will form an abstract space of their own, referred to as a *manifold* which is essentially independent of the chosen parameterization. Each point on this manifold will then define a possible state of the system. The space spanned by a parameterization is called the *model space*. It is complemented by the *data space* containing the possible observations after the forward action. These physically measurable entities can again be parameterized in different ways and form a second manifold. Whether the model and data spaces are distinct, overlap or even coincide greatly depends on the problem in question.

Typical examples for inverse problems in image processing are tomography, deconvolution, denoising and inpainting.

1.2.1 Regularization

The corresponding forward problem is often surjective (an onto mapping) and more than one cause could have led to the same effect. In such a case we can say that multiple causes would be *consistent* with the effect. This underdetermined character leads us to the definition of the *well-posedness* of a problem as introduced by J. Hadamard [22]. It requires three conditions to be met: The existence and uniqueness of a solution as well as its continuous dependency on the data on some manifold. Only this really motivates a theory of inverse problems of its own merit.

In many image processing problems the forward process can be modeled exactly (either with filters or, for inpainting and quantization problems, with their loss-introducing function) or stochastically (for denoising). A solution almost always exists for the simple reason that the data we have at hands derives from it; be it in the form of another image or the physical reality.

However, there are still many possible solutions and in order to find a good one we must necessarily introduce some additional conditions to reach it in a unique and stable way. We can either use additional knowledge (if available) or enforce a certain signal model to get the sought after solution or one with desirable properties. This step is then called *regularization*.

A possible way of regularization introduced by A. Tikhonov is to minimize the ℓ_2 norm of an analysis operator $\mathbf{A}$. If $\mathbf{U}$ transforms $\boldsymbol{x}$ into the model space to $\boldsymbol{y} = \mathbf{U}\boldsymbol{x} + \boldsymbol{n}$, then we want to recover an $\hat{\boldsymbol{x}}$ that satisfies $\|\mathbf{U}\hat{\boldsymbol{x}} - \boldsymbol{y}\|_2^2 \leq \epsilon$ where ϵ depends on the level of the noise $\boldsymbol{n}$. A Tikhonov regularization now requires an operator $\mathbf{A}$ that leads to small energies for good signals. The signal can then be estimated as the result of the convex optimization

$$\begin{aligned} \hat{\boldsymbol{x}} = \arg\min_{\hat{\boldsymbol{x}}} \|\mathbf{A}\hat{\boldsymbol{x}}\|_2^2 \\ \text{subject to } \ \|\mathbf{U}\hat{\boldsymbol{x}} - \boldsymbol{y}\|_2^2 \leq \epsilon. \end{aligned} \tag{1.2}$$

The solution to (1.2) can be given explicitly by a linear estimator as discussed by Mallat et al. [29].

For cases where it is easier to estimate $\|\mathbf{A}\boldsymbol{x}\|$ than ϵ we can reformulate (1.2) as

$$\begin{aligned} \hat{\boldsymbol{x}} = \arg\min_{\hat{\boldsymbol{x}}} \|\mathbf{U}\hat{\boldsymbol{x}} - \boldsymbol{y}\|_2^2 \\ \text{subject to } \ \|\mathbf{A}\hat{\boldsymbol{x}}\|_2^2 \leq \gamma. \end{aligned} \tag{1.3}$$

For every ϵ there is a corresponding γ relating the two formulations, but this equivalence is seldom a trivial one.

1.2.2 Total Variation

As natural images originate from projections of physical objects with clear boundaries onto the image plain, they can themselves be expected to consist of locally homogeneous regions delimited by sharp contours. This motivates the assumption of a piecewise smooth image model and enforcing it can be used as a regularization method for such images. The *total variation* introduced here quantifies this regularity. In the following we study scalar fields (e.g. luminance images) only, but extensions to vector valued fields are possible.

The variation of a grayscale image $\boldsymbol{x}$ is given by the magnitude of the gradient. The discrete gradient at a location (i, j) can for instance be given by the first order difference

$$(\nabla \boldsymbol{x})_{i,\,j} = \begin{pmatrix} (D_x \boldsymbol{x})_{i,j} \\ (D_y \boldsymbol{x})_{i,j} \end{pmatrix} = \begin{pmatrix} x_{i,j+1} - x_{i,j} \\ x_{i+1,j} - x_{i,j} \end{pmatrix}$$

within the image and zero outside. The variation at (i, j) is $|(\nabla \boldsymbol{x})_{i,j}|$ and hence we define the total variation as the sum over the whole image

$$\|\boldsymbol{x}\|_{\mathrm{TV}} = \frac{1}{N} \sum_{i,j} |(\nabla \boldsymbol{x})_{i,j}| \,. \tag{1.4}$$

In other words, the total variation is the ℓ_1 norm of the gradient of an image. Usually the variation is the euclidean magnitude of the gradient

$$|(\nabla \boldsymbol{x})_{i,j}| = \|(\nabla \boldsymbol{x})_{i,j}\|_2 = \sqrt{\left((D_x \boldsymbol{x})_{i,j}\right)^2 + \left((D_y \boldsymbol{x})_{i,j}\right)^2}$$

and as such rotation invariant and anisotropic. However, it can be (and often is) replaced by the isotropic

$$|(\nabla \boldsymbol{x})_{i,j}| = \|(\nabla \boldsymbol{x})_{i,j}\|_1 = |(D_x \boldsymbol{x})_{i,j}| + |(D_y \boldsymbol{x})_{i,j}|$$

instead.

One class of algorithms to solve TV minimization problems is called iterative shrinkage and thresholding (IST). Bioucas-Dias et al. [6] propose a faster modification called two-step IST (TwIST) and Beck et al. [5] presented the fast iterative shrinkage-thresholding algorithm (FISTA) which reaches the optimal global convergence rate. Other algorithms based on second order methods have a fast convergence rate but the high complexity for each step leads to impractical run times for big problem sizes.

The drawbacks of TV regularization are a possible loss of texture information or other small scale features (because they have a high variation), a loss of contrast and geometric distortion as well as the “staircase” effect that is caused because piecewise constant areas are favored. Even the fastest known TV minimization algorithms still exhibit a fairly high computational complexity.

The toy example in Fig. 1.1 on the facing page shows an image at different levels of total variation. We see that TV minimization preserves the edges and contours but leads to a loss of texture and details. The problem was formulated in a Lagrangian way as

$$\hat{\boldsymbol{x}} = \arg\min_{\hat{\boldsymbol{x}}} \|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2 + \lambda \, \|\hat{\boldsymbol{x}}\|_{\mathrm{TV}}$$

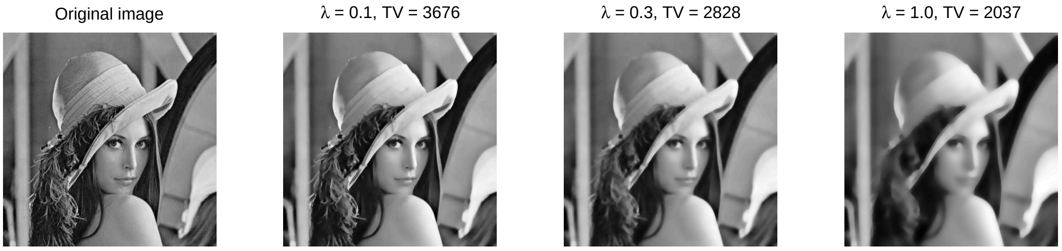


Figure 1.1: *Image* Lena *[45] (a) original and (b – d) with minimized total variation at different levels.*

where the parameter λ determines the desired level of the total variation.

1.3 Algorithms

Given a dictionary $\boldsymbol{\Psi}$ and a signal $\boldsymbol{x}$ we are looking for a sparse representation $\boldsymbol{s} \in \mathbb{R}^M$ that closely approximates $\boldsymbol{x} \approx \boldsymbol{\Psi s}$. In other words, we want to find a best K-term approximation (where $K = |\{s_i | s_i \neq 0\}|$ is the cardinality of the support of $\boldsymbol{s}$) with respect to the norm $\|\boldsymbol{x} - \boldsymbol{\Psi s}\|_2$.

In the case of a complete basis the solution is unique and given by the inverse of $\boldsymbol{\Psi}$. If a fast transform exists the complexity can be further reduced to $\mathcal{O}(N \log N)$ for the FFT and related transforms or $\mathcal{O}(N)$ for wavelet transforms with compact support. As discussed above in a KLT basis the first K terms also give the best K-term approximation.

Although decoding from a representation in an overcomplete dictionary is easily done by applying the multiplications of (1.1), the inverse problem of finding the best representation is a hard problem of combinatorial nature. For this reason exactly solving for the sparsest representation can only be considered for small problem sizes. Because most image processing problems introduce a high dimensional space, sub-optimal but efficient algorithms have been proposed. The most notable variants are basis pursuit (BP), matching pursuit (MP) and orthogonal matching pursuit (OMP).

In the following we assume all dictionaries to be normalized such that $\|\boldsymbol{\psi}_k\| = 1 \; \forall \, k$.

1.3.1 Matching pursuit

The greedy matching pursuit algorithm is a fast approach. It iteratively selects the atom that correlates most with the signal, calculates the residue $\boldsymbol{r}$ and continues until K coefficients are extracted or the residue reaches zero. Any remaining coefficients are left to zero.

However, this algorithm can happen to select inappropriate atoms requiring a lot of additional vectors to correct for the wrong choice made before.

Algorithm 1: Matching Pursuit

```
Data: $\boldsymbol{\Psi}$, $\boldsymbol{x}$
Result: $\boldsymbol{s}$
Initialization: $\boldsymbol{r}^{(1)} = \boldsymbol{x}$, $\boldsymbol{s} = \boldsymbol{0}$
for $0 < i \leq K$ do
    $a_l = \left\langle \boldsymbol{\psi}_l, \boldsymbol{r}^{(i)} \right\rangle \quad \forall\, l$
    $k = \arg\max_l |a_l|$
    $\boldsymbol{r}^{(i+1)} = \boldsymbol{r}^{(i)} - a_k \boldsymbol{\psi}_k$
    $s_k = a_k$
end
```

1.3.2 Orthogonal matching pursuit

A second algorithm reducing this risk is the orthogonal matching pursuit [53] which is similar in spirit to MP, but – analog to a Gram-Schmidt process – orthogonalizes the dictionary after each iteration. This guarantees that no components in the direction of previously selected atoms are introduced as it can be the case with MP.

Let us introduce the set Γ to accumulate the indices of selected atoms such that $i \in \Gamma \Leftrightarrow s_i \neq 0$.

Algorithm 2: Orthogonal Matching Pursuit

```
Data: $\boldsymbol{\Psi}$, $\boldsymbol{x}$
Result: $\boldsymbol{s}$
Initialization: $\Gamma = \emptyset$, $\boldsymbol{r}^{(0)} = \boldsymbol{x}$
for $0 \leq i < K$ do
    $a_l = \left\langle \psi_l, r^{(i)} \right\rangle \quad \forall\, l \notin \Gamma^{(i)}$
    $l_{\max} = \arg\max_l |a_l|$
    $\Gamma^{(i+1)} = \Gamma^{(i)} \cup \{l_{\max}\}$
    $\boldsymbol{s}^{(i)}_{\Gamma^{(i)}} = \boldsymbol{\Psi}^{\dagger}_{\Gamma^{(i)}} \boldsymbol{r}^{(i+1)}$
    $\boldsymbol{r}^{(i+1)} = \boldsymbol{r}^{(i)} - \boldsymbol{\Psi} \boldsymbol{s}^{(i)}$
end
$\boldsymbol{s} = \boldsymbol{s}^{(K)}$
```

The algorithm necessarily stops after $K \leq N$ iterations with a zero residue because all dimensions are covered at this point. It has an exponential convergence rate. The cost of each iteration can be reduced by using a QR factorization of $\boldsymbol{\Psi}$ for the orthogonalization step, but the complexity is still much higher than for MP.

It is worth noting that OMP does not minimize the residual error at each step. It does, however, minimize the residual error given the selected atoms.

1.3.3 Basis pursuit

Basis pursuit relaxes the problem by approximating the ℓ_0 norm with the convex ℓ_1 norm. The approximation of the signal in a dictionary can then be seen as an underdetermined inverse problem that is regularized by the ℓ_1 norm. It involves all coefficients at once and can be implemented as a linear program (LP). Linear programs are a category of problems that can be put into their canonical form as

$$
\begin{aligned}
(\mathbf{LP}): \qquad & \hat{\boldsymbol{x}} = \arg\max_{\boldsymbol{x}} \ \boldsymbol{c}^{\mathsf{T}}\boldsymbol{x} \\
& \text{subject to} \ \ \mathbf{A}\boldsymbol{x} \preceq \boldsymbol{b}.
\end{aligned}
$$

We discuss the implications of this approach in more detail in the context of compressed sensing in Sec. 1.4.3 on page 13.

Overcomplete dictionaries are inherently redundant and the related algorithms will likely perform redundant calculations as well. A possible cure is to use a *divide and conquer* strategy, for instance by using a tree-based implementation or by splitting the dictionary into incoherent subparts to reduce the cost of the search in each iteration step. In some cases this will improve the runtime without penalizing the approximation performance too much.

1.4 Compressed Sensing

Compressed sensing (CS) is a method to perfectly reconstruct a signal $\boldsymbol{x} \in \mathbb{R}^N$ from less than N non-adaptive, linear projections. It relies essentially on the possibility to represent $\boldsymbol{x}$ sparsely in a known basis $\boldsymbol{\Psi} \in \mathbb{R}^{N\times N}$ with only $K \ll N$ non-zero coefficients.

1.4.1 Johnson-Lindenstrauss Embedding

We start the discussion of compressed sensing with a result concerning low distortion embeddings of points in a high dimensional euclidean space into a lower dimensional one.

Lemma 1 (Johnson and Lindenstrauss [26])

> Given $0 < \epsilon < 1$ and an integer K, let $M \in \mathbb{N}$ be $M \geq M_0 = \mathcal{O}(\epsilon^{-2} \log K)$. For every set $\mathcal{P}$ of K points in $\mathbb{R}^N$ there exists $f : \mathbb{R}^N \to \mathbb{R}^M$ such that for all $\boldsymbol{u}, \boldsymbol{v} \in \mathcal{P}$
>
> $$(1-\epsilon)\left\|\boldsymbol{u}-\boldsymbol{v}\right\|^2 \leq \left\|f(\boldsymbol{u})-f(\boldsymbol{v})\right\|^2 \leq (1+\epsilon)\left\|\boldsymbol{u}-\boldsymbol{v}\right\|^2.$$

Thus it is possible to embed k points from an n dimensional space into an $m \leq n$ dimensional one while preserving the distance between the points up to an arbitrary constant ϵ.

Besides the original work, other proofs were also given by Gupta and Dasgupta [21] or Frankl and Meahara [18]. The latter give a a constructive proof with an explicit value for

$$m_0 = \left\lceil 9(\epsilon^2 - \frac{2\epsilon^3}{3})^{-1} \log |\mathcal{P}| \right\rceil + 1$$

by considering a scheme with projection on random orthonormal vectors.

This result has applications in various fields where the dimensionality of the problem determines the computational complexity. Reducing the dimensionality can for instance benefit machine learning applications. And more importantly compressed sensing can be seen as such an embedding.

1.4.2 Measurements

As before we denote the representation of $\boldsymbol{x} \in \mathbb{R}^N$ over $\boldsymbol{\Psi} \in \mathbb{R}^{N \times N}$ as

$$\boldsymbol{x} = \sum_{i=1}^{N} \boldsymbol{\psi}_i \, s_i = \boldsymbol{\Psi s}$$

where $\boldsymbol{s} \in \mathbb{R}^N$ is the K-sparse coefficient vector.

Instead of directly working with the signal $\boldsymbol{x}$, we project it onto a different space of lower dimension M with $\boldsymbol{\Phi} \in \mathbb{R}^{M \times N}$. In this context the components of the resulting vector

$$\boldsymbol{y} = \boldsymbol{\Phi x} = \boldsymbol{\Phi \Psi s} \in \mathbb{R}^M$$

are then called *measurements*.

The compressed sensing theory now states that if $\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$ are sufficiently incoherent (meaning that the columns $\{\boldsymbol{\psi}_i\}$ of $\boldsymbol{\Psi}$ can not represent sparsely the rows of $\boldsymbol{\Phi}$) and if $\boldsymbol{s}$ is sufficiently sparse, we can recover the signal $\boldsymbol{x}$ from its measurements $\boldsymbol{y}$ with high probability even though the system $\boldsymbol{y} = \boldsymbol{\Phi x}$ is highly underdetermined in terms of linear algebra.

Many pairs of bases are known to be incoherent, so for example the Fourier and Wavelet bases [10]. In particular a basis built from i.i.d. random draws from a Gaussian or a Bernoulli distribution will be incoherent with any other fixed basis with a high probability in high dimensions. A basis drawn from a Gaussian distribution is in a sense universal: Because of the rotational symmetry of the distribution no direction is favored a priori and it can be used with any dictionary with the same high probability.

A sufficient condition linking the matrices $\boldsymbol{\Phi}$, $\boldsymbol{\Psi}$ and the sparsity K is the Restricted Isometry Property (RIP).

Definition 1 (Restricted Isometry Property)

If for all K-sparse $\boldsymbol{s} \in \mathbb{R}^N$, there exists $0 \leq \delta_K < 1$ such that

$$(1-\delta_K)\,\|\boldsymbol{s}\|_2^2 \;\leq\; \|\mathbf{A}\boldsymbol{s}\|_2^2 \;\leq\; (1+\delta_K)\,\|\boldsymbol{s}\|_2^2$$

we say that $\mathbf{A}$ satisfies the RIP of order K with radius δ_K.

In other words, we ask that all submatrices with up to K columns of $\boldsymbol{\Psi}$ are close to isometries and as such approximatively distance preserving.

Theorem 1 (Perfect Recovery Condition, Candès and Tao [10])

If $\mathbf{A}$ satisfies the RIP of order $2K$ with radius δ_{2K}, then for any K-sparse signal $\boldsymbol{s}$ sensed by $\boldsymbol{y} = \mathbf{A}\boldsymbol{s}$, $\boldsymbol{s}$ is with a high probability perfectly recovered by the ideal program

$$(\mathbf{P_0}): \qquad \begin{aligned} &\hat{\boldsymbol{s}} = \arg\min_{\boldsymbol{s}} \|\boldsymbol{s}\|_0 \\ &\text{subject to } \hat{\boldsymbol{y}} = \boldsymbol{\Phi}\boldsymbol{\Psi}\boldsymbol{s} \end{aligned} \tag{1.5}$$

and unique.

A complete proof is given ibidem. The uniqueness property can be shown by contradiction: If $\boldsymbol{s}$ and $\boldsymbol{t}$ are two different K-sparse representations for $\boldsymbol{x}$ for which the RIP of order $2K$ holds, then $\boldsymbol{s}-\boldsymbol{t}$ is at most $2K$-sparse and

$$(1-\delta_{2K})\,\|\boldsymbol{s}-\boldsymbol{t}\|_2^2 \;\leq\; \|\mathbf{A}\boldsymbol{s}-\mathbf{A}\boldsymbol{t}\|_2^2 \;=\; \|\boldsymbol{y}-\boldsymbol{y}\|_2^2 \;=\; 0.$$

Only $\boldsymbol{s} = \boldsymbol{t}$ can satisfy this condition and hence $\boldsymbol{s}$ is the unique K-sparse solution. □

Although it is a hard (NP-complete) problem to exactly verify the RIP for a given matrix, it has been shown [10] that Gaussian random matrices satisfy it with high probability if

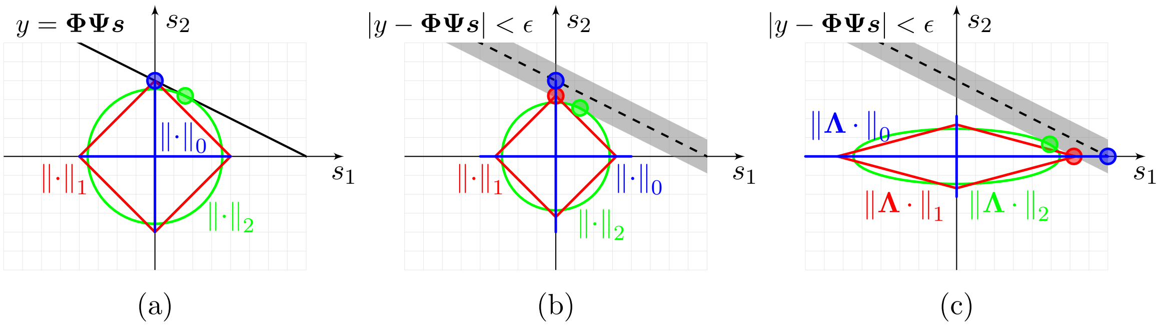
$$M \geq c\,K \ln\left(\frac{N}{K}\right) \tag{1.6}$$

is satisfied for a constant c depending only on δ and with a high probability that exponentially tends towards 1 as N increases.

1.4.3 Signal recovery

If the number of measurements M is chosen sufficiently big according to (1.6), then the minimization $(\mathbf{P_0})$ defined in (1.5) is shown to find the unique solution $\hat{\boldsymbol{s}} = \boldsymbol{s}$ [10]. Once $\hat{\boldsymbol{s}}$ is found we can project it back to get $\hat{\boldsymbol{x}} = \boldsymbol{\Psi}\hat{\boldsymbol{s}}$ in the original domain.

However, this minimization problem is again of combinatorial nature and therefore computationally impossible for all but the smallest problem sizes. In particular, applications to images with their high number of dimensions would not be possible. But luckily above problem $(\mathbf{P_0})$ can be relaxed and reformulated as a convex optimization!



$y = \mathbf{\Phi\Psi}\boldsymbol{s}$
s_2
$\|\cdot\|_0$
$\|\cdot\|_1$
s_1
$\|\cdot\|_2$
$|y - \mathbf{\Phi\Psi}\boldsymbol{s}| < \epsilon$
s_2
$\|\cdot\|_1$
$\|\cdot\|_0$
s_1
$\|\cdot\|_2$
$|y - \mathbf{\Phi\Psi}\boldsymbol{s}| < \epsilon$
s_2
$\|\mathbf{\Lambda}\cdot\|_0$
s_1
$\|\mathbf{\Lambda}\cdot\|_1$
$\|\mathbf{\Lambda}\cdot\|_2$

(a) (b) (c)

Figure 1.2: *Illustration of ℓ_2 (green), ℓ_1 (red) and ℓ_0 (blue) minimization for (a) equality constraints, (b) quadratically bounded constraints and (c) quadratic bounds with reweighted norms. The constraints are shown in black and gray and the respective minimal ℓ_p balls in color. (Inspired by [11])*

Among all the ℓ_p norms, the one with the smallest value of $p \in \mathbb{R}$, which still satisfies the triangle inequality and hence is a norm and convex, is $\|\cdot\|_1$. Therefore it is a good candidate to replace the $\|\cdot\|_0$ pseudo-norm and to restate the problem as a constrained convex optimization problem.

Theorem 2 ($\ell_0 - \ell_1$ equivalence, Donoho and Huo [13])

If $\mathbf{\Phi\Psi}$ satisfies the RIP of order $2K$ with radius $\delta_{2K} < \sqrt{(2)} - 1$, then

$$(\mathbf{P_1}): \quad \begin{aligned} &\hat{\boldsymbol{s}} = \arg\min_{\boldsymbol{s}} \|\boldsymbol{s}\|_1 \\ &\text{subject to } \hat{\boldsymbol{y}} = \mathbf{\Phi\Psi}\boldsymbol{s} \end{aligned} \tag{1.7}$$

is equivalent to $(\mathbf{P_0})$ and will find the same unique $\hat{\boldsymbol{s}}$.

This relaxed problem $(\mathbf{P_1})$ is again the basis pursuit and can be solved by standard optimization approaches such as a linear program. It is a rather surprising result that the two problems are exactly equivalent under some conditions and this is one of the reasons for the popularity of compressed sensing.

We can illustrate the intuition behind the different optimization problems by visualizing a contrived case where $N = 2$ and $M = K = 1$. Figure 1.2 shows the space of $\boldsymbol{s} \in \mathbb{R}^N$ where the black line represents the subspace of all possible $\boldsymbol{s}$ leading to a given measurement value. However only 2 of them are 1-sparse and both ℓ_0 and ℓ_1 minimization will find the exact solution in the noiseless case (a) and come close even with noise in (b) while for both cases an ℓ_2 minimization would yield completely different results. Part (c) will be discussed in Sec. 1.4.5 below.

In practice the minimal ratio M/K is sufficiently small (typically in the order of 2 to 4) for applications to be possible.

1.4.4 Compressible Signals

These findings do not only hold for exactly sparse signals, but can also be applied to *compressible signals* where the coefficient magnitudes decay sufficiently fast, but do not necessarily reach zero. If such a representation exists, the signal can be approximated by a best K-term approximation using a thresholded coefficient vector. In particular if the sorted magnitudes $|s|_{(i)}$ of the coefficients closely follow a power-law such that

$$|s|_{(i)} < C\, i^{-\alpha} \tag{1.8}$$

for some constants C and α we can relax [10] above problem $(\mathbf{P_1})$ to solve

$$\begin{aligned}(\mathbf{P_{QC}}): \qquad & \hat{\boldsymbol{s}} = \arg\min_{\boldsymbol{s}} \|\boldsymbol{s}\|_1 \\ & \text{subject to } \|\hat{\boldsymbol{y}} - \boldsymbol{\Phi\Psi s}\|_2 < \epsilon\end{aligned} \tag{1.9}$$

instead.

This formulation can also be used if the measurements are affected by noise $\boldsymbol{e}$ such that $\hat{\boldsymbol{y}} = \boldsymbol{y} + \boldsymbol{e}$. In which case $(\mathbf{P_{QC}})$ is called basis pursuit denoising (BPDN). The relaxation parameter ϵ needs to be chosen carefully. Let us assume that the noise is Gaussian with $e_i \sim \mathcal{N}(0, \sigma^2)$ i.i.d. and that we know its variance σ^2. Then, even if we knew the perfect $\hat{\boldsymbol{x}} = \boldsymbol{x}$ by oracle, we had an error of at least $\epsilon_0 = \|\boldsymbol{\Phi}\hat{\boldsymbol{x}} - \hat{\boldsymbol{y}}\|_2 = \|\boldsymbol{e}\|_2$ with an expected value of $E(\epsilon_0) = \sqrt{M}E(e_i) = \sqrt{M}\sigma$. Thus a practical ϵ needs to be chosen bigger and a value proposed by Candès [9] is

$$\epsilon = \sigma\sqrt{M}\sqrt{1 + 2\sqrt{2/M}} \geq \epsilon_0\,. \tag{1.10}$$

Robust recovery for compressed sensing is therefore still possible with noise as we have to expect it in any practical application.

This is the reconstruction method we will use in the following (See Sec. 2.2.2 on page 22).

1.4.5 Extensions

Additional constraints

CS decoding can be combined with additional constraints beyond sparsity. In some cases the positivity of the coefficients can be enforced, the scale structure of wavelets can be exploited or the total variation can be taken into account. Of particular interest is the effect of quantization on CS. Basis pursuit dequantization [24] and optimal quantizer design for CS have been studied before [46].

Reweighting

First, we show how previous knowledge about the distribution of the image coefficients in the transform domain can improve the performance. The intuition behind this approach is, that if we knew the perfect solution $\boldsymbol{x}$ by oracle, we could define the diagonal weighting matrix $\boldsymbol{\Lambda}$ as

$$\Lambda_{ii} = \frac{1}{|s_i| + \gamma}$$

such that the reweighted ℓ_1 norm

$$\|\boldsymbol{\Lambda} s\|_1 = \sum_i |\Lambda_{ii} s_i| = \sum_i \left| \frac{s_i}{|s_i| + \gamma} \right| \approx \left|\{s_i | s_i \neq 0\}\right| = \|\boldsymbol{s}\|_0 \ .$$

would actually approximates the ℓ_0 norm. Above, the parameter $0 < \gamma \ll 1$ serves for regularization of s_i close or equal to zero. The reweighted ℓ_1 minimization then becomes

$$\begin{aligned} (\mathbf{P_{RW}}): \qquad & \hat{\boldsymbol{s}} = \arg\min_{\boldsymbol{s}} \|\boldsymbol{\Lambda s}\|_1 \\ & \text{subject to } \|\hat{\boldsymbol{y}} - \boldsymbol{\Phi\Psi s}\|_2 < \epsilon \end{aligned} \tag{1.11}$$

By introducing $\boldsymbol{q} = \boldsymbol{\Lambda s}$ we can equivalently write

$$\begin{aligned} & \hat{\boldsymbol{q}} = \arg\min_{\boldsymbol{q}} \|\boldsymbol{q}\|_1 \\ & \text{subject to } \left\|\hat{\boldsymbol{y}} - \boldsymbol{\Phi\Psi\Lambda}^{-1}\boldsymbol{q}\right\|_2 < \epsilon, \quad \hat{\boldsymbol{s}} = \boldsymbol{\Lambda}^{-1}\hat{\boldsymbol{q}} \end{aligned}$$

and reuse existing optimization algorithms without change [11]. This choice actually leads us to a norm similar to the Mahalanobis distance but for the ℓ_1 case.

Candès et al. [11] suggest using this approach iteratively to outperform unweighted ℓ_1 minimization even in the case of not exactly sparse signals as long as the exponent α in (1.8) remains below 1.

Part (c) of Fig. 1.2 on page 14 motivates the reweighting in cases where an estimate for s_i is known. There, the reweighted ℓ_p balls are drawn and we can see that in such cases the correct solution can be recovered that would otherwise be decoded incorrectly. It also implies that we do not need to have a very precise value of the scaling factors in order make use of this technique.

Joint sparsity

Joint Sparsity Models (JSM) are motivated by the fact that both the location and the values of the non-zero values in a sparse vector are unknowns. Hence if we already knew the support of a sparse signal we could drastically reduce the number of measurements required to decode it, because only the non-zero values would remain as unknowns. One way to reduce this number of unknowns is by assuming that the support of two or more subsequent signals is related.

For a sequence of subsequent signals $\{\boldsymbol{x}_j\}_j$ Baron et al. introduce three such models [4]:

- *JSM-1: Common sparse component plus innovations*: Each signal $\boldsymbol{x}_j$ is represented as the sum of a common component $\boldsymbol{x}_C$ and an innovation component $\Delta\boldsymbol{x}_j$ such that all of them have a sparse representation

  $$\boldsymbol{x}_C = \boldsymbol{\Psi}\boldsymbol{s}_C, \quad \|\boldsymbol{s}_C\|_0 = K_C$$

  and

  $$\boldsymbol{\Delta x}_j = \boldsymbol{\Psi}\Delta\boldsymbol{s}_j, \quad \|\Delta s_j\|_0 = K_j$$

  respectively.

- *JSM-2: Common sparse supports*: In this case all signals have different coefficients, but the support for all $\boldsymbol{s}_j$ is identical.

- *JSM-3: Non-sparse common component plus sparse innovations*: The last model is similar to the first one, but does not make any assumption on the sparsity of the common component that is equally superimposed on all signals.

The assumptions of JSM-1 can be exploited by solving a single linear program on

$$\tilde{\boldsymbol{y}} = \tilde{\boldsymbol{\Phi}}\tilde{\boldsymbol{\Psi}}\tilde{\boldsymbol{s}}$$

for all signals at once by defining

$$\tilde{\boldsymbol{s}} = (\boldsymbol{s}_C, \boldsymbol{s}_1, \cdots, \boldsymbol{s}_L)^\mathsf{T}, \quad \tilde{\boldsymbol{y}} = (\boldsymbol{y}_1, \cdots, \boldsymbol{y}_L)^\mathsf{T}$$

$$\tilde{\boldsymbol{x}} = (\boldsymbol{x}_1, \cdots, \boldsymbol{x}_L)^\mathsf{T} = (\boldsymbol{x}_C + \Delta\boldsymbol{x}_1, \cdots, \boldsymbol{x}_C + \Delta\boldsymbol{x}_L)^\mathsf{T}$$

$$\tilde{\boldsymbol{\Phi}} = \begin{pmatrix} \boldsymbol{\Phi}_1 & & \\ & \ddots & \\ & & \boldsymbol{\Phi}_L \end{pmatrix}, \quad \tilde{\boldsymbol{\Psi}} = \begin{pmatrix} \boldsymbol{\Psi} & \boldsymbol{\Psi} & & \\ \boldsymbol{\Psi} & & \ddots & \\ \boldsymbol{\Psi} & & & \boldsymbol{\Psi} \end{pmatrix}$$

Recovery strategies, in particular a simultaneous orthogonal matching pursuit, for the other two cases are given in [15].

Practical algorithms

Different algorithms to solve the compressed sensing problem have been proposed and are publicly available. Starting with ℓ_1-*magic* [9] that implements some of the most common problems as either linear or second order cone programs with a rather slow path-following primal-dual method, over gradient projection for sparse reconstruction (GPSR) [17] to the fast Nesterov's Optimal Gradient Method implemented in NESTA [8]. The latter smooths the ℓ_1 norm and iteratively reduces this barrier until convergence.

Because the product $\boldsymbol{\Phi}\boldsymbol{\Psi}$ itself can be seen as an overcomplete dictionary for the measurements, the orthogonal and the non-orthogonal matching pursuit can both be used for CS as well. They tend to be much faster than solutions to the basis pursuit formulation, but also less accurate thus requiring more measurement to achieve a comparable performance.

A comprehensive list of algorithms and toolboxes beyond this short introduction is maintained by the digital signal processing group at Rice university [37].

Applications

Other promising applications of compressed sensing include among others magnetic resonance imaging (MRI) [28], imaging with single pixel cameras using a digital micro-mirror device [14], spread spectrum receivers or data gathering in large wireless sensor networks [27]. There are many motivations for a low measurement rate, be it a shorter exposure to ionizing radiation of a patient, a higher throughput in a sensor network, higher bandwidth of an analog to digital converter (ADC) – possibly below the Nyquist rate – or just a lower energy consumption. Furthermore a random embedding can increase the decoder stability.

Outline

Within the scope of this thesis two different problems related to the concepts of sparse image representations presented above were studied. The following two chapters present them.

The first one investigates ways to use compressed sensing for lossy but stable communication over varying channels. The specific application of a video multicast is studied. This part of my work was previously summarized in the paper *Compressed Sensing Based Video Multicast* [40] and accepted for publication at the Visual Communications and Image Processing (VCIP) conference 2010. It forms the basis for the following chapter.

The second problem consists of reconstructing an image by using a coarsely quantized view and a previously known second view of the same scene at high quality. It is applied to stereo image pairs that were distributedly coded with JPEG, but jointly reconstructed. This scheme improves the dequantization while remaining consistent with the widely adopted JPEG image standard. This part of my work was also summarized in a short conference paper entitled *Joint Decoding of Stereo JPEG Image Pairs* and submitted to the International Conference on Image Processing (ICIP) 2010 as [41]. It is presented in chapter 3.

This thesis is concluded with a brief discussion of possible future work in chapter 4.

Chapter 2

Compressed Sensing Based Video Multicast

2.1 Motivation

We consider a scenario where a video is simultaneously transmitted to multiple receivers with different, wireless channels. In this configuration it is difficult to allocate a fixed rate for the encoder to guarantee a low distortion for all receivers.

Following a traditional approach we would separate the source and channel coding. Shannon's separation theorem [43] largely simplifies the problem of optimal coding by splitting it into two different subproblems and still guarantees optimality to be achievable. Even though it holds in many cases, it does not apply to the case of multi-user channels as we target it. Because such schemes are designed for a given channel capacity, all receivers who can meet the requirements will be able to decode the content at the same suboptimal quality imposed by the encoding while some with less favorable channels will fail to decode it at all for most of the time. Because of the design of both variable length codes and video codecs such as MPEG, even a single error can propagate for a long time and lead to a big distortion. This behavior starts rather abruptly when the channel capacity falls below the value used for the source-coder design and appears in all separation based multicast schemes. It is commonly referred to as *cliff effect*. Additionally, parameters of a wireless channel such as Signal to Noise Ratio (SNR), bandwidth or losses are prone to vary significantly between different receivers and over time, which further complicates the optimal resource allocation at the encoder.

This cliff effect and the varying channel statistics have led to the idea of joint source-channel coding (JSCC) where the source symbols are directly mapped to channel symbols by a single encoder. In this case a part of the distortion will be introduced by the channel rather than the encoding, leading to a graceful degradation with respect to the channel capacity. As videos are suitable for lossy compression it would be desirable to have a scheme which allows any receiver to decode the content with a quality corresponding to its channel and that at the same time remains efficient.

One way to practically implement a JSCC solution for video multicast was recently presented under the name *SoftCast* [25]. Its main idea is inspired by Peterson [35] who showed that a group of coefficients $\boldsymbol{s}_j$ with variance σ_j^2 should be scaled proportionally to $\sigma_j^{-1/2}$ (at the same time obeying some constraint on the total energy) before analog transmission through an additive white Gaussian noise (AWGN) channel in order to suffer the least distortion. *SoftCast* first decorrelates an image by applying a discrete cosine transform and then applies above scaling to groups of coefficients depending on their variances. They then compare their scheme with MPEG 4 and an MDC scheme at different, but fixed rates. They claim to have distortions comparable to the best out of those schemes at a given rate while having smooth degradation across rates. However, only intra-coding is used by the approach and for the comparisons.

An alternative way to multicast to receivers with different channels is unequal error protection [52]. The source is encoded into two or more streams of decreasing importance, starting for instance with the motion vectors or a low resolution stream. The channel coding is then chosen such that the basic stream is decodable by all receivers thanks to strong error correcting codes, while the enhancements have less protection and can only be decoded after high capacity channels. This does not lead to a continuous degradation but rather introduces multiple "cliffs". Our goal with JSCC should however be to achieve a continuous degradation curve.

The main contribution presented in this chapter is a practical scheme for wireless video multicast. It is based on the theory of compressed sensing to generate measurements of equal importance from the content. We choose a compressed sensing based approach because it allows us to recover signals with a sparse representation even if some of the measurements are lost or distorted.

We first compare various combinations of measurement matrices in experiments for this application and evaluate the performance with respect to noise and loss levels in the channel. We further introduce two possible enhancements to the basic scheme. In the first case the encoder transmits additional side information about the distribution of the sparse coefficients which helps to improve the reconstruction quality. In the second the inter-frame correlation of videos is exploited at the decoder side without changing the encoder design. Finally we use simulations to compare the scheme with the above mentioned *SoftCast* as a main benchmark and find better results for some region of operating points.

This chapter is based on the brief introduction to compressed sensing given before in Sec. 1.4. The proposed scheme is described in in the following section and followed by an analysis of the experimental results in Sec. 2.3.

2.2 Proposed Scheme

In our scheme (Illustrated by Fig. 2.1 on the next page) a single encoder transforms a video into a number of measurements through multiplication with the random measurement matrix. These measurements are then directly transmitted towards the receivers over possibly very different channels distorting the signal with losses and noise. Finally each decoder will use the measurements he receives to reconstruct the sparsest signal in some basis such that

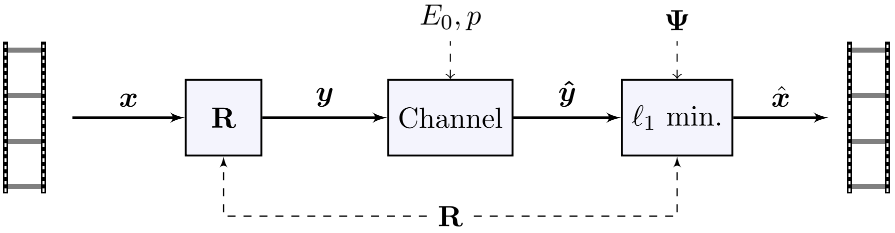


Figure 2.1: *Flow graph of the proposed scheme (only one channel and receiver shown).*

the decoded signal is compatible with the measurements. Most natural images are compressible in some bases, given by the DCT, wavelet transform or possibly others. This implies that a sparse approximation exists and makes our approach possible.

2.2.1 Encoding

In our application a grayscale image is split into subblocks of size $n \times n$. A block is represented as a column vector of length $N = n^2$ by concatenating its columns. The encoder works on each block $\boldsymbol{x}$ of each frame of a video. It first subtracts the mean value from a frame and then applies the full random matrix $\mathbf{R} \in \mathbb{R}^{N \times N}$ to $\boldsymbol{x}$. Those measurements are then interleaved across blocks and assembled into packages. This ensures that lost packages will not erase a single block completely but rather affect all of them evenly.

We have evaluated different random matrices $\mathbf{R}$ and sparse bases $\boldsymbol{\Psi}$; in particular matrices drawn uniformly from a Gaussian distribution (called $\mathbf{G}$ in the following) with $G_{ij} \sim \mathcal{N}(0,1)/\sqrt{N}$, $\mathbf{G}^1$ derived from $\mathbf{G}$ by normalizing each column, $\mathbf{G}^{\perp}$ an orthogonalized $\mathbf{G}$, $\mathbf{B}$ drawn from a Bernoulli distribution with $B_{ij} \in \{-1, 1\}/\sqrt{N}$ and a Hadamard transform $\mathbf{H}$. In terms of sparse bases we compared the discrete cosine transform basis $\mathbf{D}$ and a Haar Wavelet basis $\mathbf{W}$; both of them can be implemented with an efficient algorithm.

For the analysis of our approach we assume two possible channel models. In both cases we use direct analog transmission of the measurement values, i.e. no quantization or other coding is applied. First, we depart from a channel with Additive White Gaussian Noise (AWGN) in which packages are erased at random with a loss rate p. After a full set of N measurements is created, these two distortions are supposed to interfere as follows: First, noise is added to the full measurements such that the Channel Signal to Noise Ratio (CSNR) satisfies

$$\mathrm{CSNR} = \frac{\|\boldsymbol{y}\|_2^2}{\|\boldsymbol{e}\|_2^2}$$

exactly in order to make results easier to compare. Finally a fraction $M = \lfloor (1-p)\, N \rfloor$ of the measurements are kept and handed over to the decoder.

Second, we also apply a noiseless block erasure channel, as it could arise from a best-effort network. For the noisy case described above this corresponds to the limit of an infinite CSNR.

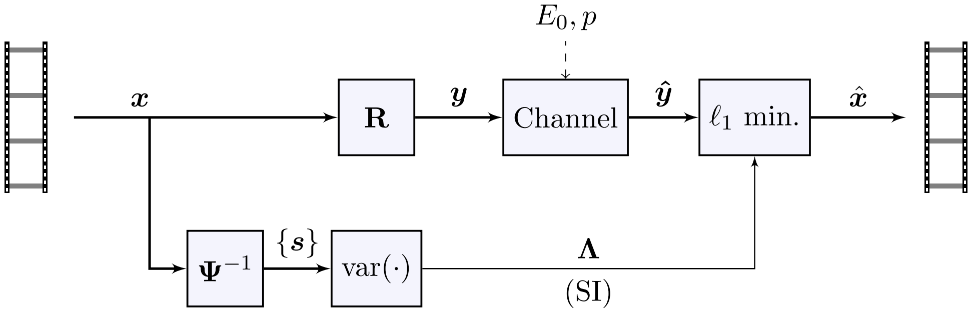


Figure 2.2: *Flow graph of a scheme with reweighting.*

Both channel models act directly on the baseband, no assumptions about modulation are made. This allows us to compare the trade-off between noisy and lossy channels as will be discussed in Sec. 2.3. Possible reasons causing package loss include collisions occurring for some receivers or congestion at the transmitter side.

2.2.2 Decoding

Before the measurements are decoded we need to compose the random matrix for each block according to the package loss pattern and then estimate the noise level.

If the encoder used the random matrix $\mathbf{R} \in \mathbb{R}^N$ to generate the measurements and among the N measurements of a given block the ones indexed by the set $S \subseteq \{1, \ldots, N\}$ with cardinality $|S| = M$ were received, we construct the measurement matrix

$$\boldsymbol{\Phi}^\mathsf{T} = \left\{ (\mathbf{R}^\mathsf{T})_i \,\middle|\, i \in S \right\}$$

from the rows of $\mathbf{R}$ indexed by S. The received measurements are deinterleaved and arranged accordingly into $\hat{\boldsymbol{y}}$. We assume that $\mathbf{R}$ is either known to all parties of the system or communicated by the transmitter by means of a simple random seed.

A noiseless channel would lead to a zero error vector, but this will not be the case for a practical channel. Hence we will use the ($\mathbf{P_{QC}}$) decoding algorithm together with the estimate for ϵ from (1.10). Then $\hat{\boldsymbol{s}}$ is calculated by solving ($\mathbf{P_{QC}}$) using the ℓ_1-magic [9] implementation for Matlab. Finally $\hat{\boldsymbol{s}}$ is projected back into the pixel domain and $\hat{\boldsymbol{x}}$ is displayed.

So far we have only assumed that an image can be sparsely approximated in some fixed bases. But we can also use previous knowledge about the coefficient distribution as well as the inter-frame correlation of videos to improve the performance of this basic scheme. These two modifications to the decoder will be presented in the following.

Reweighted decoding

In Sec. 1.4.5 on page 15 we have shown how previous knowledge about the distribution of the image coefficients in the transform domain can improve the performance of the decoder

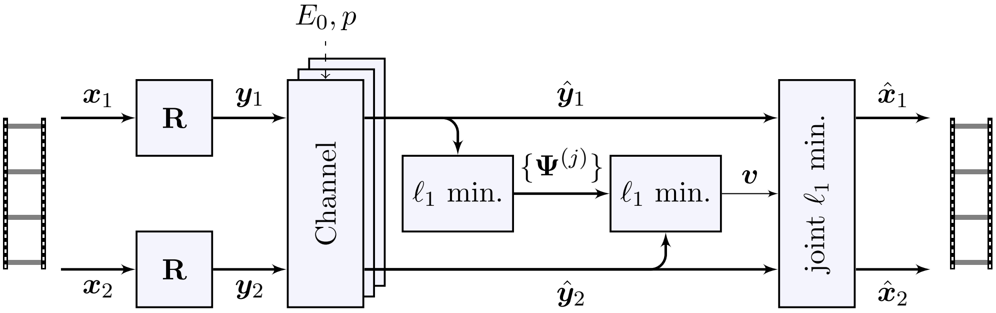


Figure 2.3: *Flow graph of a scheme with 2 frame inter decoding.*

by weighting the coefficients unevenly.

The coefficients of natural images in the DCT domain are unevenly distributed. This allows us to use the variance of each coefficient in a frame as an estimate for its magnitude and plug these values into the reweighted minimization as follows:

$$W_{ii} \propto \frac{1}{\sqrt{E(s_i^2)}} .$$

Unfortunately the sorted magnitudes of the coefficients of natural images in the transform domain do not decay sufficiently fast to use this approach iteratively. But their low frequency components are usually still dominant over the high frequency components. The estimated coefficient magnitudes are unevenly distributed and remain consistent within a given frame.

Figure 2.2 illustrates the modified scheme. In contrast to the basic scheme, the encoder now also needs to apply the transform $\boldsymbol{\Psi}$ to all blocks of the image which adds to its complexity and will also fix the choice of $\boldsymbol{\Psi}$ for all decoders that use this side-information. We notice, that the weights need to be transmitted from the encoder to all the receivers and hence we need a lossless side-channel for this scheme to work. But the rate required for this is small: If a frame is divided into B blocks then a fraction of $1/B$ of the total data will be required for this side information, remaining below 1% for practical block-sizes. The fact that these variances vary only slowly from one frame to the next can further reduce the rate. Furthermore the *SoftCast* scheme we compare to assumes the same information to be available at the decoder.

Inter-frame decoding

Without increasing the encoder complexity – or even changing it at all – we can use the correlation among frames to decode multiple frames together. This follows the paradigm of Distributed Video Coding (DVC) . The following related ideas have been studied previously.

Prades-Nebot et al. [36] propose a CS based scheme for video transmission. For each transmitted block three modes with different rates are possible. First, a block can be entirely skipped if it does not or only slightly varies from the previous frame. Second, a small set of CS measurements can be transmitted. This enables the receiver to do motion-estimation in the compressed domain and to insert the appropriate block from the previous frame. Finally, the receiver can request a full set of measurements via a feedback channel to recover a block directly if the other modes fail. This scheme requires every second frame to be intra coded to prevent error-propagation and an important part of its gain is due to frequent use of the first two modes.

Marcia et al. [30] study joint CS reconstruction of coded aperture images. Because there is no motion between the images, they can successfully apply a joint sparsity model.

A predictive coder can benefit from the fact that the residue after a reasonable prediction has lower energy than the frame itself and can code it in a more compact fashion. But the residue is a dense, unpredictable image and has much less structure than a natural image and no universally good basis can be given. For this reason it will be less sparse in a basis designed for natural images and we would actually need more coefficients to reconstruct the residue than we would require for the image itself. Thus it is not a favorable option to apply CS directly to the difference of two images or a residue and we will need to find a better way of exploiting the correlation between consecutive frames.

In the following we will use a joint sparsity model to exploit it. Although subsequent frames of a video are usually highly correlated, this correlation is subject to motion. Therefore we can expect a joint sparsity model to be successful only if this motion is reflected by some sort of geometric transform between the bases of each frame. The fact that the compressed sensing model allows a decoder to independently choose any matrix $\boldsymbol{\Psi}$ as long as $\boldsymbol{s}$ remains sparse makes this approach possible.

We will treat two frames $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ together, but this easily extends to three or more as well. We assume a joint sparsity model of a common sparse component plus innovations, commonly referred to as *JSM-1* following the framework presented in Sec. 1.4.5.

The main idea is to first estimate the local motion between two frames using an intra-decoded frame 1 as a reference for the next frame 2. For each block j of frame 2 we are about to reconstruct, we take all possible patches in frame 1 that are supported around block j as spanning vectors for a sparse representation. We call the matrix constructed from these patches $\boldsymbol{\Psi}^{(j)}$. It will not form a complete basis, but if the two frames are related by motion one of its elements will be highly correlated with the measurements of that block. The index of this best atom will give us an estimate for the motion vector $\boldsymbol{v}$ between the two frames around block j.

After the motion is estimated based on two independent reconstructions a joint basis can represent two concatenated blocks (one from each frame). This joint basis consists of a common part which supports both frames using a given basis for the first and a shifted version of it for the second frame. It is augmented with two separate bases for each frame.

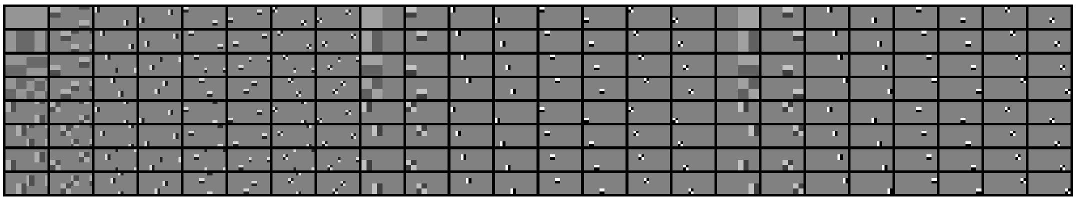


Figure 2.4: *An example of an overcomplete dictionary $\boldsymbol{\Psi}^{(j)}$ for inter frame decoding containing a common and two disjoint parts. A shift of $\boldsymbol{v}^{(j)} = (-5, 4)^\mathsf{T}$ is applied for the second frame.*

This makes the joint basis overcomplete but possibly leads to a higher sparsity of $\boldsymbol{s}$. Finally we get

$$\boldsymbol{\Psi}^{(j)} = \frac{1}{\sqrt{2}} \begin{pmatrix} \boldsymbol{\Psi} & \sqrt{2}\boldsymbol{\Psi} & \mathbf{0} \\ T_{\boldsymbol{v}^{(j)}}(\boldsymbol{\Psi}) & \mathbf{0} & \sqrt{2}\boldsymbol{\Psi} \end{pmatrix} \tag{2.1}$$

where the transform $T_{\boldsymbol{v}^{(j)}}(\cdot)$ denotes the shift by the motion vector $\boldsymbol{v}^{(j)}$. The two concatenated blocks can then be represented by a set of joint ($\boldsymbol{s}_{\text{joint}}$) and independent ($\boldsymbol{s}_1, \boldsymbol{s}_2$) coefficients as follows:

$$\hat{\boldsymbol{x}}_{\text{joint}} = \begin{pmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \end{pmatrix} = \boldsymbol{\Psi}\, \boldsymbol{s} = \boldsymbol{\Psi} \begin{pmatrix} \boldsymbol{s}_{\text{joint}} \\ \boldsymbol{s}_1 \\ \boldsymbol{s}_2 \end{pmatrix}.$$

Each of the steps is performed locally for each block and the block size will also determine the search space of the motion estimation.

We use a circular shift for the transform $T(\cdot)$ in the joint optimization. This provides the two advantages that we can guarantee a high level of incoherence without the need to remove duplicate atoms and that it can be implemented using a fast transform. Figure 2.4 illustrates a possible $\boldsymbol{\Psi}^{(j)}$.

2.3 Experimental Results

We have implemented our scheme in Matlab and evaluated it at various operating points defined by the channel loss rate p and the CSNR. This allows us to better understand how the trade-off between noisy and lossy channels acts on CS decoding. For comparison the Peak Signal to Noise Ratio (PSNR) was used, a standard measure for image applications. It is defined as

$$\text{PSNR} = 10 \log_{10} \frac{\max_i\{x_i\}}{\frac{1}{N} \left\|\hat{\boldsymbol{x}} - \boldsymbol{x}\right\|_2^2}.$$

The minimal PSNR for a uniformly gray image is approximately $10 - 15$ dB and visually good quality can be claimed around 25 dB and above.

All graphs in the following figures compare the image PSNR for different operating points in this parameter space.

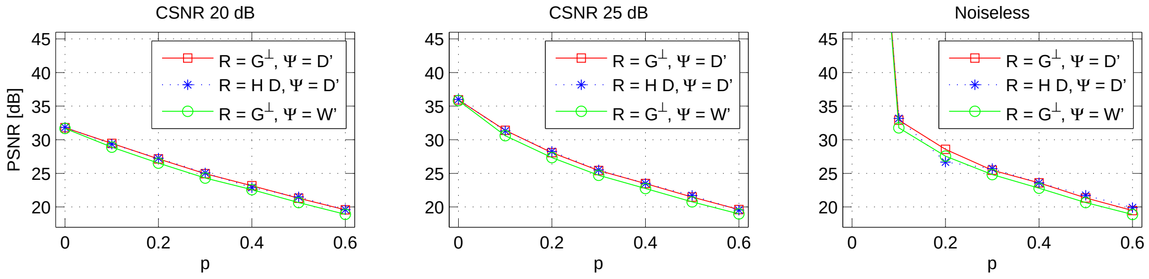


Figure 2.5: *Comparison of different matrices* $\mathbf{\Phi}$ *and* $\mathbf{\Psi}$ *as described before: (—□—)* $\mathbf{R} = \mathbf{G}^{\perp}, \mathbf{\Psi} = \mathbf{D}^{\mathsf{T}}$*, (··∗··)* $\mathbf{R} = \mathbf{H}\,\mathbf{D}, \mathbf{\Psi} = \mathbf{D}^{\mathsf{T}}$ *and (—○—)* $\mathbf{R} = \mathbf{G}^{\perp}, \mathbf{\Psi} = \mathbf{W}^{\mathsf{T}}$*. The graphs show results at different noise levels where each compares the image PSNR verus the loss ratio* p *for the sequence* football.

2.3.1 Basic scheme

For all the investigated combinations of $\mathbf{R}$ and $\mathbf{\Psi}$, experiments were run for a wide range of these two parameters in order to better understand the trade-off. They are visualized in Fig. 2.5.

The best performance is achieved when an orthogonal random matrix ($\mathbf{G}^{\perp}$) is used together with decoding into the DCT basis $\mathbf{D}$. It is slightly worse for just a random Gaussian matrix ($\mathbf{G}$) and for one with orthonormal columns ($\mathbf{G}^{\mathbf{1}}$); however, the differences are quite small and graphs are omitted. The same holds if the encoder additionally performs a decorrelating transform (i.e. $\mathbf{R} = \mathbf{G}\,\mathbf{D}$); this is to be expected from the CS theory because the combined matrix $\mathbf{\Phi}\,\mathbf{\Psi}$ remains the same. On the other hand when we use the deterministic measurement matrix $\mathbf{H}$, results drop by around 1 dB for almost all operating points.

Concerning the choice of the sparse representation we observe that a wavelet basis $\mathbf{W}$ performs worse than the DCT. We can also see that no cliff effect appears and we achieve a graceful degradation with respect to both dimensions of distortion. Hence one of the principal design goals is met. The decay with respect to the loss rate p is smooth, but faster than we anticipated. In most cases we could further improve the performance slightly by operating on bigger blocks, but considering the higher complexity this is a less attractive option.

2.3.2 Reweighting

Figure 2.6 shows the achieved improvements for various operating points. We see that at higher loss rates p, reweighting improves results by around 1–2 dB in PSNR while penalizing the results at full measurement rate slightly. But the choice of the decoding method is left to the decoder which could switch back to unweighted decoding at those operating points. (However, the decision to do so would need to be taken blindly.) The same holds in the rare cases where the reweighted optimization does not converge.

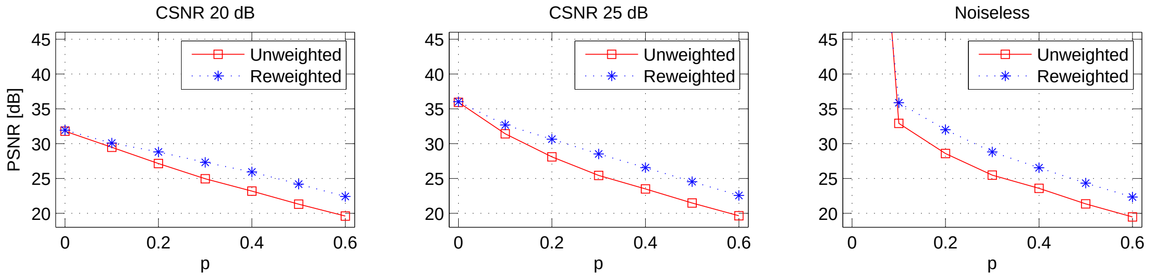


Figure 2.6: *Comparison of (—□—) default ℓ_1 minimization with (···∗···) reweighting. The graphs show results at different noise levels and each compares the image PSNR vs. the loss ratio p for the sequence* football.

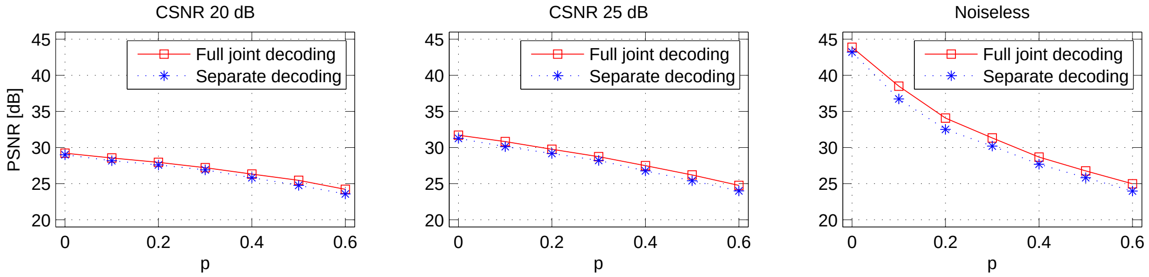


Figure 2.7: *Comparison of (···∗···) default ℓ_1 minimization with (—□—) inter-frame decoding for two consecutive frames of the sequence* football*. The graphs show results at different noise levels and each compares the image PSNR vs. the loss ratio p.*

2.3.3 Inter-frame decoding

Our implementation uses blocks of size 8×8 for both motion estimation and reconstruction. Figure 2.8 illustrates the reconstruction of one block in the joint reconstruction step. There we see the distribution of the coefficients among the joint and the two distinct bases. If we consider that videos have usually a high correlation between frames, then – in the spirit of the Slepian-Wolf theorem [44] – it should be possible to get a considerable performance gain from inter-frame decoding alone. Nevertheless our results (shown in Fig. 2.7) are less favorable. Overall we achieve only little gain for low CSNR and improve by up to 1 dB for better channels even in the presence of losses. Although experiments show that an overcomplete basis as given by (2.1) leads to slightly better results than a complete one, it also increases the complexity. We conclude that a more sophisticated joint sparsity model or a different approach should be sought after in order to achieve a more significant performance gain.

2.3.4 Comparison with "SoftCast"

Finally we compare our best configuration (Reweighted decoding applied to $\mathbf{R} = \mathbf{G}^{\perp}, \boldsymbol{\Psi} = \mathbf{D}$) with *SoftCast*; the R-D curves are shown in Fig. 2.9 and the corresponding visual comparison in Fig. 2.10 for a CSNR of 25 dB.

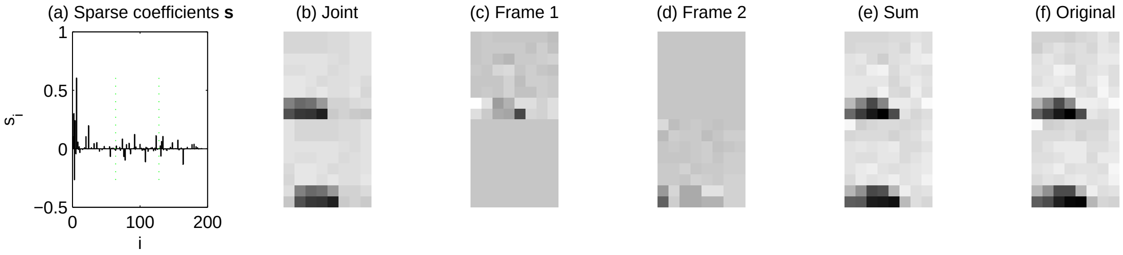


Figure 2.8: *An arbitrary block being reconstructed into the overcomplete dictionary* $\mathbf{\Psi}^{(j)}$. *Part (a) shows the sparse coefficient vector split into the three parts of the joint basis and the two separate bases. The other subfigures show the same block for both frames on top of each other as follows: (b) – (d) the joint and distinct contributions respectively, (e) the sum of them and (f) the original.*

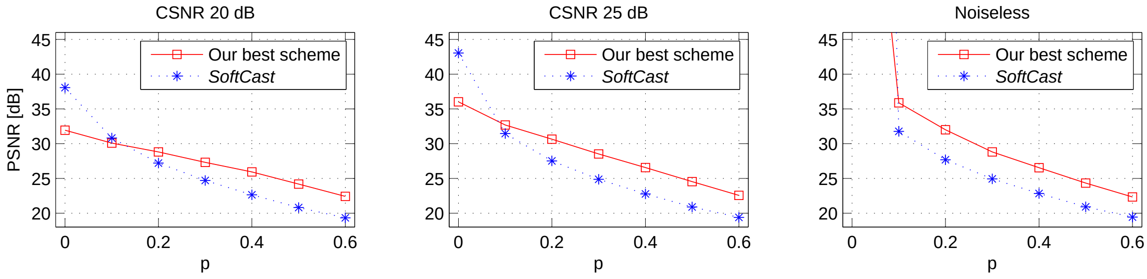


Figure 2.9: *Comparison of (—□—) our scheme with (···✳···)* SoftCast. *The four graphs show results at different noise levels and each compares the image PSNR vs. the loss ratio p for the sequence* football.

Our scheme is competitive at high loss rates p and low channel noise. However, the *SoftCast* scheme is clearly superior in the contrary cases of low losses and for high noise. The fact that our scheme lies behind *SoftCast* at low losses is less surprising considering that we perform only direct random projections at the encoder where *SoftCast* is based on optimal scaling as the essential encoding step. On the other hand we achieve a performance gain in the order of 2 dB over their scheme starting from only 10–20% loss.

2.4 Discussion

We have proposed a scheme for wireless video multicast based on compressed sensing which does not suffer from a cliff-effect. We compared our results with a recent scheme designed for the same purpose and find competitive results in the case of high losses and low noise. Furthermore extensive experiments were deployed to compare various measurement matrices.

The advantage of our scheme is a non-adaptive, simple encoder. It automatically scales

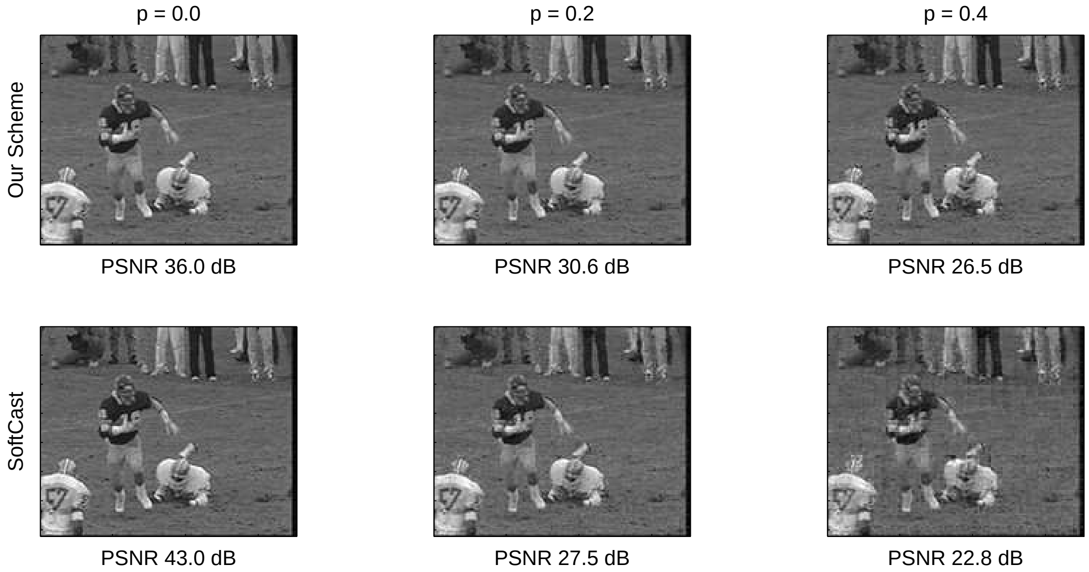


Figure 2.10: *Visual results of our scheme for frame 70 of the* Football *sequence at a CSNR of* 25 *dB. Compare these results with Fig. 2.9.*

with the number of receivers and does not require solving any resource allocation problem. Additionally this introduces the interesting property that opposed to the traditional approach where the decoder is fully specified and encoding is left for various implementations, here a decoder has the free choice of its method, in practice the choice of the sparse basis and a correlation model. This adds a certain universality and makes the setup future proof. The main drawback is a high complexity at the receiver side, that is required to solve the optimization problem. This makes the implementation of a real-time application for higher resolutions challenging. However, greedy pursuits could be employed instead of BP at the expense of a higher measurement rate.

As previously investigated by Goyal [19] and others, CS can not be expected to be optimal for compression in the information theoretic sense when applied to data that is already fully sampled. Nevertheless we conjectured a higher overall gain to be achievable in such a JSCC scenario than what we found through experiments.

By looking at the results for inter-frame decoding, we have to conclude that the proposed scheme built on the joint sparsity model is most likely not optimal. Future work could try to better exploit this correlation to achieve a higher reconstruction performance. Either by improving the decoding – model-based compressive sensing [3] could be considered for instance – or by changing the encoder to apply the random coding to a group of frames at once.

Chapter 3

Joint Decoding of Stereo JPEG Image Pairs

In this part we propose and study a second application of sparsity based image reconstruction methods. Because we would like to improve on the rather small gain from inter-frame decoding found in the previous chapter, we study a case where a similar but simpler correlation is present. Furthermore the effects introduced by quantization was left out in the context of compressed sensing before, although it could be a necessary step in practice. For these reasons we investigate the joint decoding of quantized stereo image pairs and focus on how the correlation between them can be exploited with sparse image representations.

3.1 Motivation

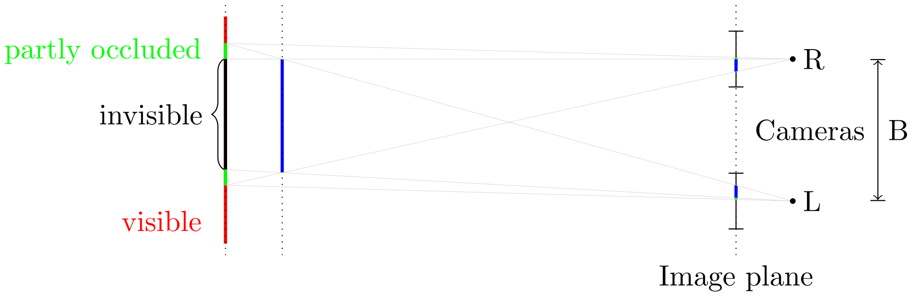
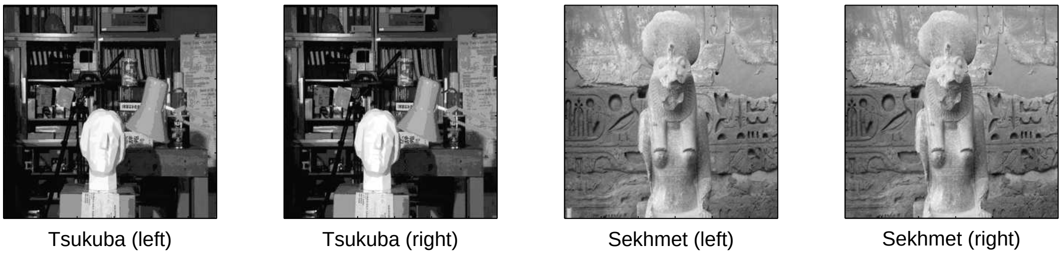


Figure 3.1: *Illustration of the stereo geometry shown for a planar cut through the scene with a background and a foreground object and parallel cameras with a baseline B.*

Stereoscopic imaging existed for a long time and might gain popularity again in the digital world with stereo enabled screens and graphic cards becoming more widely available on the consumer market, due to better hardware such as auto-stereoscopic displays, shutter or wavelength dependent glasses which offer a sufficient viewing comfort level. Furthermore, emerging high definition content will even more depend on efficient compression methods than it is already the case today.

Stereo perception is usually achieved by simultaneously presenting two images of the same scene taken from two slightly displaced positions to both eyes. We will denote these two



Tsukuba (left) Tsukuba (right) Sekhmet (left) Sekhmet (right)

Figure 3.2: *The two stereo image pairs* Tzukuba *(from the Middlebury stereo vision dataset [39]) and* Sekhmet *(taken with a commercial stereo camera) that were used for the following experiments.*

images with left (L) and right (R). These images are taken with two parallel cameras at a distance B (the baseline) which introduces a an apparent displacement of the objects called parallax. The depth information directly translates into the disparity that relates the two images and in the case of perfectly rectified views the possible transformations reduce to horizontal shifts only. This is illustrated by Fig. 3.1 on the preceding page for an idealized situation. Because both images represent the same reality they are highly redundant (as illustrated by the examples in Fig. 3.2) and accordingly we can expect significant compression ratios to be achievable. We can see how occlusions are introduced around object boundaries. This leads to information that is only present in one view while most of the scene is equally captured by both cameras.

We can draw a parallel to motion estimation and compensation in video compression with the simplification that no physical motion is present in the scene. Because only the cameras are displaced all movements can be explained by rigid body motion. Therefore a predictive coding scheme could be employed by intra coding the first image, followed by the disparity field for a predictor and the residue. Such a traditional coding has previously been proposed by Perkins [34] for example. But the proprietary transmission format required by such an approach is unlikely to replace established general purpose image formats and it is worth investigating how much we could still improve without departing from JPEG encoding. Indeed, today's digital cameras are widely equipped with a JPEG encoder and stereoscopic image pairs are encoded as separate JPEG images by most applications, be it JPEG Stereo (JPS), Multi-Picture Object (MPO) [1] or others; thus they directly double the bandwidth requirements. Cameras also have a limited amount of processing power which further motivates a distributed coding scheme.

We propose a joint decoding strategy in order to enhance the quality of the reconstructed image pairs. Overall, the joint decoder allows for a lower overall bitrate while maintaining similar quality for both views. We cast the reconstruction problem as a regularized convex optimization problem that is constrained by consistent reconstruction conditions. We show that proper regularization permits to increase the accuracy of the disparity estimation, hence to obtain better reconstruction quality.

We consider an asymmetric coding scheme where one of the images is encoded at high quality, the other one at a reduced quality. Studies by Seuntiens et al. [42] and others indicate that the human brain can tolerate a fair amount of asymmetric image quality for stereo viewing such that the perceived quality lies between that of the two views. Such an asymmetric scheme could also be of interest to applications where the second view is not always required, notably if no stereo display is available.

This work is related to the distributed coding framework, where joint decoding is used to reconstruct correlated signals that have been independently encoded. However, we do not work here on the coding strategy, but rather rely on a classical encoding solution. The joint reconstruction of compressed images has been considered also in the compressed sensing community, where different approaches have been proposed to represent images or parts of them as a sparse linear combination of other images assembled in a dictionary. If such a representation exists, it can under some conditions also be recovered in a stable way from linear projections onto a set of random vectors that reduces its dimensionality.

This has led to applications in video coding where the dictionary is composed of blocks from a previous frame [36], face recognition where the candidate faces build the dictionary [51], or multi-view representations [48]. In our solution a reconstructed block will be based on local dictionaries of candidate blocks. Alternatively, super-resolution reconstruction from image sequences has a similar objective of quality enhancement with multiple compressed images. It tries to either estimate a dense displacement field or to fuse different frames of a video together to enhance its quality. This is often formulated as an inverse optimization problem with a smoothness constraint on the displacement field (e.g. [23] and [50]), but unlike here more than two images are usually involved.

3.2 Proposed Scheme

3.2.1 Encoding

JPEG is a block-based still image compression scheme that compacts the image energy in a small number of coefficients and introduces losses mostly at high frequencies where they are visually more acceptable [33]. This is done by applying the two-dimensional DCT denoted by $\mathbf{D}$, followed by scalar quantization with up to ten times bigger step sizes at the highest frequencies than for the lower ones. The quantization step sizes are given by a table $\boldsymbol{q}$. A typical quantization table is given in matrix form by Eq. (3.1) below. If $\boldsymbol{b}$ is an image block and $\boldsymbol{y} = \mathbf{D}\boldsymbol{b}$ its representation in the transform domain, then quantization can be written as

$$\overline{y}_i = q_i \left[\frac{y_i}{q_i} \right]$$

where $[\cdot]$ denotes rounding to the nearest integer. Because of the lossy nature of JPEG encoding we have a certain freedom to fill in the coarsely quantized coefficients from the other image at higher quality.

$$\boldsymbol{q}_{50} = \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix} \tag{3.1}$$

In the following we will always use the left image as the intra coded reference and the right one as the compressed image that we want to enhance. We limit our studies to luminance images only, but this extends to the other color components as well, because they are coded in the same fashion.

3.2.2 Decoding

The compressed version of the right image defines a set of possible solutions for approximating the original image version, which are all consistent with respect to the compressed one, meaning that they would yield exactly the same JPEG bitstream after a recompression using the same quantization matrix. This is the range we will operate in to reconstruct the image. Although a midpoint or a centroid dequantization followed by an inverse DCT will likely minimize the reconstruction error if no further information is present, they are not the only choices within the aforementioned admissible region. Thus we will use the compressed image only to formulate a constraint on the output image. Our scheme operates on the blocks of 8×8 pixels defined by JPEG. We can formulate this elementwise constraint in the transform domain as

$$\left|\mathbf{D}\left(\hat{\boldsymbol{b}}^{(i)} - \overline{\boldsymbol{b}}^{(i)}\right)\right| \preceq \frac{1}{2}\boldsymbol{q}\,, \tag{3.2}$$

where $\overline{\boldsymbol{b}}^{(i)}$ is the i^{th} block in the pixel domain after JPEG compression, $\hat{\boldsymbol{b}}^{(i)}$ is the estimate of that block after enhancement and the element-wise inequality $|x_j| \leq y_j \; \forall j$ is written as $|\boldsymbol{x}| \preceq \boldsymbol{y}$.

Now we build a dictionary $\boldsymbol{\Psi}^{(i)}$ composed of possible candidate blocks $\boldsymbol{\psi}_j^{(i)}$ from the reference image. Figure 3.3 illustrates the origin of the dictionary for block i. They are gathered from a range from 0 disparity up to the maximum disparity D_h in horizontal direction around the location in question. We further extend this range to $2D_v + 1$ shifts in the vertical direction to accommodate slight misalignments of the two cameras. Each of the totally $D = D_h(2D_v + 1)$ dictionary elements $\boldsymbol{\psi}_j^{(i)}$ now has an associated disparity vector $\boldsymbol{d}_j \in \{0, \ldots, D_h\} \times \{-D_v, \ldots, D_v\}$ that will be used later on to build the global disparity field.

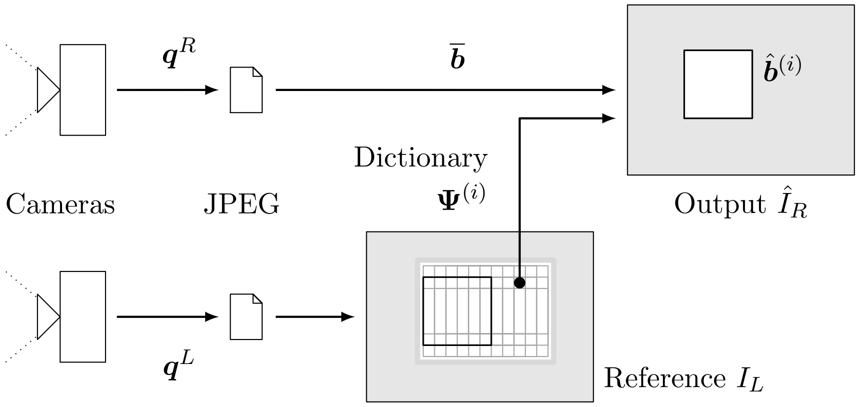


Figure 3.3: *Schematic view of the proposed scheme. Illustrating the separate encoding, the composition of the dictionary for a given block i from the reference view and the reconstruction of that block.*

We now want to represent a block $\boldsymbol{b}^{(i)}$ as a linear combination of dictionary elements with the coefficient vector $\boldsymbol{s}^{(i)} \in \mathbb{R}^D$ as

$$\boldsymbol{b}^{(i)} \approx \sum_j \boldsymbol{\psi}_j^{(i)} s_j = \boldsymbol{\Psi}^{(i)} \boldsymbol{s}^{(i)}.$$

In general it is not possible to find such a decomposition that also satisfies (3.2). Thus we introduce the slack variables $\hat{\boldsymbol{b}}^{(i)}$ that are constrained as above and approximated by a linear combination over $\{\boldsymbol{\psi}_j^{(i)}\}$.

3.2.3 Regularization

However, this inverse problem is still ill posed and we can regularize it in two ways. First, only a small number of the dictionary elements will contribute – ideally only a single one – and we can thus require $\boldsymbol{s}$ to be sparse. Although a high sparsity is best described with a low ℓ_0 pseudo-norm $\|\boldsymbol{s}\|_0 = |\{s_j | s_j \neq 0\}|$ this is not a convex function and would lead to a problem of combinatorial nature. Consequently we approximate it by the convex ℓ_1 norm $\|\boldsymbol{s}\|_1 = \sum_j |s_j|$.

Second, we can assume the disparity field $\mathbf{V}$ to be piecewise smooth because disparity discontinuities will occur only at object boundaries. Furthermore, not every block contributes an equal amount of depth information. For this reason we can improve the reconstruction by enforcing a low total variation. We first calculate the disparity $\boldsymbol{v}^{(i)}$ for each block as the weighted sum of the contributing disparities

$$\boldsymbol{v}^{(i)} = \sum_j \boldsymbol{d}_j \, s_j^{(i)}$$

and use the total variation of a scalar field as defined by Eq. (1.4). Because $\boldsymbol{v} = [v_x \, v_y]^\mathsf{T}$ has two components – one for the vertical and one for the horizontal displacement – there are also two total variations that we add together for the minimization.

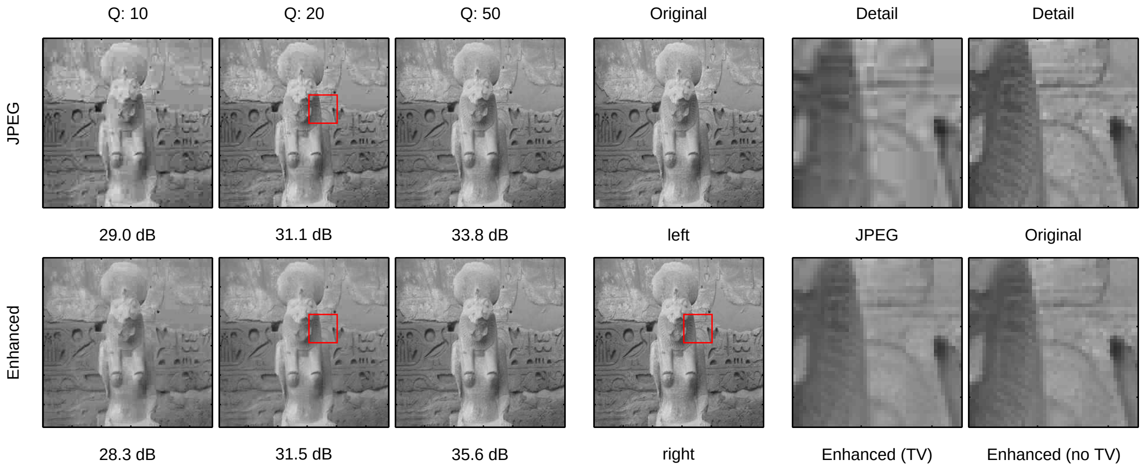


Figure 3.4: *This experiment on a* 288×288 *pixel image of* Sekhmet *taken with a commercial stereo camera illustrates our method. Results for three different JPEG quality settings with and without enhancement are followed by the original left and right view and enlarged details from the second image pair. We can improve the visual quality even if PSNR results drop and find less blocking artifacts if the total variation of the disparity field is minimized. Compare these results with Fig. 3.6.*

This last step makes our problem a global one involving all blocks at once. In the following, all N blocks of an image are concatenated such that $\boldsymbol{b}^\mathsf{T} = [\boldsymbol{b}^{(1)\,\mathsf{T}} \cdots \boldsymbol{b}^{(N)\,\mathsf{T}}]$ and the transform $\mathbf{D}$ becomes a block diagonal matrix.

Putting it all together we obtain the objective function

$$\begin{aligned} \hat{\boldsymbol{b}} &= \arg\min_{\hat{\boldsymbol{b}}} \left\|\hat{\boldsymbol{b}} - \boldsymbol{\Psi}\boldsymbol{s}\right\|_2^2 + \lambda_s \left\|\boldsymbol{s}\right\|_1 + \lambda_v \left(\left\|\boldsymbol{v}_x\right\|_{\mathrm{TV}} + \left\|\boldsymbol{v}_y\right\|_{\mathrm{TV}}\right) \\ &\text{subject to } \left|\mathbf{D}\left(\hat{\boldsymbol{b}} - \overline{\boldsymbol{b}}\right)\right| \preceq \tfrac{1}{2}\boldsymbol{q}. \end{aligned} \tag{3.3}$$

The parameter λ_v is the Lagrange multiplier weighting the importance of a smooth disparity field. The second parameter λ_s acts on the sparsity of the signal and trades off fidelity versus sparsity. All three terms in (3.3) scale with the number of blocks N, hence we can set the relative values of λ_s and λ_v independently of the image size.

The choice of λ_s is crucial, because on one hand a big value makes $\boldsymbol{s}$ approach $\mathbf{0}$, while on the other hand a small value can lead to a non-sparse $\boldsymbol{s}$ and a blurred result. We choose it such that the terms of the objective function are of similar magnitude and verify empirically that $\lambda_s = 5 \times 10^{-2}$ leads to good results for the tested images and a dictionary of size $D = 13 \times 9 = 117$. The choice of λ_v is less critical and we set it to $\lambda_v = 1 \times 10^{-2}$. All results presented in the following were obtained with this same set of parameters.

In order to solve (3.3) we use the `cvx` package for Matlab [7, 20]. Larger images are further partitioned into independently reconstructed regions of smaller size to keep the memory requirements manageable. Solving this problem is quite involved, but it scales linearly with

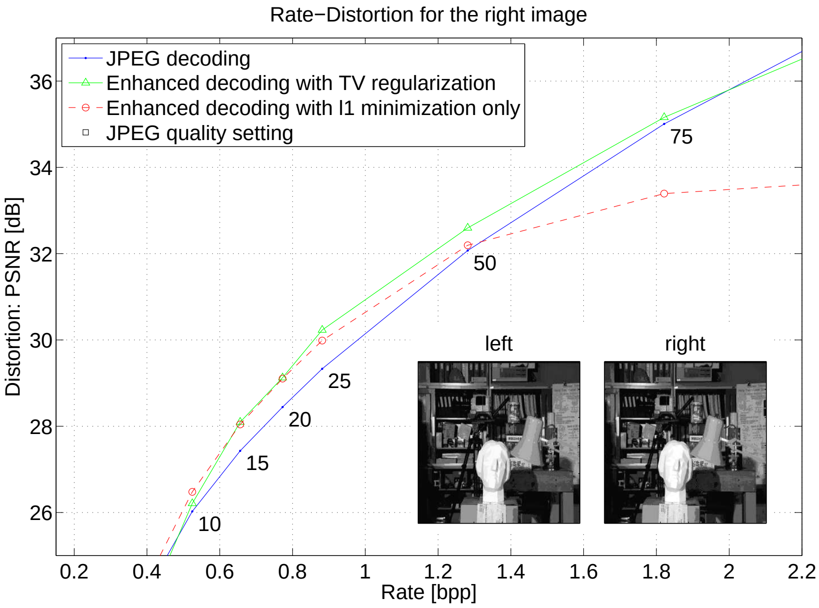


Figure 3.5: *Rate-distortion comparison of JPEG (—●—) and joint decoding (—△—) for the* Tsukuba *stereo test set* (192 × 192 *pixels). Also shown is the curve for unconstrained* ℓ_1 *minimization (- ○ -) given by Eq. (3.4).*

the image size. An accurate knowledge of the maximum disparity reduces the number of variables and thus the runtime of the optimization.

3.3 Experimental Results

In this section we analyze the performance of the joint decoding algorithm. As we can see from Fig. 3.4 we are able to enhance an image with correctly positioned details even if the right image is highly compressed. The ubiquitous blocking artifacts introduced by JPEG disappear and texture is added. Despite the small improvement in peak signal-to-noise ratio (PSNR) at low bitrates, the visual quality of the images improves clearly and in a consistent way. The enlarged areas show that even small details can be recovered that would simply be blurred out by JPEG. For medium bitrates (JPEG quality around 50) the PSNR improves by about 1 dB on average and more as the rate-distortion comparisons in Fig. 3.5 and Fig. 3.6 highlight. In the optimal region we can accommodate a bitrate saving of 20% and above for the right view at a similar decoding quality.

Regions that are occluded in the reference view cannot possibly be reconstructed by this method. In the middle range (Quality 50 − 75) only few blocks have a PSNR decrease and they lie usually in such regions around disparity discontinuities as well as the very right border of the image. Nevertheless, ghosting artifacts appear seldom.

Even though the results shown here were obtained with well aligned image pairs, additional

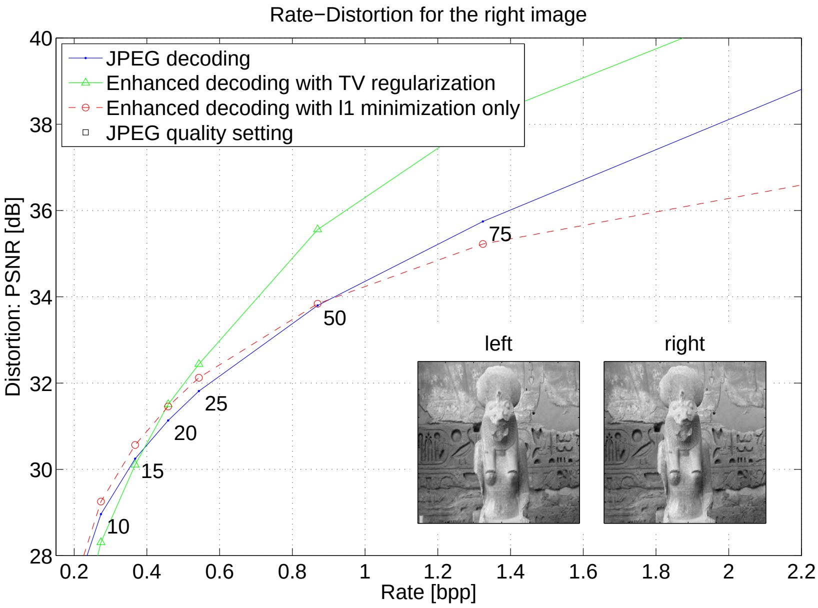


Figure 3.6: *Rate-distortion comparison of JPEG (—•—) and joint decoding (—△—) for the right view of the* Sekhmet *image from Fig. 3.4 (288 × 288 pixels). Also shown is the curve for unconstrained ℓ_1 minimization (- ○ -) given by Eq. (3.4).*

experiments (Fig. 3.7) after multiple pixel shifts and a 2° rotation still exhibit good performance.

Furthermore, we study the influence of the individual parts of the optimization. First, we can set $\lambda_v = 0$ to remove the regularization of the disparity field. We find that at low bitrates the compressed image might not contain enough details to reliably find a corresponding block in the reference view and it is in this region where the additional regularization of the disparity field leads to a further improvement. Although the increase in PSNR is only little, less artifacts are visible. The last part of Fig. 3.4 shows such a case. Second, we can also compare our method with the unconstrained ℓ_1 minimization

$$\hat{\boldsymbol{b}} = \boldsymbol{\Psi}\hat{\boldsymbol{s}}, \quad \hat{\boldsymbol{s}} = \arg\min_{\boldsymbol{s}} \ \left\|\bar{\boldsymbol{b}} - \boldsymbol{\Psi}\boldsymbol{s}\right\|_2^2 + \lambda_s \left\|\boldsymbol{s}\right\|_1 . \tag{3.4}$$

This tends to saturate at some PSNR level, but also gives an improvement at low rates as seen from Fig. 3.5 and Fig. 3.6.

Finally, we should note that in these experiments the reference image was always given at full quality. If the reference itself is compressed the improvements will naturally decrease; however, reference images at a quality setting of 80 and 90 could still be used successfully, as it has been confirmed by additional experiments.

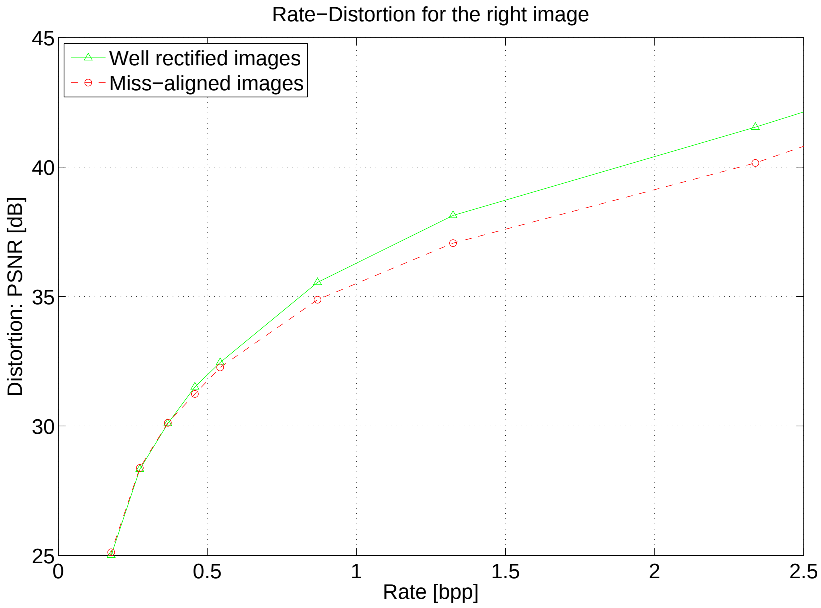


Figure 3.7: *Rate-distortion comparison of a well rectified image (—△—) and a non rectified one (- ○ -) after a multiple pixel shift and with a* 2° *rotation of the right view for* Sekhmet*.*

3.4 Discussion

We have presented a joint decoding solution for stereo image pairs. This method permits to reduce the bitrate of one view of a stereo image pair that is based on two separately coded, standard compliant JPEG images, but produces visually much better results than separate decoding. The only assumptions we make about the image pair is a relatively good alignment and a known maximum disparity. However both are only required to reduce the runtime of the algorithm. Because the images from stereo cameras have a relatively short baseline (based on the average eye distance of about 6 cm which is considerably shorter then the usual distance to the closest object) the search-space can be kept quite small. Due to the multiple objectives in the minimization, a careful tuning of the parameters is crucial.

The optimization is still fairly slow and a tailor made minimization algorithm instead of the general purpose solver could clearly bring an improvement. We also note that because the DCT is a unitary transform, the whole minimization can be directly formulated in transform domain only. This makes it unnecessary to repeatedly apply the transform during the minimization and increases the speed by a factor of 2 and more. It would certainly be desirable to dispose of a symmetric scheme where no high resolution reference image would be required but two compressed view could be jointly decoded while remaining consistent with the quantization.

One could also ask why we did not enforce a TV constraint on the image itself. Although no such experiments were carried out, the complexity would increase by quite a bit because the image is 64 times denser than the motion field and a low TV could again remove the texture that we wanted to add back to the compressed image.

We showed that it is possible to bring the quality of a second view closer to that of the reference image and simultaneously mitigate the effect of JPEG compression artifacts. We studied only natural images of Lambertian scenes; it is to be expected that image pairs with specularities, reflections, transparent objects or other non-Lambertian features would benefit less from the enhancement.

The presented scheme provides good results for a viewing application because the two images will be of comparable quality. On the other hand it might not be a good preprocessing step for vision applications, although the coarse depth maps obtained as a side product indicate that a fair amount of disparity estimation can still be done after compression.

Chapter 4

Conclusions

We have studied two applications that greatly profit from the fact that sparse representations of images exist and can be found efficiently. Such representations offer an elegant way to formulate a problem and the available tools to solve them are versatile enough to attack a wide range of problems in a similar fashion.

The introduced video multicast scheme based on compressed sensing has the nice property to enable a future-proof and universal coding scheme. The encoder has little to worry about the structure of the signal or the properties of the channel. At the same time the decoder can use all its knowledge and computational power to reconstruct the signal. Such a scheme exploits the lossy compressibility of the signal, and at the same time makes it possible to recover in a stable way from errors, thus forming a practical joint source-channel coding.

A question raised in the first part of this work is how further correlation models could be integrated with a compressed sensing decoder to better exploit inter frame correlation. The approach inspired by joint sparsity model yielded rather small improvements and exploring better methods of coupling CS decoding with a correlation model could pay off. Future work could address this, as well as other channel and distortion models. It would also be of interest to study how unevenly introduced errors affect a CS based image communication scheme and how they can be mitigated. They could for instance be introduced if a signal is transmitted over two or more channels with different noise levels or by quantization.

This questions have also led us to study a simpler case of recovery from quantization for a pair of correlated images. We have studied a way to reconstruct an image from coarse and uneven quantization by representing it in a dictionary of image patches and with proper regularization such that it remains consistent with the quantized version. This work could possibly be extended to decode both images in this fashion, ideally in a symmetric way, or by employing dictionary learning.

Overall, sparse signal approximations in overcomplete dictionaries make a wide range of applications possible and the future will show us what more can be achieved with them.

List of Figures

Bibliography

[1] Multi-Picture Format. *Camera & Imaging Products Association Standardization Committee*, 2009.

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

[3] R.G. Baraniuk, V. Cevher, M.F. Duarte, and C. Hegde. Model-Based Compressive Sensing. page 20, August 2008.

[4] D. Baron, R.G. Baraniuk, M.B. Wakin, M.F. Duarte, and S. Sarvotham. Distributed compressed sensing. *preprint*, pages 1–50, 2005.

[5] A. Beck and M. Teboulle. A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems. *SIAM Journal on Imaging Sciences*, 2(1):183, 2009.

[6] J.M. Bioucas-Dias and M.A.T. Figueiredo. Two-step algorithms for linear inverse problems with non-quadratic regularization. In *IEEE International Conference on Image Processing ICIP*, pages 5–8, 2007.

[7] S.P. Boyd and M Grant. CVX: Matlab software for disciplined convex programming. *(web page and software)*, 2009.

[8] E.J. Candes, S. Becker, and J. Bobin. NESTA: A Fast and Accurate First-order Method for Sparse Recovery. *Arxiv preprint arXiv:0904.3367*, 91125:1–37, 2009.

[9] E.J. Candes and J. Romberg. l1-magic : Recovery of Sparse Signals via Convex Programming. *URL: www.acm.caltech.edu/l1magic/downloads/l1magic.pdf*, 2005.

[10] E.J. Candes, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory*, 52(2):489–509, 2006.

[11] E.J. Candes, M.B. Wakin, and S.P. Boyd. Enhancing Sparsity by Reweighted l1 Minimization. *Journal of Fourier Analysis and Applications*, 14(5-6):877–905, October 2008.

[12] I. Carron. Nuit Blanche, 2010.

[13] D.L. Donoho and X. Huo. Uncertainty principles and ideal atomic decomposition. *IEEE Transactions on Information Theory*, 47(7):2845–2862, 2001.

[14] M.F. Duarte, M.A. Davenport, D. Takhar, J.N. Laska, K.F. Kelly, and R.G. Baraniuk. Single-Pixel Imaging via Compressive Sampling. *IEEE Signal Processing Magazine*, 25(2):83–91, March 2008.

[15] M.F. Duarte, S. Sarvotham, D. Baron, M.B. Wakin, and R.G. Baraniuk. Distributed Compressed Sensing of Jointly Sparse Signals. *Conference Record of the Thirty-Ninth Asilomar Conference onSignals, Systems and Computers, 2005.*, pages 1537–1541, 2005.

[16] M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society*, 15(12):3736–45, December 2006.

[17] M.A.T. Figueiredo, R. D. Nowak, and S.J. Wright. Gradient Projection for Sparse Reconstruction: Application to Compressed Sensing and Other Inverse Problems. *IEEE Journal of Selected Topics in Signal Processing*, 1(4):586–597, December 2007.

[18] P. Frankl and H. Maehara. The Johnson-Lindenstrauss lemma and the sphericity of some graphs. *Journal of Combinatorial Theory, Series B*, 44(3):355–362, 1988.

[19] V.K. Goyal, A.K. Fletcher, and S. Rangan. Compressive Sampling and Lossy Compression. *IEEE Signal Processing Magazine*, 25(2):48–56, March 2008.

[20] M. Grant and S.P. Boyd. *Graph implementations for nonsmooth convex programs, Recent Advances in Learning and Control (a tribute to M. Vidyasagar)*, pages 95–110. Springer, 2008.

[21] A. Gupta and S. Dasgupta. An elementary proof of the Johnson-Lindenstrauss Lemma. *Technical Report 99-006, UC Berkeley*, 1999.

[22] J. Hadamard. Sur les problèmes aux dérivées partielles et leur signification physique. *Princeton University Bulletin*, pages 49–52, 1902.

[23] B.K.P. Horn and B.G. Schunck. Determining Optical Flow. *Artificial Intelligence*, 17, 1980.

[24] L. Jacques and P. Vandergheynst. *Compressed Sensing: "When sparsity meets sampling"*, pages 1–30. Wiley-Blackwell, 2010.

[25] S. Jakubczak, H. Rahul, and D. Katabi. One-Size-Fits-All Wireless Video. *HotNets*, 2009.

[26] W.B. Johnson and J. Lindenstrauss. Extensions of Lipschitz mappings into a Hilbert space. In *Conference in modern analysis and probability*, pages 189–206, New Haven, CI, 1984. AMS.

[27] C. Luo, F. Wu, J. Sun, and C.W. Chen. Compressive Data Gathering for Large-Scale Wireless Sensor Networks. In *MobiCom*, pages 145–156. ACM, 2009.

[28] M. Lustig, D. L. Donoho, J. M. Santos, and J. M. Pauly. Compressed sensing MRI. *IEEE Signal Process. Mag.*, 25:72–82, 2007.

[29] S. Mallat and G. Peyre. *A Wavelet Tour of Signal Processing: The Sparse Way.* Academic Press, Burlington, 3 edition, 2009.

[30] R. Marcia and R.M. Willett. Compressive coded aperture video reconstruction. *Proc. European Signal Processing Conf.(EUSIPCO)*, (1), 2008.

[31] G. Monaci and P. Vandergheynst. Learning structured dictionaries for image representation. *2004 International Conference on Image Processing, 2004. ICIP '04.*, pages 2351–2354, 2004.

[32] B. A. Olshausen and D. J. Field. Sparse coding with an overcomplete basis set: a strategy employed by V1? *Vision research*, 37(23):3311–25, December 1997.

[33] W. Pennebaker and J. Mitchell. *JPEG still image data compression standard.* New York, 1993.

[34] M.G. Perkins. Data Compression of Stereopairs. *IEEE Transactions on Communications*, 40(4):684–696, 1992.

[35] D.P. Petersen and K-H. Lee. Optimal Linear Coding for Vector Channels. *IEEE Transactions on Communications*, 24(12):1283–1290, December 1976.

[36] J. Prades-Nebot, Y. Ma, and T. Huang. Distributed video coding using compressive sampling. *Proceedings of the Picture Coding Symposium*, 2009.

[37] R. Baraniuk et al. Compressive Sensing Resources.

[38] A. Said and W.A. Pearlman. A new, fast, and efficient image codec based on set partitioning in hierarchical trees. *IEEE Transactions on Circuits and Systems for Video Technology*, 6(3):243–250, 1996.

[39] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47(1/2/3):7–42, 2002.

[40] M. B. Schenkel, C. Luo, F. Wu, and P. Frossard. Compressed Sensing Based Video Multicast. *Proc. VCIP*, (accepted for publication), 2010.

[41] M. B. Schenkel, C. Luo, F. Wu, and P. Frossard. Joint Decoding of Stereo JPEG Image Pairs. *Proc. ICIP*, (submitted for publication), 2010.

[42] P. Seuntiens, L. Meesters, and W. Ijsselsteijn. Perceived quality of compressed stereoscopic images: effects of symmetric and asymmetric JPEG coding and camera separation. *ACM Transactions on Applied Perception*, V, 2006.

[43] C.E. Shannon. A Mathematical Theory of Communication. *Bell System Technical Journal*, 27:379–423, 623–656, 1948.

[44] D. Slepian and J.K. Wolf. Noiseless coding of correlated information sources. *IEEE Transactions on information Theory*, 19:471–480, 1973.

[45] L. Söderberg. Lena. *Playboy*, 20(11):135–141, 1972.

[46] J.Z. Sun and V.K. Goyal. Quantization for Compressed Sensing Reconstruction. In *Proc. SAMPTA*, Marseille, 2009.

[47] A. Tarantola. *Inverse problem theory.* SIAM, Philadelphia, 2005.

[48] V. Thirumalai and P. Frossard. Motion estimation from compressed linear measurements. *Proceedings of ICASSP*, 2010.

[49] M.E. Tipping. Sparse bayesian learning and the relevance vector machine. *The Journal of Machine Learning Research*, 2001.

[50] A. Wedel, T. Pock, C. Zach, H. Bischof, and D. Cremers. An improved algorithm for TV-L1 optical flow. *Statistical and Geometrical Approaches to Visual Motion Analysis: International Dagstuhl Seminar*, pages 23–45, 2009.

[51] J. Wright, A.Y. Yang, A. Ganesh, S.S. Sastry, and Y. Ma. Robust face recognition via sparse representation. *IEEE transactions on pattern analysis and machine intelligence*, 31(2):210–27, February 2009.

[52] G.H. Yang, D. Shen, and V.O.K. Li. UEP for video transmission in space-time coded OFDM systems. *IEEE INFOCOM*, pages 1200–1210, 2004.

[53] Z. Zhang and S.G. Mallat. Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, 41(12):3397–3415, 1993.