Universidade Federal do Rio de Janeiro

Escola Politécnica

Departamento de Eletrônica e de Computação

# *Compressive Sensing*

## Novos Paradigmas para Aquisição e Compressão de Imagens

**Adriana Schulz**

Projeto apresentado ao Departamento de Engenharia Eletrônica e de Computação da Escola Politécnica da UFRJ como requisito parcial para obtenção do título de Engenheira Eleterônica e de Computação

**Rio de Janeiro**

**2008**

UNIVERSIDADE FEDERAL DO RIO DE JANEIRO
ESCOLA POLITÉCNICA
DEPARTAMENTO DE ELETRÔNICA E DE COMPUTAÇÃO

# *Compressive Sensing*

## Novos Paradigmas para Aquisição e Compressão de Imagens

Autora:

Adriana Schulz

Orientador:

Prof. Eduardo Antônio Barros da Silva, Ph.D.

Orientador:

Prof. Luiz Carlos Pacheco Rodrigues Velho, Ph.D.

Examinador:

Prof. Gelson Vieira Mendonça, Ph.D.

Examinador:

Prof. Lisandro Lovisolo, D.Sc.

**DEL**

**Dezembro de 2008**

ב"ה

*A meu avô, Isaac Hilf,*
*de abençoada memória*

# Agradecimentos

Aos professores, Eduardo Barros da Silva e Luiz Velho, pelos ensinamentos, pelo estímulo, pela dedicação e pela orientação deste trabalho.

Aos professores Luiz Wagner Biscainho, Marcello Campos e Paulo Diniz, pelos ensinamentos e por despertar meu interesse em Processamento de Sinais.

Ao professor Carlos D'avila, pelos ensinamentos e pelo apoio recebido ao longo de todo curso.

Aos professores Gelson Mendonça e Sérgio Lima Netto, pelos ensinamentos e pela oportunidade de desempenhar atividades didáticas através do programa de monitoria.

Aos colegas do LPS e do Visgraf, pela presença sempre alegre e pelas enriquecedoras discussões.

A minha família, pelo amor e pela compreensão, tão fundamentais para meu crescimento.

# Resumo

Este trabalho aborda um novo paradigma que contraria os cânones usuais em aquisição de dados. A teoria de *Compressive Sensing* (CS) assegura que é possível recuperar sinais esparsos a partir de um número bem menor de amostras que as necessárias nos métodos clássicos, e usa para isto protocolos de sensoriamento não-adaptativos.

Além de ser um tema recente que tem causado grande impacto na comunidade científica por representar uma quebra de paradigma na área de amostragem e aquisição de dados, a teoria se torna interessante na medida em que envolve ferramentas matemáticas importantes e noções de aquisição, compressão, redução de dimensionalidade e otimização.

Os fundamentos de CS expostos neste trabalho poderão ser utilizados como guia bibliográfico para aqueles que se iniciam nesta área. Além disso, os exemplos elaborados permitem a avaliação do desempenho da técnica em diferentes contextos.

**Palavras-Chave: processamento de imagens, compressão de dados, amostragem, representação de sinais, transformadas, programação linear.**

# Abstract

This work addresses a new paradigm that rises against the common knowledge of data acquisition. The theory of *Compressive Sensing* (CS) gives a stable and robust algorithm that allows sensing at rates much smaller then the Nyquist limit by means of *non-adaptive* protocols.

In addition to being a novel idea that has had a great impact on the academic community, it is a very rich theory that covers interesting mathematical tools as well as notions of acquisition, compression, dimensional reduction and optimization.

Here are presented the fundamental aspects involved in CS which may be used as a bibliographic guide for those who are initiating on this field. Moreover, the elaborated examples allow the evaluation of CS performance in different acquisition scenarios.

**Keywords: image processing, data compression, sampling, signal representation, transforms, linear programing.**

*“Don’t just read it; fight it! Ask your own questions, look for your own examples, discover your own proofs.”*

Paul Halmos

# Sumário

# Lista de Figuras

# Lista de Abreviações

| | |
|---|---|
| **CS** | *Compressive Sensing* |
| **KLT** | *Karhunen-Loève Transform* |
| **PCA** | *Principal Components Analysis* |
| **DCT** | *Discrete Cosine Transform* |
| **DWT** | *Discrete Wavelet Transform* |
| **RIP** | *Restricted Isometry Property* |
| **UUP** | *Uniform Uncertanty Principle* |
| **TV** | *Total Variation* |

# Capítulo 1

# Introdução

Aquisição e reconstrução de sinais são essenciais em qualquer sistema de processamento de sinais, e teoremas de amostragem promovem a ponte entre os domínios contínuos e discretos. O principal teorema que estabelece um limite para a taxa de amostragem de um sinal garantindo sua reconstrução é o teorema de Shannon-Nyquist para sinais de banda limitada.

Observa-se, entretanto, que sinais naturais tendem a ser compressíveis, i.e., se amostrados pontualmente, muitos dos coeficientes adquiridos são redundantes. Assim, é o foco de muitos trabalhos de pesquisa reescrever os dados amostrados de forma a reduzir número de bits necessários para representá-los. Estes métodos realizam o que é designado por compressão.

O modelo de amostragem seguido de compressão é muito eficiente e é usado em muitas aplicações com bom desempenho. No entanto, a possibilidade de comprimir os dados adquiridos sugere que Nyquist era pessimista, pois considerava o pior cenário no qual tudo que se assume sobre os sinais é a limitação em banda. Mas, e se, ao invés de considerar a taxa de Nyquist, tentarmos recuperar os dados por sensoriamento na taxa de informação?

É a isso que se refere *Compressive Sensing* (CS) [1]. Esta teoria surge como um

[1] Como a teoria em questão é muito nova, ela ainda não tem uma denominação definitiva estabelecida. De fato, os pesquisadores da área usam os termos *Compressive Sensing* e *Compressible Sampling* de maneira intercambiável. Por esse motivo, decidimos não traduzir o título do trabalho

novo paradigma que assegura ser possível recuperar sinais esparsos a partir de um número bem menor de amostras que as necessárias nos métodos clássicos, e usa para isto protocolos de sensoriamento *não-adaptativos*.

Além de ser um tema recente, que tem causado grande impacto na comunidade científica, por representar uma quebra de paradigma na área de amostragem e aquisição de dados, a teoria se torna interessante na medida em que envolve ferramentas matemáticas importantes e noções de aquisição, compressão, redução de dimensionalidade e otimização.

## 1.1 Objetivos

O objetivo deste trabalho é fazer uma exposição dos fundamentos de CS que poderá ser usada como guia bibliográfico para aqueles que se iniciam nesta área. Assim, sua importância está na introdução do estudo de CS na Universidade Federal do Rio de Janeiro e no estímulo a novos projetos de pesquisa relacionados.

Além disso, tomamos o cuidado de elaborar exemplos de aplicações em diferentes cenários de aquisição, o que nos permitiu responder a algumas questões interessantes e avaliar o desempenho da técnica.

## 1.2 Organização

O Capítulo 2 expõe métodos clássicos para compressão de imagens que utilizam o paradigma de amostragem seguido de compressão. São estudados métodos que utilizam transformadas na tentativa de explorar a redundância de sinais naturais para mapear os dados em coeficientes menos correlacionados e, portanto, esparsos.

Crescendo em níveis de abstração, este modelo de compressão é relacionado, no Capítulo 3, a métodos de representação e reconstrução de sinais, que são então estudados com um enfoque especial à teoria de aproximação.

e, assim adotar a expressão “Compressive Sensing” em inglês, a qual nos parece mais apropriada. Será utilizada também a abreviação “CS” para designar o tema.

Com estas primeiras análises, fica estabelecido o embasamento para a investigação de CS, que é realizada no Capítulo 4.

No Capítulo 5 são apresentadas conclusões e direções para trabalhos futuros.

# Capítulo 2

# Compressão de Imagens

Ao longo das ultimas décadas, a revolução de tecnologias de multimídia vem possibilitando o acesso a grandes quantidades de dados em situações adversas. Um fator chave que viabiliza estes procedimentos é a habilidade de exprimir informação em uma forma compacta.

Compressão de dados, portanto, propõe a redução do número de bits necessários para representar um sinal. Com este objetivo, são exploradas as estruturas dos dados (como esparsidade e redundância) e características dos usuários (como limitações das habilidades perceptuais dos seres humanos).

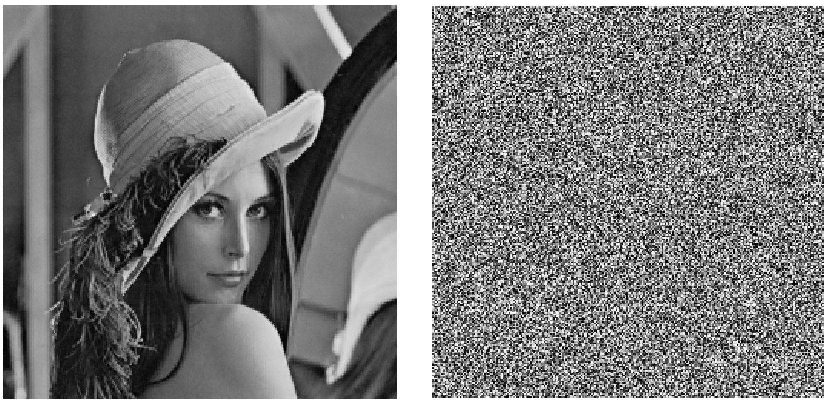
Para avaliar eficiência de compressão, podem ser levados em consideração propriedades dos algoritmos (esparsidade, velocidade, consumo de memória), o grau de compressão e a fidelidade da reconstrução em relação ao sinal original.

Neste trabalho, será utilizado o critério *taxa-distorção*, que avalia o compromisso entre o número médio de bits necessário para representar o valor de cada sinal e uma quantificação da diferença entre o sinal original e sua reconstrução após compressão.

Neste capítulo, serão estudados os elementos básicos e classificações de técnicas de compressão.

## 2.1 Codificação por Transformada

A maioria dos sinais observados na natureza apresentam alguma possibilidade de compressão. Este fato não é surpreendente se considerarmos que a redundância facilita a percepção humana. Por exemplo, é mais fácil e prazeroso ler um texto com algum nível de repetição, escutar músicas sem variações abruptas e assistir vídeos com pouca diferença entre quadros. A mesma coisa acontece com imagens, onde pixels adjacentes costumam apresentar similaridades. A Figura 2.1 compara uma imagem não redundante (direita) com uma redundante (esquerda).



(a) Imagem *lena*.

(b) Ruído branco Gaussiano.

Figura 2.1: Na imagem *lena*, pixels que não pertencem à região de contorno são muito similares aos adjacentes. O ruído branco, por sua vez, não é compressível. (Extraído de [1].)

A existência de redundância indica que armazenar uma imagem como uma matriz de pixels na qual cada coeficiente corresponde à sua intensidade é ineficiente, uma vez que muitos valores serão equivalentes.

A solução é encontrar uma representação esparsa, i.e., uma representação na qual a informação está concentrada em apenas poucos coeficientes, os outros sendo nulos. Se este objetivo for alcançado, o número de coeficientes que deverão ser armazenados (ou transmitidos) será altamente reduzido.

Codificação por transformada [2] é o nome dado a técnicas de compressão de imagens que aplicam modificações na representação de sinais no sentido de minimizar redundância. A Figura 2.2 introduz as três operações básicas da codificação por transformada.

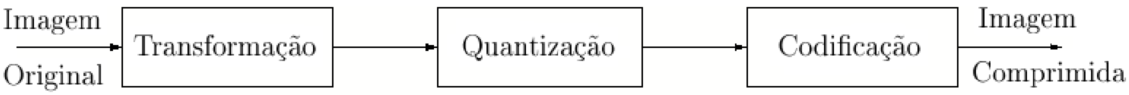


Figura 2.2: Operações d codificação por transformada.

A *transformação* da imagem em um conjunto de coeficientes com pouca redundância é o primeiro passo do processo de compressão. Simultaneamente, ela minimiza a correlação entre coeficientes e maximiza a concentração de energia. No entanto, a obtenção de uma matriz com muitos zeros não é suficiente para reduzir o número de bits necessários para a reconstrução do sinal.

É interessante enfatizar que valores de pixels geralmente variam entre 0 e 255, i.e, cada pixel é representado por 8 bits. Entretanto, após aplicada uma transformação, os coeficientes podem assumir valores arbitrários em ponto-flutuante. Além disso, estas operações usualmente geram um grande número de coeficientes muito pequenos, mas não nulos.

Ambos estes problemas são solucionados durante o procedimento de *quantização*, que tem como objetivo representar uma grande faixa de valores a partir de um conjunto relativamente pequeno de símbolos. Apesar de diminuir fortemente a taxa, este processo geralmente acarreta perdas de informação.

O último passo tem como objetivo mapear os símbolos na menor seqüência de bits possível. Este procedimento, denominado *codificação*, leva em consideração as características estatísticas dos símbolos e a posição na matriz dos coeficientes significativos (não negativos).

Maiores detalhes sobre cada uma destas etapas, assim como a descrição de dois dos principais padrões de compressão de imagem, encontram-se no Apêndice B.

## 2.2 Classificação de Técnicas de Compressão

Muitos autores distinguem entre técnicas de compressão sem perdas e com perdas, a primeira indicando representações inversíveis e, a segunda, representações nas quais parte da informação é perdida. Como a quantização envolve distorções, fica claro que este trabalho está focado em esquemas de compressão com perdas.

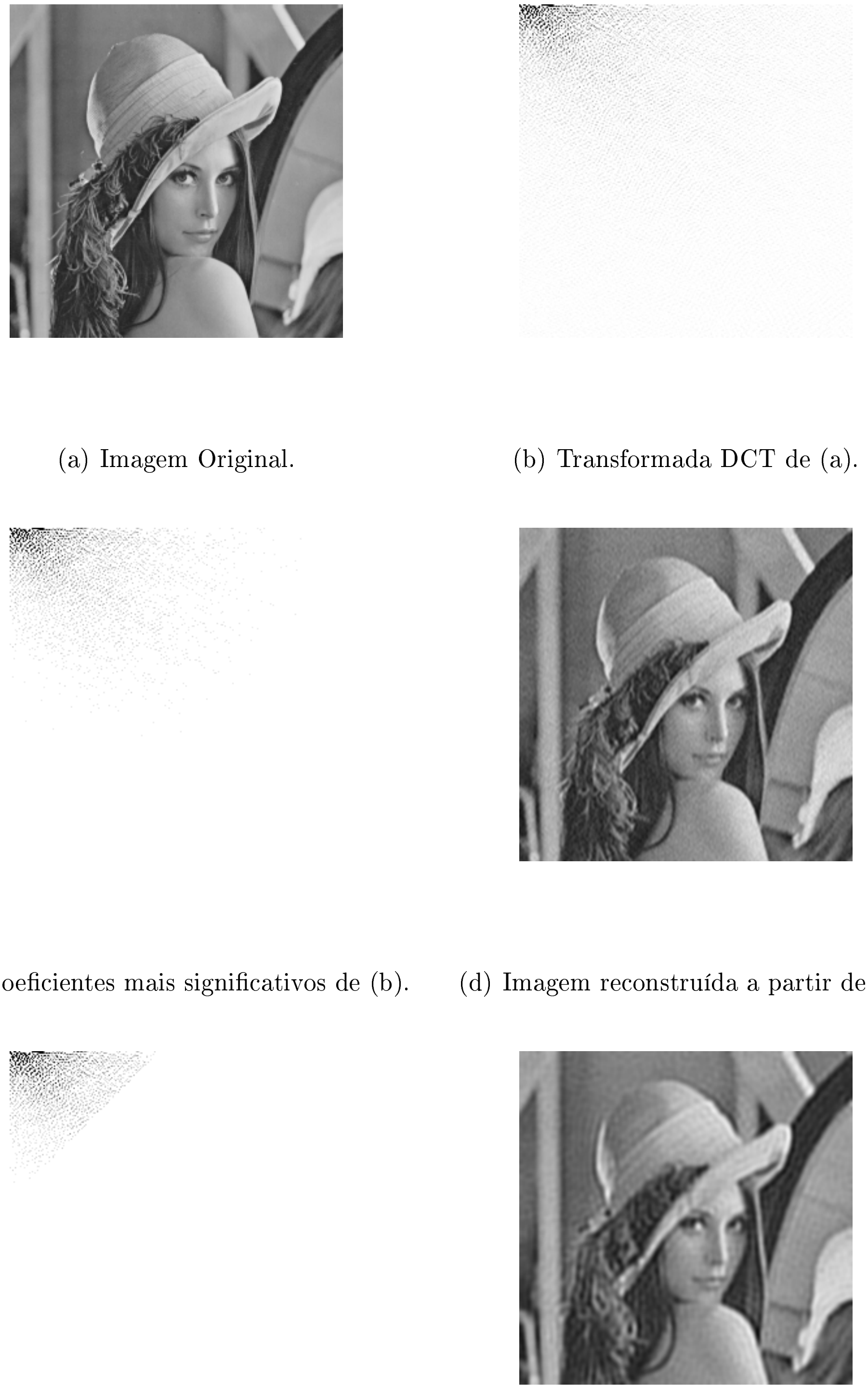
Justifica-se o uso deste tipo de compressão para imagens, pois métodos com perdas possibilitam menores taxas, e o sistema visual humano não é sensível a pequenas distorções.

Também é possível classificar a compressão em não-linear e linear. A primeira explora a esparsidade de uma dada imagem antes de quantizar e codificá-la e, a segunda, é cega, i.e., não é necessário saber *a priori* a posição dos coeficiente significativos.

Em outras palavras, se $A$ e $B$ são imagens e $\hat{A}$ e $\hat{B}$ são suas versões comprimidas, a compressão linear de $A + B$ resulta em $\hat{A} + \hat{B}$. Já em esquemas de compressão não lineares, esta propriedade não pode ser garantida.

Na Figura 2.3 compara-se a reconstrução da imagem *lena* com 1 de cada 10 coeficientes usando compressão via DCT[1] linear e não-linear. Em 2.3(c) zera-se os menores coeficientes da transformada DCT e em 2.3(e) zera-se os coeficientes da DCT que não estão posicionados no canto superior esquerdo da matriz de transformada. As imagens 2.3(d) e 2.3(f) são reconstruídas a partir da DCT inversa de 2.3(c) e 2.3(e), respectivamente. Pode-se concluir que, apenas de métodos não lineares serem mais simples e não exigirem o armazenamento da posição dos coeficientes significativos, sua eficiência é razoavelmente menor.

[1] A Tra Transformada Discreta de Cosseno (*Discrete Cosine Transform* - DCT) está descrita no Apêndice B.



(a) Imagem Original.

(b) Transformada DCT de (a).

(c) Coeficientes mais significativos de (b).

(d) Imagem reconstruída a partir de (c).

(e) Coeficientes de (b) no canto superior esquerdo.

(f) Imagem reconstruída a partir de (e).

Figura 2.3: Exemplo de imagem reconstruída com 1 de cada 10 coeficientes via DCT linear e não-linear.

# Capítulo 3

# Representações de Sinais

Representação é um aspecto fundamental em processamento de sinais que propõe a descrição completa e não ambígua de um sinal como uma seqüência de coeficientes. A importância deste procedimento pode ser associada à natureza contínua de sinais existentes, que precisa ser superada para possibilitar o processamento digital.

Discretização, no entanto, não é o único benefício buscado. Boas representações de sinais possibilitam uma série de processos, como análise, filtragem de ruído e compressão. A idéia é que, dependendo de como um sinal é descrito, alguns de seus aspectos podem ser enfatizados, i.e, pode-se distribuir as informações de interesse entre componentes específicos e, assim, facilitar o acesso às mesmas.

Neste capítulo, serão exploradas diferentes maneiras de representar sinais e serão analisadas suas características básicas assim como métodos de reconstrução. Maiores detalhes sobre este assunto encontram-se no Apêndice C.

## 3.1 Paralelo com Compressão de Imagens

No capítulo anterior, foi estudada a codificação por transformada como um método de comprimir imagens expondo a mesma informação em um número menor de coeficientes. É interessante constatar que explorar redundância no sentido de mapear os dados em um conjunto de coeficientes menos correlacionados é equivalente a escolher uma nova representação para o sinal.

## 3.2 Decomposições de Sinais

Definimos uma representação como uma função $R : \mathbf{H} \rightarrow \mathcal{S}$ que mapeia um espaço de Hilbert[1] $\mathbf{H}$ em um espaço de seqüências. Para um dado sinal, $x \in \mathbf{H}$, sua representação $R(x)$ é uma seqüência:

$$R(x) = (s_1, s_2, s_3...) \in \mathcal{S}$$

onde $s_n$ é um par $(\alpha_n, g_{\gamma_n})$, o primeiro representando um coeficiente e, o segundo, uma forma de onda.

Associado a $R$ está um conjunto de funções $\mathcal{D} = (g_\lambda)_{\lambda \in \Gamma}$ denominado *dicionário*. Note que o dicionário pode ser não enumerável. Entretanto, $(g_{\gamma_n})_{n \in \mathbb{Z}}$, usado na representação de um sinal particular $x$, consiste em um subconjunto enumerável.

Em alguns casos, a função $R$ é inversível e o sinal $x$ pode ser perfeitamente reconstruído com base em sua representação $R(x)$. Quando isto ocorre, diz-se que a representação é *exata* e o sinal original é obtido a partir da combinação linear

$$x = \sum_{n \in \mathbb{Z}} \alpha_n g_{\gamma_n}$$

No entanto, quando a representação não é exata, é possível fazer uso de certas técnicas para aproximar a reconstrução de $x$.

A dimensão $N$ de um espaço $\mathbf{H}$ está associado ao número de elementos do dicionário que são necessários para a geração do mesmo. Um bom esquema de representação exige o uso de um dicionário *completo*, i.e., qualquer função em $\mathbf{H}$ pode ser expandida como a combinação das funções $(g_\lambda)_{\lambda \in \Gamma}$. É notável, porém, que o tamanho do dicionário pode ser maior que N. Neste caso, diz-se que o dicionário é redundante pois existe mais de uma forma para representar o mesmo sinal. É importante enfatizar que, em alguns casos, trabalha-se com dimensões infinitas.

[1] Um espaço de Hilbert é um espaço com produto interno que é completo como espaço métrico, i.e., um espaço vetorial abstrato no qual distâncias e ângulos podem ser medidos e que é completo, significando que se uma seqüência de vetores tente a um limite, este limite também pertence ao espaço.

Portanto, para a decomposição de um sinal, é necessário obter a seqüência de formas de onda do dicionário $(g_{\lambda_n})_{n\in\mathbb{Z}}$ e os coeficientes correspondentes. Existem muitos métodos que atingem este objetivo explorando propriedades de determinados tipos de sinais, como mencionado anteriormente.

Em seguida, serão distinguidos dois modelos de representação: bases e *frames.*

### 3.2.1 Bases

Uma base é um conjunto de elementos linearmente independentes $(\phi_\lambda)_{\lambda\in\Gamma}$ que geram o espaço de Hilbert $\mathbf{H}$. Note que a independência linear implica que o conjunto seja mínimo.

### 3.2.2 *Frames*

*Frames* são generalizações do conceito de bases. Um *frame* consiste em uma família de vetores $(\phi_\lambda)_{\lambda\in\Gamma}$ que caracteriza qualquer sinal $x$ em um espaço de Hilbert $\mathbf{H}$ a partir de seu produto interno $\{\langle x, \phi_\lambda\rangle\}_{\lambda\in\Gamma}$, onde o conjunto de índices $\Gamma$ pode ser finito ou infinito.

A teoria de *frames*, desenvolvida por Duffin and Schaeffer, determina uma condição para que o *frame* defina uma representação completa e estável:

**Definition 1.** *Uma seqüência* $(\phi_\lambda)_{\lambda\in\Gamma}$ *é um* frame *de* $\mathbf{H}$ *se existem duas constantes* $A > 0$ *e* $B > 0$ *tal que para cada* $x \in \mathbf{H}$

$$A\|x\|^2 \leq \sum_{\lambda\in\Gamma} |\langle x, \phi_\lambda\rangle|^2 \leq B\|x\|^2$$

*Quando* $A = B$ *diz-se que o* frame *é apertado.*

É importante enfatizar que a representação por *frame* pode ser redundante.

## 3.3 Teoria de Aproximação

A possibilidade de gerar representações a partir de diferentes bases é útil para processamento de dados, uma vez que permite a aproximação de certos tipos de sinais por apenas poucos vetores.

### 3.3.1 Aproximação em uma Base Linear

Dado um sinal $x$ e uma base ortogonal $\mathcal{B} = (\phi_\lambda)_{\lambda \in \Gamma}$, uma aproximação projeta $x$ sobre $M$ vetores da base

$$x_M = \sum_{n \in I_M} \langle x, \phi_n \rangle \phi_n \tag{3.1}$$

A escolha dos $M$ vetores pode ser feita *a priori* ou *a posteriori* (dependendo do sinal $x$). No primeiro caso, a aproximação é dita linear e, no segundo, não-linear.

Apesar de aproximações lineares serem mais simples de implementar, distorções geradas dependerão altamente do sinal de entrada, enquanto, no caso não-linear, os vetores de projeção podem ser adaptados de forma a minimizar o erro de aproximação.

### 3.3.2 Aproximação em Dicionários Super-completos

Expansão linear em uma única base pode não ser eficiente, pois a informação estará diluída ao longo de toda a base. Em dicionários super-completos, entretanto, é possível expressar o mesmo sinal usando um número menor de coeficientes. Mallat ilustra esta idéia [3] comparando representações de sinais a vocabulários lingüísticos. Enquanto um pequeno vocabulário pode ser suficiente para expressar qualquer idéia, em alguns casos, serão necessárias frases inteiras para substituir palavras disponíveis apenas em dicionários maiores.

Devido à redundância, existem, no entanto, maneiras inumeráveis de representar um mesmo sinal. Por consequência, o objetivo de técnicas que usam estes dicionários é encontrar uma representação que concentre a informação em um pequeno número de coeficientes.

# Capítulo 4

# Amostragem Compressiva (*Compressive Sensing*)

Até o momento, estudamos o paradigma de amostragem seguido de compressão, i.e., para uma dada imagem, encontra-se uma representação esparsa e, em seguida, codifica-se os coeficientes. A desvantagem deste método deriva da necessidade de:

- armazenar um grande número de amostras;
- computar todos os coeficientes da transformada; e
- encontrar a localização dos maiores coeficientes.

Este é o procedimento utilizado em grande parte dos instrumentos de captura de dados modernos. Câmeras digitais comuns, por exemplo, capturam um grande número de amostras (i.e., da ordem de mega-pixels), mas apenas armazenam uma versão comprimida da imagem no formato JPEG. Logo, esta técnica desperdiça a maior porcentagem dos dados adquiridos e, portanto, constata-se uma perda de eficiência.

Isto sugere que métodos mais inteligentes e computacionalmente menos custosos podem ser aplicados para solucionar o problema de aquisição de informação. Neste contexto, surge *Compressive Sensing*, que se baseia na amostragem do sinal original a uma taxa razoavelmente menor que o limite de Nyquist e na reconstrução por meio de otimização convexa.

## 4.1 Aspectos Essenciais

O objetivo é construir um esquema de aquisição que capture a imagem já em sua forma comprimida. Considere, por exemplo, o método de compressão baseado na transformada DCT. Se fosse conhecida *a priori* a posição dos coeficientes mais significativos da DCT (como em um esquema linear de compressão), seria possível simplesmente medir seus valores e desconsiderar a exploração de outras informações.

Note que a palavra *amostra* assume um novo significado. Neste contexto, substitui-se amostragem pontual por medidas lineares mais genéricas dos sinais. Cada medida, $y_m$ do sistema de aquisição é o produto interno do sinal $x$ com uma função de teste diferente $\phi_m$ (por exemplo, uma coluna da matriz de transformada DCT). Ou seja,

$$y_1 = \langle x, \phi_1 \rangle \, , \quad y_2 = \langle x, \phi_2 \rangle \, , \quad \ldots \quad , \quad y_M = \langle x, \phi_M \rangle$$

onde $M$ é o número de medidas.

Entretanto, como foi visto nos capítulos anteriores, aproximações lineares geralmente apresentam desempenhos que estão longe do ótimo. Assim, apesar de $x$ ser esparso em algum domínio, não se pode saber com certeza onde se encontram os coeficientes significativos. Além disso, é desejável obter um solução *não-adaptativa* para o problema de forma a possibilitar o uso do mesmo procedimento de captura para qualquer sinal.

### 4.1.1 O Problema Algébrico

Seja $s$ um sinal representado em um domínio esparso, i.e,

$$s = \Psi x$$

onde $x$ é o sinal original e $\Psi$ é a transformação que torna $s$ esparso, por exemplo, a DCT.

Tomar poucas medidas equivale a multiplicar $x$ por uma matriz gorda[1] $\Phi_\Omega$, como

[1] Usamos o termo *gorda* para fazer referência a matrizes onde o número de colunas excede o número de linhas.

ilustrado na Figura 4.1, onde cada linha corresponde a uma função de medida $\phi_m$.

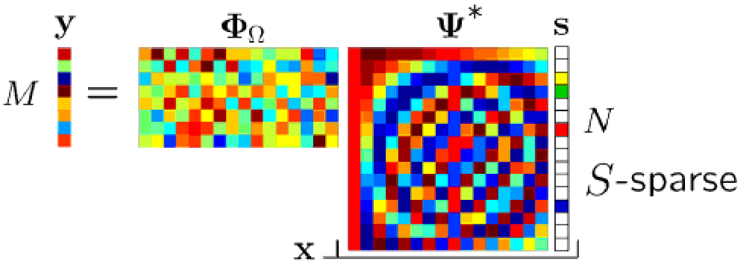


Figura 4.1: A matriz de aquisição. (Extraída de [4].)

$$y = \Phi_\Omega x$$

$$x = \Psi^* s \Longleftrightarrow s = \Psi x$$

$$y = \Theta_\Omega s, \text{ where } \Theta_\Omega = \Phi_\Omega \cdot \Psi^*$$

O algoritmo de reconstrução envolve encontrar $x$ tal que $y = \Phi_\Omega x$, ou, analogamente, $s$ tal que $y = \Theta_\Omega s$. Este problema, no entanto, é mal condicionado, pois existe uma infinidade de soluções possíveis. Apesar isso, nem todas as soluções satisfazem a propriedade de esparsidade de $s$ e, portanto, uma escolha simples consistiria em procurar, entre todas as soluções possíveis, aquela que torna $s$ esparso.

### 4.1.2 Esparsidade e a Norma $l_1$

Esparsidade pode ser descrita a partir da norma $l_0$

$$\|\alpha\|_{l_0} = \sharp\{i : \alpha(i) \neq 0\}$$

Por conseqüência, a solução desejada é

$$\min_x \|\Psi x\|_{l_0} \quad \text{sujeito a} \quad \Phi_\Omega x = y$$

Ou, alternativamente,

$$\min_s \|s\|_{l_0} \quad \text{sujeito a} \quad \Theta_\Omega s = y$$

Apesar deste problema ser combinatório e NP-complexo, é possível provar que sinais esparsos possuem normas $l_1$ relativamente pequenas. A Figura 4.2 motiva a relação entre esparsidade e norma $l_1$.

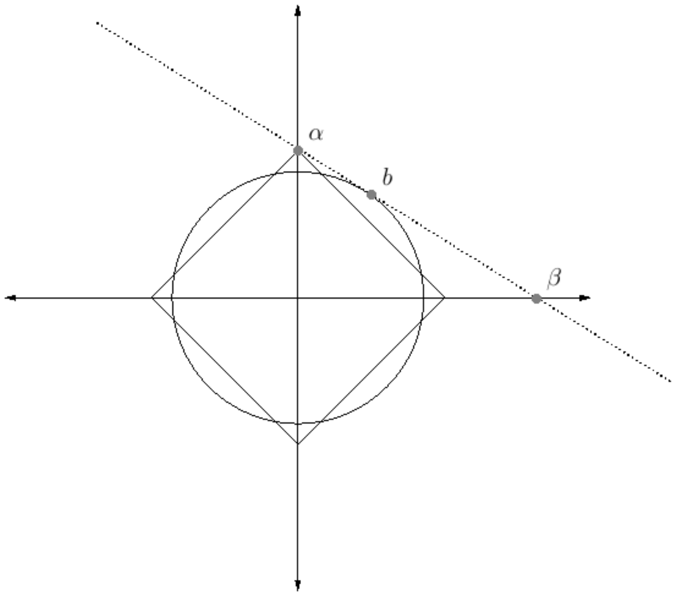


Figura 4.2: Esparsidade e norma $l_1$.

Considere a busca pelo sinal $s$ que possua a menor norma $l_0$ e respeite a equação linear que restringe sua posição em $\mathbb{R}^2$ à linha pontilhada. Note que a minimização da norma $l_2$ gera como solução ótima $s = b$, que está distante das soluções esparsas $\alpha$ and $\beta$. Por outro lado, a minimização da norma $l_1$ resulta em $s = \alpha$, que é a solução exata desejada.

A norma $l_1$ é convexa, o que torna o problema de otimização computacionalmente tratável. Sendo assim, as análises e resultados enunciados a seguir serão baseados na minimização desta norma.

### 4.1.3 O Algoritmo de Reconstrução

A partir deste ponto, pode-se entender a idéia de *Compressive Sensing* em termos de seu algoritmo de reconstrução. A teoria envolve tomar apenas poucas medidas de um sinal e recuperá-lo a partir da solução de um problema do otimização convexa

$$\min_{s} \|s\|_{l_1} \quad \text{sujeito a} \quad \Theta_\Omega s = y$$

Apesar de estarem claro os motivos que justificam o uso deste procedimento, ainda é necessário avaliar sua eficiência. Como é possível garantir que a solução esparsa é aquela que reconstrói o sinal original? Quais as características que devem ser assumidas a respeito da matriz de medidas e do número de amostras? Que tipos de resultados podem ser garantidos?

Uma série de teoremas e definições foram propostos com o objetivo de formalizar esta idéia e especificar condições que garantam bons resultados. Eles serão discutidos na próxima seção. Detalhes sobre o surgimento da teoria e relações com conceitos previamente estabelecidos encontram-se no Apêndice D.

## 4.2 Aspectos Teóricos

Usaremos para este estudo duas abordagens diferentes:

- CS Básico - teoria que estipula restrições para a recuperação exata de sinais esparsos.
- CS Robusto - expansão da abordagem anterior para possibilitar aplicações de CS a sinais que não são exatamente esparsos e cujas medidas estão corrompidas por ruído.

### 4.2.1 CS Básico

CS Básico lida com a análise de restrições que garantem a perfeita reconstrução a partir da minimização da norma $l_1$, considerando que existe um domínio no qual o sinal $x$ é $S$-esparso[2] e que as medidas não estão corrompidas por ruído.

[2] Notação:

Usamos $x$ para fazer referência ao sinal de entrada e $s$ para denotar a representação $S$-esparsa. $T$ é o subconjunto de $\mathbb{R}^N$ ao qual $s$ pertence e possui tamanho $|T| = S$. $\Omega$ é o subconjunto randômico onde as medidas são tomadas e têm tamanho $|\Omega| = M$.

Uma importante definição utilizada para a formalização da teoria de CS é a *coerência*, que define uma medida da correlação entre as funções de sensoriamento, $\phi_k$, e as funções que geram o domínio onde o sinal é esparso, $\psi_k$. A definição segue assumindo que ambos têm norma $l_2$ unitária.

**Definição 1** (Coerência entre $\Psi$ and $\Phi$ [5])**.**

$$\mu(\Phi, \Psi) = \sqrt{N} \max_{i,j} |\langle \phi_i, \psi_j \rangle| \quad , \quad \|\phi_i\|_{l_2} \quad \|\psi_i\|_{l_2} = 1$$

Também pode-se definir coerência a partir da matriz $\Theta$.

**Definição 2** (Coerência mutua [6])**.**

$$\mu(\Theta) = \sqrt{N} \max_{i,j} |\Theta_{i,j}|$$

É importante constatar que melhores resultados são obtidos quando a coerência é pequena, i.e., quando os dois domínios são altamente descorrelacionados. Esta observação pode ser motivada pelo grande número de coeficientes nulos que são retornados quando amostras são feitas diretamente no domínio onde o sinal é esparso. A vantagem da incoerência é que, ao adquirir uma série de combinações aleatórias das entradas, aprende-se algo novo sobre o sinal esparso a cada medição.

Com base nesta definição pode-se enunciar o principal teorema de CS básico.

Denota-se por $\Phi$ a matriz que gera $\mathbb{R}^N$, onde cada linha é uma função de medição $\phi_m$ que será aplicada ao sinal $x$. Assim, tomar poucas amostras equivale a fazer

$$y = \Phi_\Omega x$$

onde $\Phi_\Omega$ é uma matriz gorda gerada a partir da seleção aleatória de M linhas de $\Phi$. Como $x$ é esparso no domínio $\Psi$, a representação esparsa de $x$ é dada por

$$s = \Psi x$$

E portanto, como $\Psi$ é uma matriz unitária (transformação ortogonal),

$$y = \Phi_\Omega \Psi^* s$$
$$\Rightarrow y = \Theta_\Omega s, \text{ where } \Theta_\Omega = \Phi_\Omega \Psi^*$$

Também denota-se $\Theta = \Phi\Psi^*$ e $\Theta_{\Omega T}$ é a matriz gerada a partir da extração das $S$ colunas de $\Theta_\Omega$ correspondentes aos índices de $T$.

**Teorema 1** ([7]). *Seja $\Theta$ uma matriz ortogonal $N \times N$ e $\mu(\Theta)$ como definido previamente. Fixe o subconjunto $T$ do domínio do sinal. Escolha um subconjunto $\Omega$ do domínio de medições de tamanho $M$ e uma seqüência de sinais $z$ em $T$ uniformemente aleatória. Suponha que*

$$M \geq C_0 \cdot |T| \cdot \mu^2(\Theta) \cdot \log(N)$$

*para uma constante numérica $C_0$. Então, para qualquer função $s$ em $T$ com sinais correspondendo a $z$, a recuperação a partir de $y = \Theta_\Omega s$ e a solução de*

$$\hat{s} = \min_{s^*} \|s^*\|_{l_1} \quad \textit{subject to} \quad \Theta_\Omega s^* = y$$

*É exata ($\hat{s} = s$) com altíssima probabilidade.*

### 4.2.2 CS Robusto

Geralmente, sinais naturais não são esparsos, mas são aproximadamente esparsos ou possuem um decaimento exponencial. Além disso, as medições não são perfeitas e geralmente algum nível de ruído é adicionado a elas. Para que a teoria de CS possa ser aplicada a situações reais, ela deve ser robusta a dois tipos de erros. Por isso, grande esforço foi feito no sentido de definir condições e teoremas que garantam a expansão da teoria.

Nesta seção, serão apresentados teoremas que garantem a robustez de CS a aplicações nas quais:

- o sinal não é exatamente esparso; ou
- medições estão corrompidas por ruído.

Para isto é necessário definir a Propriedade de Isometria Restrita (*Restricted Isometry Property* - RIP).

**Definição 3** (Constante de Isometria Restrita [8]). *Para cada inteiro $S = 1, 2, \ldots, N$ define-se a constante de isometria $S$-restrita $\delta_S$ de uma matriz $\Theta_\Omega$ como o menor número tal que*

$$(1 - \delta_S)\|s\|_{l_2}^2 \leq \|\Theta_{\Omega T} s\|_{l_2}^2 \leq (1 + \delta_S)\|s\|_{l_2}^2$$

*para todos vetores $S$-esparsos.*

A isometria restrita é uma propriedade da matriz de medições $\Theta_\Omega$ que se refere à existência e limitação de $\delta_S$. A RIP estabelece uma condição que, se obedecida por $\Theta_\Omega$, garante a recuperação de sinais esparsos. Note que a constante $\delta_S$ é intrínseca à estrutura de $\Theta_\Omega$ e, portanto, ao definir restrições para seu tamanho, é possível quantificar a eficiência da matriz de aquisição.

A razão do nome RIP é simples: a energia do sinal restrito ao conjunto $\Omega$ é proporcional ao tamanho de $\Omega$. No entanto, alguns autores o descrevem esta propriedade como um Princípio Uniforme da Incerteza (*Uniform Uncertainty Principle* - UUP) porque ela garante que o sinal não pode ser concentrado, simultaneamente, em ambos os domínios de esparsidade e de medições.

Seja $s$ um sinal apenas aproximadamente esparso e $s_S$ a melhor aproximação $S$-esparsa de $s$, i.e, o resultado obtido quando força-se os $N-S$ menores coeficientes de $s$ a serem zero. Considere também que as medidas $y$ estão corrompidas pela adição de um ruído $n$ limitado por $\|n\|_{l_2} \leq \epsilon$, i.e.,

$$y = \Phi x + n$$

A partir da propriedade RIP, pode-se enunciar o seguinte resultado.

**Teorema 2** ([9]). *Considere que $y = \Theta_\Omega s + n$ onde $\|n\|_{l_2} \leq \epsilon$. Assim, se $\delta_{2S} < \sqrt{2} - 1$, a solução $\hat{s}$ para*

$$\hat{s} = \min_s \|s\|_{l_1} \quad \textit{subject to} \quad \|\Theta_\Omega s - y\|_{l_2} \leq \epsilon$$

*obedece*

$$\|\hat{s} - s\|_{l_2} \leq C_0 s^{-1/2} \cdot \|\hat{s} - s_S\|_{l_1} + C_1 \epsilon$$

*para valores razoáveis das constantes $C_0$ e $C_1$.*

É importante observar que o erro de reconstrução é uma superposição de dois fatores: erros gerados pela aproximação da esparsidade e erros que resultam do ruído aditivo.

Maiores detalhes sobre estes teoremas encontram-se no Apêndice E.

## 4.3 Resultados

No Apêndice F, teoria de CS é verificada por meio de exemplos.

Apesar das imagens estarem já armazenadas no computador como uma matriz de pixels, métodos de aquisição são simulados a partir de medições que envolvem combinações lineares destes coeficientes.

Diferentes abordagens para a aquisição são avaliadas em termos de sua Variação Sinal Ruído de Pico (*Peak Signal to Noise Ratio* - PSNR) para diferentes quantidades de medidas, $M$.

O procedimento foi baseado nos resultados obtidos por [10] e os algoritmos de otimização utilizados foram baixados de http://www.acm.caltech.edu/l1magic [11].

desvantagem deste método deriva

# Capítulo 5

# Conclusões

Durante este trabalho, a teoria de *Compressive Sensing* foi introduzida como um novo paradigma para a aquisição de imagens. Nosso estudo envolveu a revisão de procedimentos padrões de sensoriamento e compressão com a intenção de familiarizar o leitor e motivar aplicações.

Foram analisados os mais importantes teoremas e definições que formalizam a teoria de CS e foram discutidos alguns argumentos relevantes que justificam a eficiência do procedimento. Finalmente, exemplos relacionados à compressão de imagens foram produzidos, possibilitando a avaliação da técnica em diferentes cenários .

Observou-se que diversos ajustes precisam ser feitos para possibilitar aplicações de CS a condições modernas de aquisição de dados, uma vez que a taxa de compressão do método é significativamente menor que a de modelos padrões de compressão e a estratégia de recuperação da informação é computacionalmente mais cara.

No entanto, nota-se que esta teoria tem muito potencial, uma vez que contraria os cânones da área e, desta forma, permite uma nova interpretação do problema de aquisição de dados. Isto sugere que aplicações em diferentes áreas podem e devem ser experimentadas.

## 5.1 Trabalhos Futuros

Em muitas publicações recentes [12, 13, 14], pesquisadores substituíram a norma $l_1$ pela norma de variação total(*total-variation* - TV). Face à relação entre a norma TV e o gradiente da norma $l_1$, há uma suposição comumente encontrada na literatura que os teoremas ainda são válidas sob esta condição [15]. A minimização da norma TV é muito eficiente quando aplicada a imagens, pois sugere certa suavidade que é normalmente encontrada em imagens naturais. Uma extensão deste estudo deve, portanto, considerar esta abordagem.

Também seria interessante experimentar alternativas para medições a partir de Noiselets. No futuro, pretendemos testar aquisição a partir de matrizes aleatórias com distribuição Gaussiana e funções de Whash-Hadamard.

A escolha de Wavelets ortogonais é decorrente da estrutura do algoritmo de reconstrução, que exige as matrizes $\Theta$ e sua transposta como entrada. Embora Wavelets biortogonais sejam mais adequadas para reforçar a esparsidade, as matrizes correspondentes a suas transformadas não são auto-adjuntas, tornando a implementação bastante difícil. No futuro, uma análise mais cuidadosa das ferramentas disponíveis no pacote $L_1$-Magic permitirá a utilização de matrizes não auto-adjuntas.

Também será interessante verificar como CS se comporta quando a imagem é dividida em blocos. Em todos os teoremas enunciados, o número de amostras necessárias cresce com um fator $\log N$. Apesar de pesquisadores afirmarem que podemos esperar uma recuperação para a maioria dos sinais em mais de 50 % dos casos se $M \geq 4S$ [12], seria interessante considerar aquisição de imagens muito grandes e comparar o desempenho do método com e sem o particionamento em blocos. brazilian

# Appendix A

# Introduction

Acquisition and reconstruction are essential in every signal processing system and sampling theorems are responsible for the bridge between continuous and discrete domains. The most important theorem that sets a limit to the sampling rate guaranteeing recovery is the Shannon-Nyquist theorem for band-limited signals.

We know, however, that natural and manmade signals tend to be compressible, i.e., if point sampled many of the acquired coefficients will be redundant. Hence, a lot of effort has been made in order to rewrite the sampled data reducing the number of bits required to represent it. These schemes perform what is referred to as compression.

The sample-then-compress framework is very efficient and is used in many applications with a good performance. However, the fact that we are able to compress the acquired data, suggests that Nyquist was a pessimist, who considered the worst case scenario in which all that is known is that the signals are band-limited. But what if, instead of considering the Nyquist rate, we would try to recover the data by sensing at the information rate?

This is what Compressive Sensing is about. It comes out as a new paradigm for data acquisition that rises against the common knowledge of the filed. In truth, it gives a stable and robust algorithm that allows sensing at rates much smaller then the Nyquist limit and recovering the signals with little corruption.

The basic idea is that compressibility translates in the existence of a representation in which the signal is sparse (most coefficients are zero). Therefore, while taking only a small number of samples would make the recovery problem ill-posed (an infinite number of solutions would be available), the compressibility property allows us to search in all possible solutions the one that makes the recovered signal sparse.

Of course, there is a twist in the word "sample". We cannot point sample the signal and hope to reconstruct it with very a small number of measurements because, once it is sparse, most of our acquired data will be zero. Instead, we measure the signal by calculating its inner product against different test functions.

Compressive sensing is intriguing not only because it proves that it is possible to reconstruct a signal with a very small number of measurements but also because it is *nonadaptive*. By this we mean that the algorithm is completely blind, not needing to guess characteristics of the original object (apart from sparsity). Moreover, the solution is obtained by means of a linear program that solves a convex optimization problem.

## A.1 Objectives

We were motivated to study CS, not only because it is a novel idea that has had a great impact in the academic community, but also because it is a very rich theory that covers interesting mathematical tools as well as notions of acquisition, compression, dimensional reduction and optimization.

The intention of this project is to develop a presentation of the fundamental aspects involved in CS which may be used as a bibliographic guide for those who are initiating on this field. Therefore, the relevance of this work is in the introduction of the study of CS in the Federal University of Rio de Janeiro and the stimulation of related research projects.

Moreover, we were careful to elaborate examples of applications in different acquisition scenarios. The latter allowed us to answer a few interesting questions and evaluate the performance of the technique.

## A.2 Organization

In Appendix B, we consider the classic methods for image compression which apply the sample-then-compress framework. We study schemes that make use of transforms (as the DCT and Wavelets) in order to exploit signal redundancy and map the data in coefficients that are less correlated and, therefore, sparse.

Growing in abstraction levels, this compression paradigm is related in Appendix C to signal representation and reconstruction models. The latter are then studied with emphasis in approximation theory.

With the former analysis, the stage is set for the investigation of Compressive Sensing. Nevertheless, before we examine the fundamental theorems, some effort is made in Appendix D to intuitively justify the combination of sensing and compression in a single procedure.

Based on the definition of the reconstruction algorithm, we must establish the characteristics that, imposed to the acquisition model, guarantee good performances. Hence, in Appendix E, a few parameters are defined and several theorems that evaluate CS in different contexts are exposed.

In Appendix F, we verify the CS theory by means of examples. We consider applications for image compression in scenarios where the signal is either sparse or only approximately sparse, as well as when measurements are corrupted by Gaussian and quantization noise.

In Appendix G we present some conclusion and directions for future work.

# Appendix B

# Image Compression

During the last decades we have been experiencing a multimedia revolution that has enabled us to access large amounts of data in adverse situations. A key ingredient that has made these technologies possible is the ability to express information in a compact form.

Data compression, therefore, aims at reducing the number of bits required to represent a signal by exploiting structures in the data (such as sparsity and redundancy) and characteristics of the users (such as the limited perceptual abilities of human beings).

To evaluate compression efficiency, it can be taken into account properties of the algorithm (complexity, velocity, memory consumption), the amount of compression, and how closely the reconstruction resembles the original.

In this work, we will focus on the *rate-distortion* criteria, that evaluate the trade-offs between the average number of bits used to represent each sample value and a quantification of the difference between the original signal and its reconstruction after compression.

Figure B.1 illustrates a *rate-distortion* function $R(D)$ that specifies the lowest rate at which the output of a source can be encoded while keeping the distortion less than or equal to $D$. This function is very useful because it defines a bound and therefore a way to determine optimality given a particular source. It will not always be possible to design optimal compression scheme and thus the goal of many

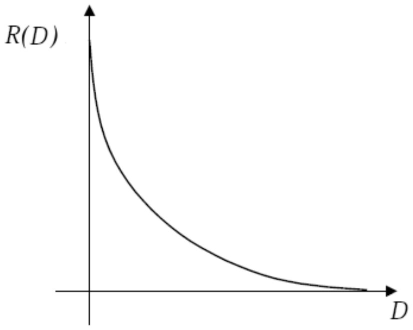


Figure B.1: The *rate-distortion* function.

researchers in this area is to improve performance by approaching the $R(D)$ curve.

In this appendix, we will overview the basic elements in compression techniques and some popular standards for image compression.

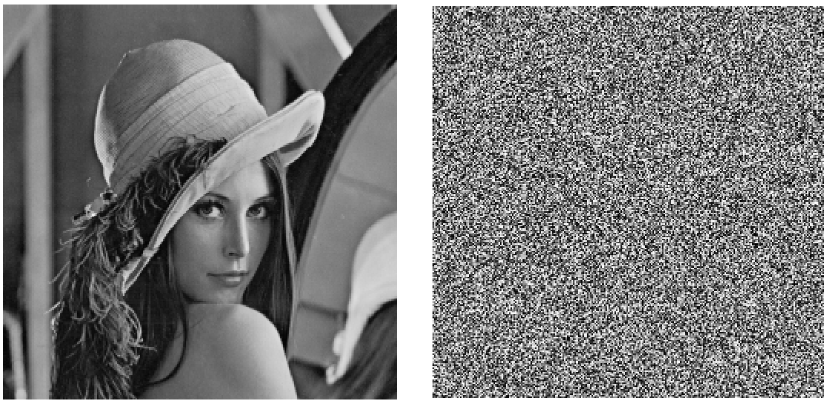
## B.1 Transform Coding

Most signals observed in nature are, in some way, compressible. This is not surprising if we consider that redundancy plays an important role in facilitating human perception. For example, it is easier and more pleasurable to read a text with repetitions, listen to songs that do not have many abrupt variations, and watch videos with trivial differences between frames. The same thing occurs with images, where adjacent pixels tent do be very similar. In Figure B.2, one can compare a redundant image (left) with a non-redundant one (right).

The existence of redundancy indicates that storing an image as a matrix in which each coefficient is the intensity of the correspondent pixel is inefficient because many will be equivalent.

The solution is to find a sparse representation, i.e., a representation in which the information is concentrated in only a few significant coefficients, the rest being zero valued. If this is accomplished, the number of coefficients that needs to be stored (or transmitted) will be largely reduced.

Transform coding [2] is the name given to data compression techniques that employ changes in signal representations to minimize redundancy. Figure B.3 intro-



(a) Image *lena*. (b) White Gaussian noise.

Figure B.2: In the image *lena*, pixels that are not in the boundary region are very similar to adjacent ones. The white noise, however, is not compressible. (Extracted from [1].)

duces the three basic operations of transform coding.

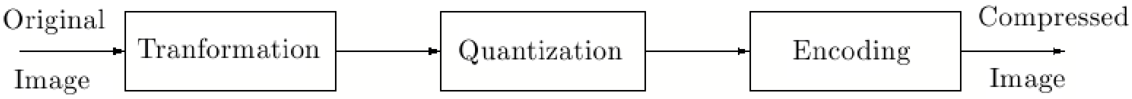


Figure B.3: Transform coding operations.

The *transformation* of the image into a set of less redundant coefficients is the first step of the compression procedure. Simultaneously, it minimizes the correlation among coefficients and maximizes the energy concentration. Nevertheless, obtaining a matrix with many zeros is not enough to reduce the number of bits required for signal reconstruction.

It is interesting to point out that pixel values usually range between 0 and 255, i.e, each pixel is represented by 8 bits. After applying a transformation, however, the coefficients can assume arbitrary floating-point values. Moreover, transformations often generate many very small coefficients instead of just zero-valued ones.

Both of these problems are resolved during the *quantization* step, which aims at representing a large range of values by a relatively small set of symbols. Though this strongly reduces the rate, it often leads to information loss.

The last step aims at mapping the symbols in the smallest stream of bits possible. This procedure, called *encoding*, takes into account the statistic characteristics of the symbols and the positions of the significant (non-zero) coefficients in the matrix.

A simple illustration of a coding scheme that uses a transformation operation is the *Differential Pulse Coded Modulation* (DPCM). The fact that, in most natural images, adjacent pixels tend to have similar values indicates that a good compression scheme would involve transmitting the difference between adjacent pixel instead of the original values.

This is the procedure of the DPCM, which uses as an estimate the value of the adjacent right pixel and transmits only the difference between the two. The advantage is that the values will now concentrate around zero and therefore more efficient quantization and coding schemes can be employed.

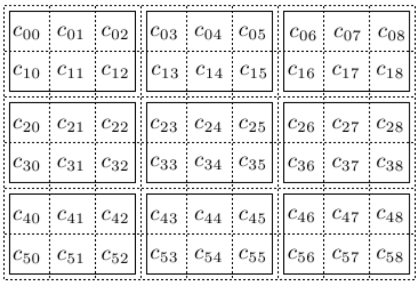
Notice that, without quantization and coding, this procedure, instead of reducing the output bit stream, enlarges it, because the pixel values which before transformation were between $\{0, 255\}$, range between $\{-255, 255\}$ after it.

In the following sections, we will study in more detail and will exemplify these three basic operations.

## B.2 Transformation

From what was just mentioned, we conclude that the goal of the transformation step is to exploit information redundancy so as to adapt the signal in order to facilitate efficient quantization and encoding.

These are usually linear transforms that are applied to a sequence of inputs. In images, we have to partition the array of pixels into blocks of size $N$ which will then be mapped to a transform sequence, as shown in figure B.4. The size of $N$ is dictated by practical considerations. While large blocks will allow a greater number of zero coefficients, transform complexity grows more than linearly with $N$ and statistical characteristics change abruptly (images are not stationary signals but we can assume stationary in a block if N is small).



| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| $c_{00}$ | $c_{01}$ | $c_{02}$ | $c_{03}$ | $c_{04}$ | $c_{05}$ | $c_{06}$ | $c_{07}$ | $c_{08}$ |
| $c_{10}$ | $c_{11}$ | $c_{12}$ | $c_{13}$ | $c_{14}$ | $c_{15}$ | $c_{16}$ | $c_{17}$ | $c_{18}$ |
| $c_{20}$ | $c_{21}$ | $c_{22}$ | $c_{23}$ | $c_{24}$ | $c_{25}$ | $c_{26}$ | $c_{27}$ | $c_{28}$ |
| $c_{30}$ | $c_{31}$ | $c_{32}$ | $c_{33}$ | $c_{34}$ | $c_{35}$ | $c_{36}$ | $c_{37}$ | $c_{38}$ |
| $c_{40}$ | $c_{41}$ | $c_{42}$ | $c_{43}$ | $c_{44}$ | $c_{45}$ | $c_{46}$ | $c_{47}$ | $c_{48}$ |
| $c_{50}$ | $c_{51}$ | $c_{52}$ | $c_{53}$ | $c_{54}$ | $c_{55}$ | $c_{56}$ | $c_{57}$ | $c_{58}$ |

$$
\begin{aligned}
v_0 &= (c_{00}, c_{01}, c_{02}, c_{10}, c_{11}, c_{12})\\
v_1 &= (c_{03}, c_{04}, c_{05}, c_{13}, c_{14}, c_{15})\\
v_2 &= (c_{06}, c_{07}, c_{08}, c_{16}, c_{17}, c_{18})\\
v_3 &= (c_{30}, c_{31}, c_{32}, c_{40}, c_{41}, c_{42})\\
v_4 &= (c_{33}, c_{34}, c_{35}, c_{43}, c_{44}, c_{45})\\
v_5 &= (c_{36}, c_{37}, c_{38}, c_{46}, c_{47}, c_{48})\\
v_6 &= (c_{50}, c_{51}, c_{52}, c_{60}, c_{61}, c_{62})\\
v_7 &= (c_{53}, c_{54}, c_{55}, c_{63}, c_{64}, c_{65})\\
v_8 &= (c_{56}, c_{57}, c_{58}, c_{66}, c_{67}, c_{68})
\end{aligned}
$$

Figure B.4: Partition of an image array into blocks of size $N = 6$ and the sequence of correspondent vectors.

Let us now analyze three very common transformations and their applications in image compression.

### B.2.1 Karhunen-Loève Transform (KLT)

KLT [16] is referred by many authors as PCA (Principal Components Analysis). In general, if we partition an image into blocks of size $N$ and then represent each block as a vector in $\mathbb{R}^N$, the correlation between the coordinates will be very large, as shown in Figure B.5.

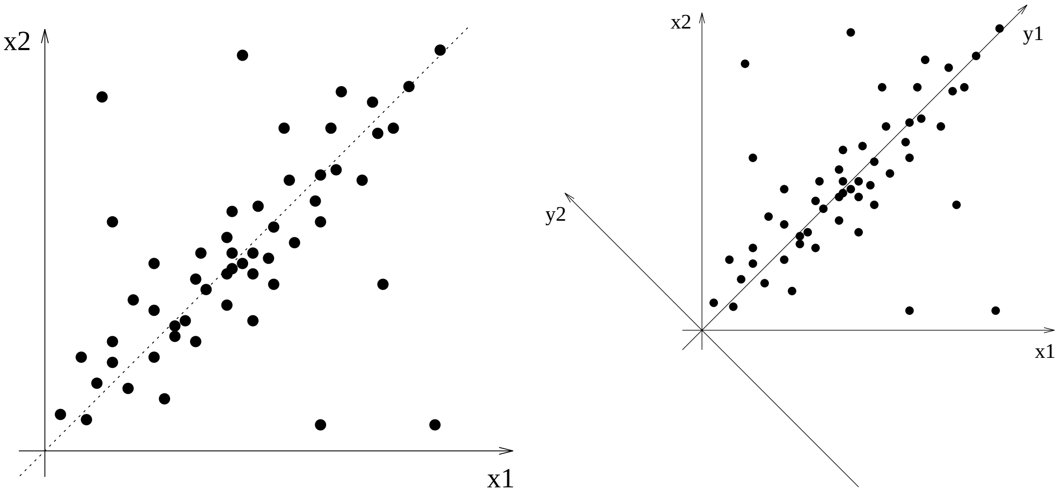
The idea of KLT is to rotate the axes in order to minimize the correlation, which can be interpreted as redundancy between coefficients, and consequently increase energy concentration.

The basis vectors of the KLT transform are given by the orthonormalized eigenvectors of its autocorrelation matrix. This indicates a drawback to this technique: it is functionally dependent on the input data.

### B.2.2 Discrete Cosine Transform (DCT)

The DCT [16] is very similar to the Fourier transform in the sense that it provides a spectral analysis of the signal. It has, however, a few properties, that make it interesting for applications in compression.

The cosine transform is very closely related to the KLT of a first-order stationary Markov sequence when the correlation parameter is close to 1 and therefore, provides



(a) (b)

Figure B.5: Each image block is represented in (a) as a vector in $\mathbb{R}^2$, and the on the KLT transform shown in (b) each vector $[a\ b]^T = ax_1 + bx_2$ will be represented by $[c\ d]^T = cy_1 + dy_2$. (Extracted from [1].)

excellent energy compaction for highly correlated data.

Moreover, it is a real transform that can be implemented by a fast algorithm and is data independent.

We represent an image in the DCT domain by a matrix where each coefficient is given by

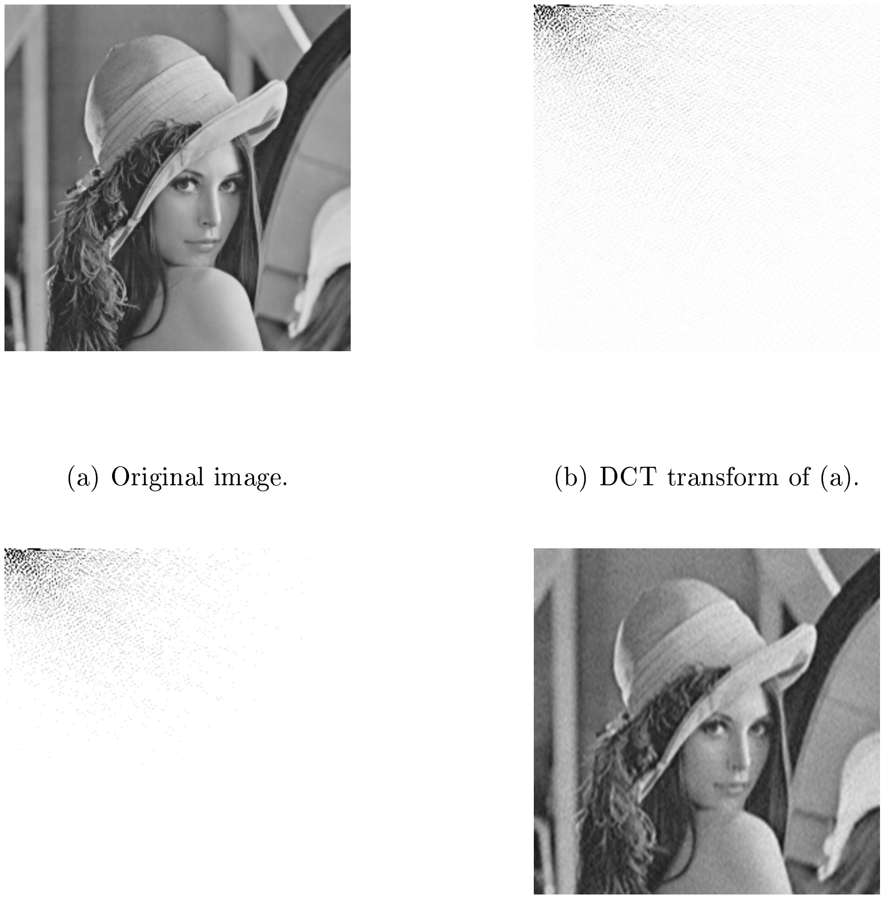
$$X_{k_1,k_2} = \alpha_1(k_1)\alpha_2(k_2) \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} x_{n_1,n_2} \cos\left[\frac{\pi}{N_1}\left(n_1 + \frac{1}{2}\right)k_1\right] \cos\left[\frac{\pi}{N_2}\left(n_2 + \frac{1}{2}\right)k_2\right]$$

where $x_{n_1,n_2}$ is the value of the pixel at $(n_1, n_2)$ and

$$\begin{cases} \alpha_i(k) = \sqrt{\frac{1}{N_i}}, & \text{if } k = 0 \\ \alpha_i(k) = \sqrt{\frac{2}{N_i}}, & \text{if } k \neq 0 \end{cases}$$

Notice that the first coefficient corresponds to the average signal level (DC value) of the signal and greater frequencies are associated with higher coefficients.

Figure B.6 illustrates the transformation applied to the image *lena*. To simplify the example, block partitioning was not used. A better result would have been achieved if we had applied the DCT individually to $N \times N$ blocks.



(a) Original image.

(b) DCT transform of (a).

(c) The most significant coefficients of (b).

(d) Image reconstructed form (c).

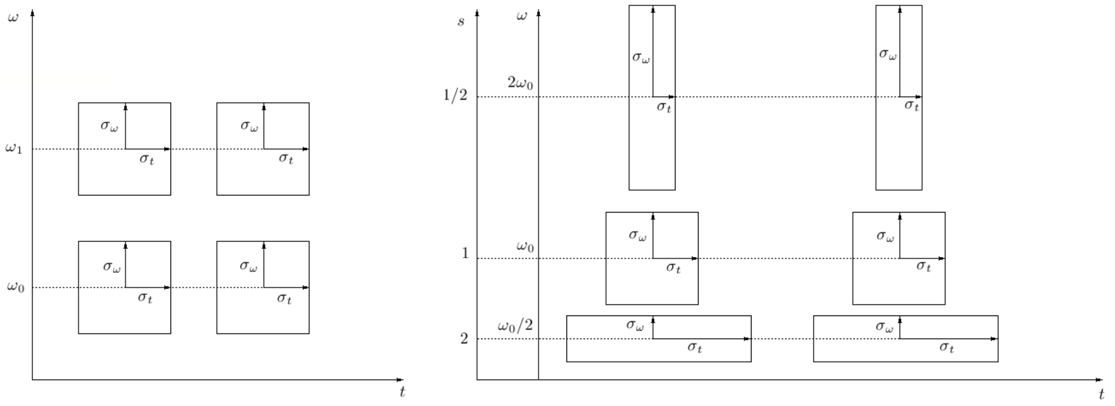
Figure B.6: Example of image reconstructed with 1 out of 10 coefficients: we set to zero the smallest values of the DCT transform and reconstruct the image by applying an inverse DCT. We observe that, since many DCT coefficients are close to zero, the distortion is rather small.

### B.2.3 Discrete Wavelet Transform (DWT)

While the time domain describes the way a signal varies in time and its Fourier transform sheds light to the frequencies distribution, the Wavelet transform can be interpreted as a way to extract information from a signal concerning both time and frequency. A first approach to achieve simultaneously both features is to apply the Fourier transform in windowed functions of the original signal $x(t)$. This is known as the *Short Term Fourier Transform* (STFT) [17], and can be defined as

$$X_F(\omega, t) = \int_{\infty}^{-\infty} x(\tau)g(\tau - t)e^{-j\omega\tau}d\tau \tag{B.1}$$

where $g(t)$ is a window function centered in zero, with variance in time $\sigma_t^2$[1], and variance in frequency $\sigma_\omega^2$.



(a) STFT

(b) Wavelet Transform

Figure B.7: Time × frequency plane for the STFT and Wavelet transform. (Extracted from [18].)

Notice from Figure B.7(a) and Equation B.1 that the information in $(\omega_0, t_0)$ mostly depends on the values of signal $x(t)$ in the intervals $[\omega_0 - \sigma_\omega, \omega_0 + \sigma_\omega]$ and $[t_0 - \sigma_t, t_0 + \sigma_t]$. The smaller $\sigma_t^2$ the better a feature can be localized in the time domain, while the smaller the $\sigma_\omega^2$, the better the frequency resolution of the STFT. However, the *uncertainty principle* states that we cannot find a window function $g(t)$ that allows both $\sigma_t^2$ and $\sigma_\omega^2$ to be arbitrarily small, i.e., it is impossible to obtain precise localization in both domains simultaneously.

Therefore, a fixed window function implies a predetermined resolution in which information is obtained. Images, however, as well as most of the natural signals, combine features of different detail levels. Therefore, a major drawback in the STFT is that the size of the window function is invariant.

[1] We calculate variance as follows

$$\sigma_t^2 = \frac{\int_{-\infty}^{\infty} t^2 g(t)dt}{\int_{-\infty}^{\infty} g(t)dt}$$

The Wavelet transform tries to solve this problem by introducing the concept of scale. A scale is closely related to the width of the window and represents a measure of the amount of detail in the signal. The Wavelet transform of a signal $x(t)$ is the decomposition of $x(t)$ on the basis of translated and scaled version of a mother function $\Phi(t)$. The mother function scaled by $s$ and translated by $t$ is described as follows:

$$\Phi_{s,t}(\tau) = \frac{1}{\sqrt{s}}\Phi(\frac{\tau - t}{s})$$

where $\frac{1}{\sqrt{s}}$ is a normalization factor.

The function $\Phi_{s,t}(\tau)$ dilates and contracts as $s$ changes, varying inversely to its Fourier transform, as shown in Figure B.8. Therefore, the interval of the signal $x(t)$ that contributes to its Wavelet transform at $(s, t)$ varies as shown in Figure B.7(b).

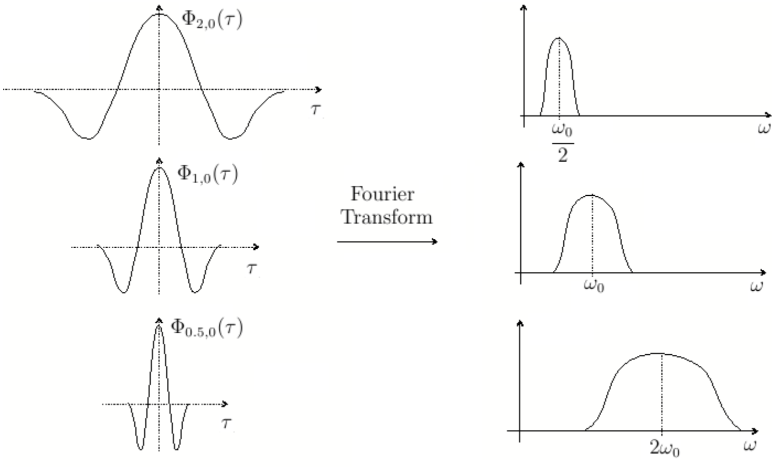


Figure B.8: Scaled wavelet functions and their Fourier transforms. (Extracted from [18].)

The values of the transformed coefficients for a given scale inform how the signal behaves at a given resolution level. In small scales, refinement signal details are explored, while in large ones, coarse details are analyzed.

The redundancy generated by mapping a one dimensional signal in a two dimensional function indicates that recovery will still be possible after discretization is done. A common partition of the time × frequency grid is shown in Figure B.9

and is known as a dyadic lattice:

$$(s, t) \in \{(2^m, n2^m t_0), n, m \in \mathbb{Z}\}$$

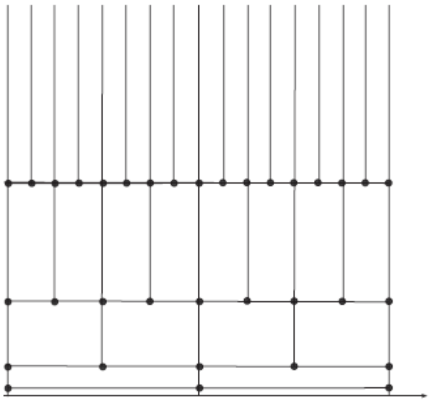


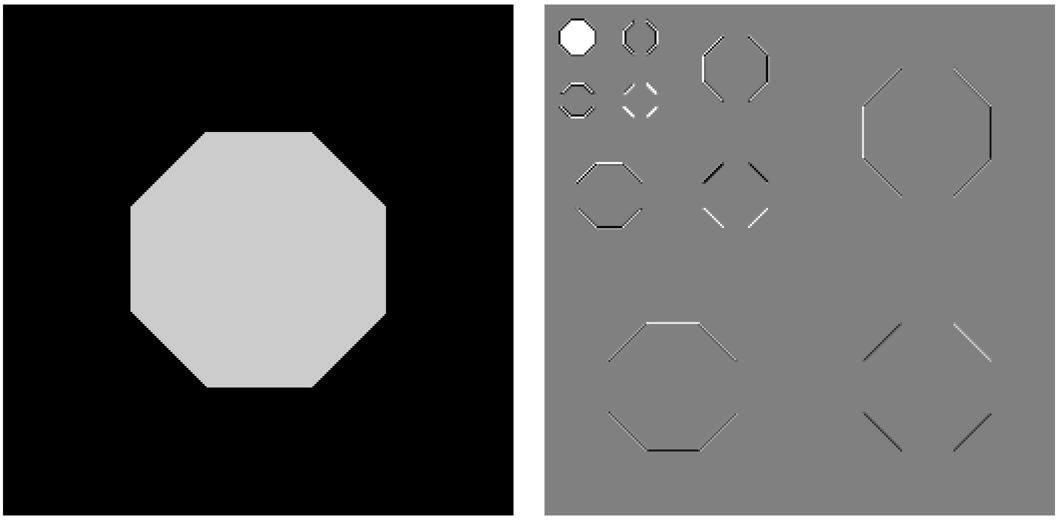
Figure B.9: The discrete grid of the DWT. (Extracted from [17].)

In terms of signal processing, a Wavelet transform is equivalent to filtering a signal in different subbands, each representing the signal information in a different resolution. This conclusion can be drawn from Figure B.8, where the scaled Wavelet function is represented in the frequency domain by band pass filters.

A common way to generate this subband decomposition is by dividing a signal into low and high pass bands and then filtering again the low pass channel in low and high pass channels. The process of dividing the resulting low pass channel is repeated until a predetermined number of stages is reached.

At each step, the low pass filtering corresponds to a smoothing of the signal and the removal of details, whereas the high pass corresponds to the differences between the scales.

In images, the DWT is applied both to rows and columns, as shown in Figure B.10. In this figure we notice that most of the coefficients are close to zero and that the horizontal, vertical and diagonal bands are closely related. These features, allied to the ability of dividing the information in detail levels, make the DWT interesting for compression applications.



(a) Original image

(b) Wavelet Transform

Figure B.10: Example of 2D Wavelet transform of three stages. In (b) the coefficients are represented on a grayscale, white corresponding to positive values, back to negative and gray to zero values. (Extracted from [19].)

## B.3 Quantization

Quantization [2] consists in representing a source output using one of a finite (and usually small) number of codewords. Since the number of codewords and the characteristics of the quantizer are closely related to the level of compression and the loss in fidelity, it is essential to bear in mind a rate-distortion criteria during this procedure.

Here we present two kinds of quantizers that differ in terms of the set of inputs and outputs, that can be either scalars or vectors.

### B.3.1 Scalar Quantization

Scalar quantization consists in dividing the scalar input range into intervals and assigning for each one a codeword and an output value.

Figure B.11 is an example of a linear quantizer, where all intervals have the same size, called *quantization step*.

In many applications it is not efficient to establish constant distances between

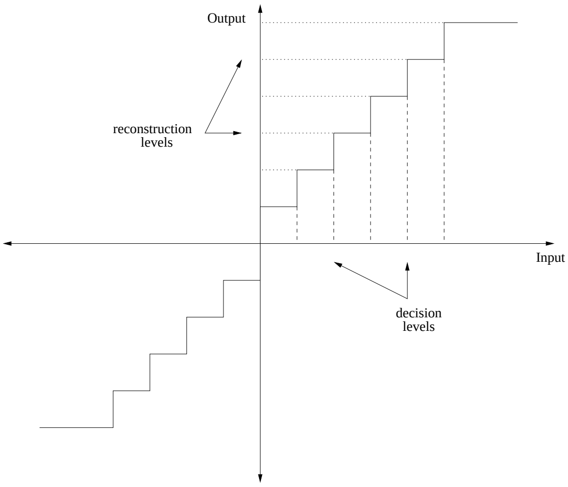


Figure B.11: Linear quantizer input-output map.

decision and reconstruction levels. If this does not happen the quantization is called non-linear. In most image compression standards, however, the latter is not used because entropy coding combined with linear quantization provides a very similar performance and is less complex to implement.

### B.3.2 Vector Quantization

From what has been studied up until now and from basic results in information theory, it is clear that encoding a sequence of outputs instead of individual samples separately is more efficient according to a rate-distortion criteria.

In this case, instead of quantizing each image pixel, we divide images into blocks of size $N$ and represent each one as a vector in $\mathbb{R}^N$. The output of the quantizer is a finite set of vectors called *codebook* and each block of the source output is associated to the closest vector in the codebook, usually by applying the Euclidean norm.

The process of finding the optimal codebook of size $k$ for a given source set of vectors $\mathcal{S}$ involves choosing the $k$ vectors of the codebook, and the $k$ *quantization cells* - each quantization cell corresponds to the subset of $\mathcal{S}$ that is associated to the $k^{th}$ code-vector. This procedure is not analytical because it involves two related considerations:

- Given the quantization cells, the best codebook is constructed by extracting the centers of each cell.

- Given the codebook, the best quantization cells are found by assigning each element in $\mathcal{S}$ to its closest vector in the codebook.

Hence, there are many algorithms for finding the best codebook given certain input data. Here we will describe one of the simplest, yet very popular, referred to as LBG:

1. Initialize the codebook by selecting k vectors at random.
2. Specify the quantization cells, i.e., assign to each source output the closest vector in the codebook.
3. Reset the codebook by selecting the centers of each quantization cell.
4. Return to step 2 unless a finalization condition is reached.

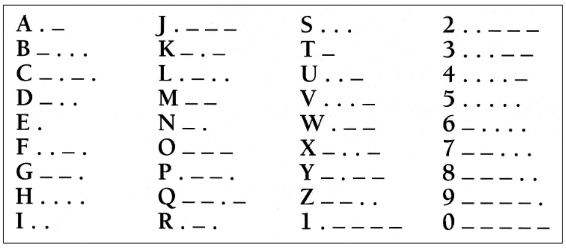
## B.4 Encoding

We refer to coding [20] as the process of assigning binary representations to the output of a source, here referred to as alphabet. For example, the ASSCII code uses 8 bits and each of the $2^8$ possible combinations is associated to one of the 256 letters or punctuation marks. This is a so called *fixed-length code* because all symbols are represented by the same number of bits.

To minimize the average number of bits per symbol, we should use fewer bits to represent symbols that occur more often. This is done in the Morse code, as illustrated in Figure B.12. Note that the smallest codeword is associated to the letter E, which is the most used in the English language.

We measure efficiency in terms of rate minimization by comparing the average symbol length with the alphabet's entropy, which is a measurement of the average information per source symbol.

Let $\mathcal{S} = \{s_1, \dots s_K\}$ be a given alphabet where each symbol has the probability of occurrence $p_k = P(\mathcal{S} = s_k)$. The entropy is given by:



| | | | |
|---|---|---|---|
| A . _ | J . _ _ _ | S . . . | 2 . . _ _ _ |
| B _ . . . | K _ . _ | T _ | 3 . . . _ _ |
| C _ . _ . | L . _ . . | U . . _ | 4 . . . . _ |
| D _ . . | M _ _ | V . . . _ | 5 . . . . . |
| E . | N _ . | W . _ _ | 6 _ . . . . |
| F . . _ . | O _ _ _ | X _ . . _ | 7 _ _ . . . |
| G _ _ . | P . _ _ . | Y _ . _ _ | 8 _ _ _ . . |
| H . . . . | Q _ _ . _ | Z _ _ . . | 9 _ _ _ _ . |
| I . . | R . _ . | 1 . _ _ _ _ | 0 _ _ _ _ _ |

Figure B.12: Morse code.

$$H(\mathcal{S}) = \sum_{k=1}^{K} p_k \cdot \log\left(\frac{1}{p_k}\right) \tag{B.2}$$

and the average code length by:

$$\bar{L} = \sum_{k=1}^{K} p_k \cdot l_k$$

where $l_k$ is the size of the codeword associated to the symbol $s_k$.

In this case, coding efficiency is measured by:

$$\eta = \frac{H(\mathcal{S})}{\bar{L}}$$

The Shannon Theorem guarantees $\bar{L} \geq H(\mathcal{S})$ and therefore the optimal code occurs when $\eta = 1$.

Along with minimizing rate, efficient codes must be uniquely decodable, i.e., there must be no ambiguity between codewords. It is also desirable that the decoding be instantaneous, which means that the decoder knows the moment a code is complete without having to wait until the beginning of the next codeword.

Now we will outline two coding procedures that are often employed in image compression standards.

### B.4.1 Huffman Code

David Huffman developed an instantaneous code where the average symbol length is very close to the entropy. It is based on two observations:

- Symbols with greater probability of occurrence should have smaller codewords.

- The two symbols that occur least frequently should have the same length.

We will demonstrate this coding procedure by an example. Let $\mathcal{S} = \{s_1, s_2, s_3, s_4\}$ be an alphabet where the probability of occurrence of each symbol is respectively $\{0.5, 0.25, 0.125, 0.125\}$.

The symbols are arranged in order of decreasing probability and the last two symbols are combined interactively until only one symbol is left. Figure B.13 illustrates this procedure and the decision tree generated by the coding strategy.

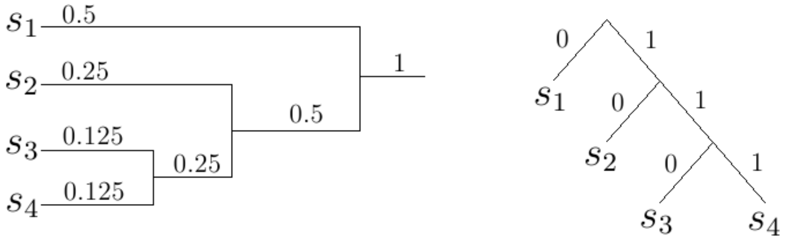


Figure B.13: Huffman code.

Table B.1: Associated codewords generated by the Huffman coding.

| **Symbol** | **Codeword** |
|---|---|
| $s_1$ | 0 |
| $s_2$ | 10 |
| $s_2$ | 110 |
| $s_2$ | 111 |

Table B.1 displays the codewords associated to each symbol. Notice that in this case, since the distribution of probabilities is dyadic, the code is optimal, i.e., $\eta = 1$.

### B.4.2 Arithmetic Code

Though very successful in many circumstances, the Huffman code becomes inefficient when a single symbol has a very large probability of occurrence. This is often the case in small alphabets, where the obligation of using an integer number of bits to represent each symbol, limits the reduction of the average code length.

In this case, a better performance would be achieved by blocking groups of symbols together and generating codes capable of characterizing entire sequences of symbols by a unique identifier. This is the proposition of the arithmetic code, which maps each sequence into the unit interval $[0, 1)$. We will illustrate the encoding procedure with an example.

Let $\mathcal{S} = \{s_1, s_2, s_3\}$ be a given alphabet where each symbol has the probability of occurrence $p_1 = 0.5$, $p_2 = 0.2$, $p_3 = 0.3$. The first step consists in dividing the unit interval into regions that are associated with each symbol. The size of each region is, of course, directly related to the symbol probability, since larger regions will require a smaller number of decimal figures to be represented.

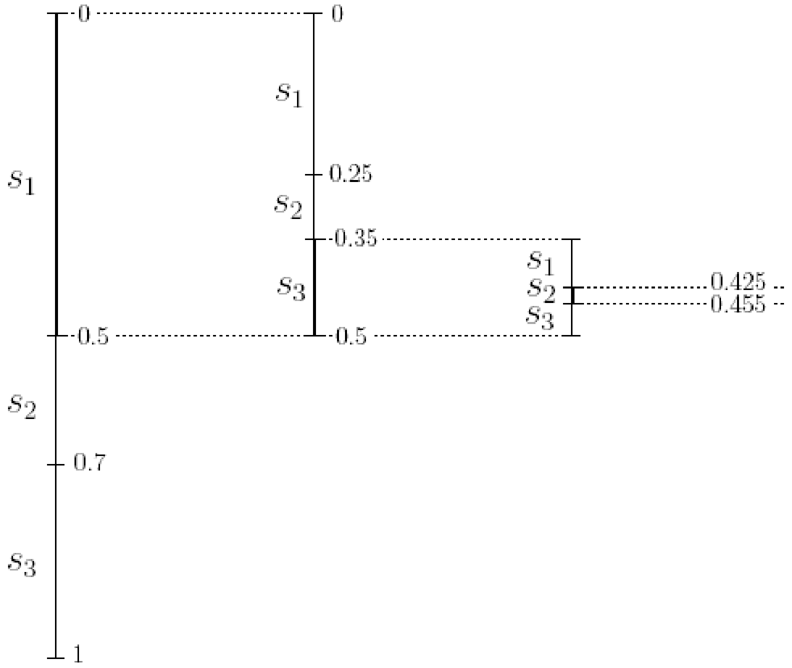


Figure B.14: Example o arithmetic encoding.

If the first symbol to be encoded is $s_1$, then the code will be a number in $[0, 0.5)$ and this interval will be divided according to the alphabet's probability distribution. This process is repeated iteratively as shown in figure B.14, which considers the sequence $(s_1, s_3, s_2)$, and the transmitted code is a number between 0.425 and 0.455, for example the mean 0.44. The decoder procedure is also done iteratively dividing the interval and finding the associated symbols.

There are, however, two problems associated with arithmetic coding:

- There is no information provided as to when the decoding should stop.

- The binary representation of a real value with infinite precision can be infinitely long.

The first problem can be solved either by informing the decoder the size of the sequence or by associating a region of the unit interval with an *end-of-transmission* symbol. Figure B.15 illustrates the EOT symbol, that brings the decoding procedure to a stop as soon as it is detected.

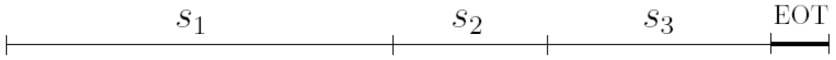


Figure B.15: The *end-of-transmission* symbol.

There are several approaches to solve the second problem. The simplest one would be to encode each decimal symbol at a time, i.e., when we reach an interval small enough to make the $n^{th}$ digit stop varying, it is transmitted.

# B.5 Standards

In this section we will illustrate image compression by describing two very important standards: JPEG and JPEG2000.

## B.5.1 JPEG

The JPEG [2] standard uses a very popular compression technique that involves DCT transform, followed by scalar quantization and Huffman coding.

This procedure starts by dividing the image into blocks of size $8 \times 8$ which are transformed by a forward DCT. This transformation isolates the important image components in the upper left portion of the matrix.

The calculated coefficients are quantized by uniform scalar quantization, where the step size varies, increasing as we move from DC coefficients to higher-order coefficients. The variation of the step size is related to the perception of the human visual system to errors in different spatial frequencies. Since the human eye is less sensitive to higher spatial frequencies, we can accept greater quantization errors for the coefficients that represent them. The following matrix shows the weight of each quantization step, i.e., the quantization step of the coefficient $c_{ij}$ is $q_{\text{global}} Q_{ij}$, where $q_{\text{global}}$ is a parameter associated with the compression rate.

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

The DC values are encoded separately from the AC ones because they vary little between adjacent blocks and, thus, it is interesting to encode the difference between neighbors. Therefore the DC values, i.e., the fist coefficient of each transformed block, are coded using DPCM followed by a Huffman entropy encoder.

To understand the coding of the AC coefficients it is important to analyze some properties of the matrix that stores the quantized coefficients of a typical DCT-transformed image block:

$$C = \begin{bmatrix} 42 & 26 & 10 & 0 & 0 & 0 & 0 & 0 \\ -3 & -2 & 0 & 2 & -1 & 0 & 0 & 0 \\ -21 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Notice that, not only is the matrix sparse, but also most of the nonzero coefficients are located on its upper-left corner. These characteristics suggest a scanning in a diagonal zigzag pattern, as shown in Figure B.16.

The JPEG standard uses *run-length* encoding; i.e., each nonzero value that is scanned in the above fashion is stored as a sequence of pairs (run, length); the first indicating the number of preceding zeros and the second the values of the component. These pairs are then encoded using a Huffman code.

A drawback of dividing the image into blocks is that coding artifacts may be generated at block edges. This effect, called *blockiness*, is illustrated in Figure B.17
.

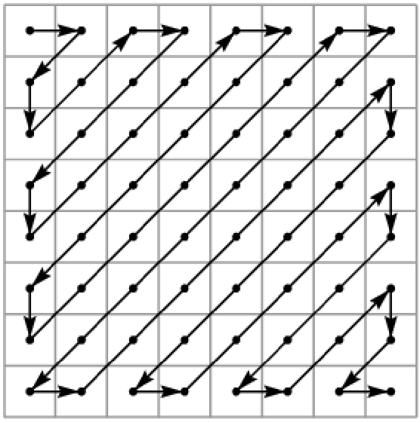


Figure B.16: The zigzag scanning pattern.

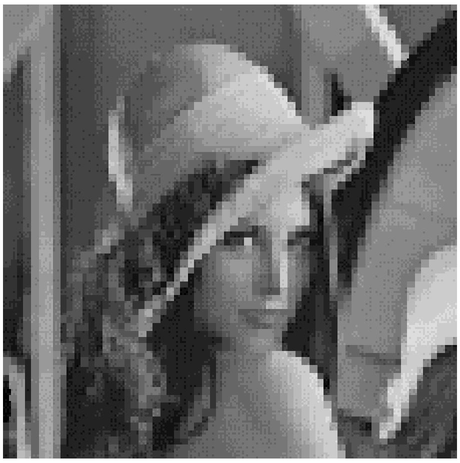


Figure B.17: Example of the blocking effect generated by a JPEG compression with very high rate.

### B.5.2 JPEG2000

JPEG2000 [21] gains up to about 20% compression performance for medium compression rates in comparison to the first JPEG standard, but has, however, notably higher computational and memory demands. It involves a Wavelet transform followed by scallar quantization and arithmetic coding.

The Wavelet transform is applied to the tilled image, where the size of the tile can vary widely, being possible to consider the whole image as one single tile. This is important because small tiles can generate blocking effects, as in the JPEG standard.

The Wavelet coefficients are quantized by a uniform scalar quantizer with step size varying between subbands considering the human visual sensibility to different scaled informations. Each bit plane[2] of the quantized coefficients is then encoded using a process called *Embedded Block Coding with Optimal Truncation* (EBCOT).

As studied in section B.2.3 Wavelet transform divide the image into subbands that represent approximation scales. Notice, however, that some Wavelet coefficients in different subbands represent the same spacial location in the image. In Figure B.10(b), it is noteworthy that the vertical subbands approximate scaled versions of each other, the same being true for horizontal and diagonal bands. This means that there exists a relation between the Wavelets coefficients illustrated in Figure B.18.

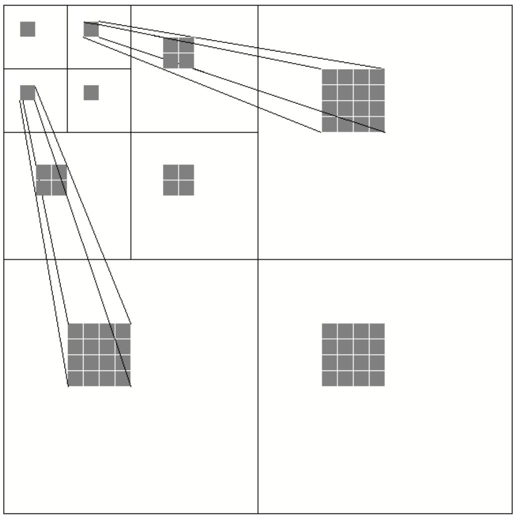


Figure B.18: Related Wavelet coefficients.

Many algorithms, as the EZW and the SPHT codes, exploit the similarity among bands of the same orientation in order to reduce the size of the encoded image. JPEG2000 coding, however, does not exploit inter-subband redundancies. Instead, the EBCOT algorithm partitions each subband into small rectangular blocks called *codeblocks* and encodes each one independently.

Though there is an efficiency loss for not exploiting the correlation between

[2] A bit plane of a digital discrete signal is a set of bits having the same position in the respective binary numbers. For example, for 8-bit data representation there are 8 bitplanes: the first one contains the set of the most significant bits and the $8^{th}$ contains the least significant bits.

subbands, this is compensated for because this method produces bit streams that are SNR and resolution scalable. For each codeblock a separate highly scalable bit stream is generated and may be independently truncated to any of a collection of different lengths.

The bits generated by the EBCOT algorithm are then encoded using an arithmetic code.

## B.6 Classification of Compression Techniques

Many authors distinguish compression techniques as lossless or lossy, the former referring to invertible representations and the latter to representations in which some of the information is lost. Since quantization involves distortion effects, it is clear that we have focused our study in lossy compression schemes. In terms of the rate-distortion criteria, lossless compression would occur when the function $R(D)$ crosses the y-axis, i.e., when the distortion is zero.

For images we are usually interested in lossy techniques because they allow lower rates and the human visual system is not sensitive to small distortions. An exception to this rule would be when dealing with medical images, where the slightest error can result in a wrong diagnosis.

Another form of classification is linear and non-linear compression. To illustrate the difference between the two we will discuss the JPEG standard for image compression.

As shown in Section B.5.1, the DCT transform results in a sparse matrix where the significant coefficients are concentrated in the upper-left corner and an encoding procedure called *run-length* coding makes use of these properties in order to reduce the size of the output stream of bits. Another approach would be to consider that all components in the lower-right corner are small, and so store only N values that belong to the region of the matrix that is usually significant, as shown in Figure B.19.

This would not be as efficient as the run-length coding because some high-

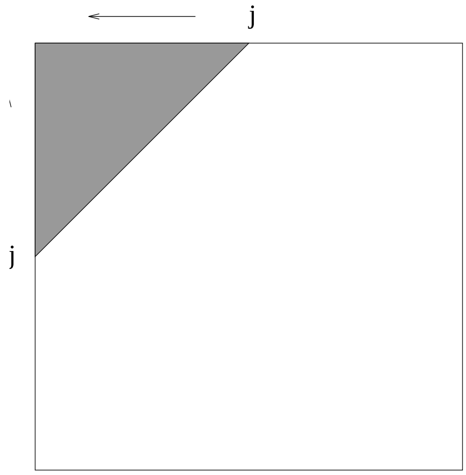
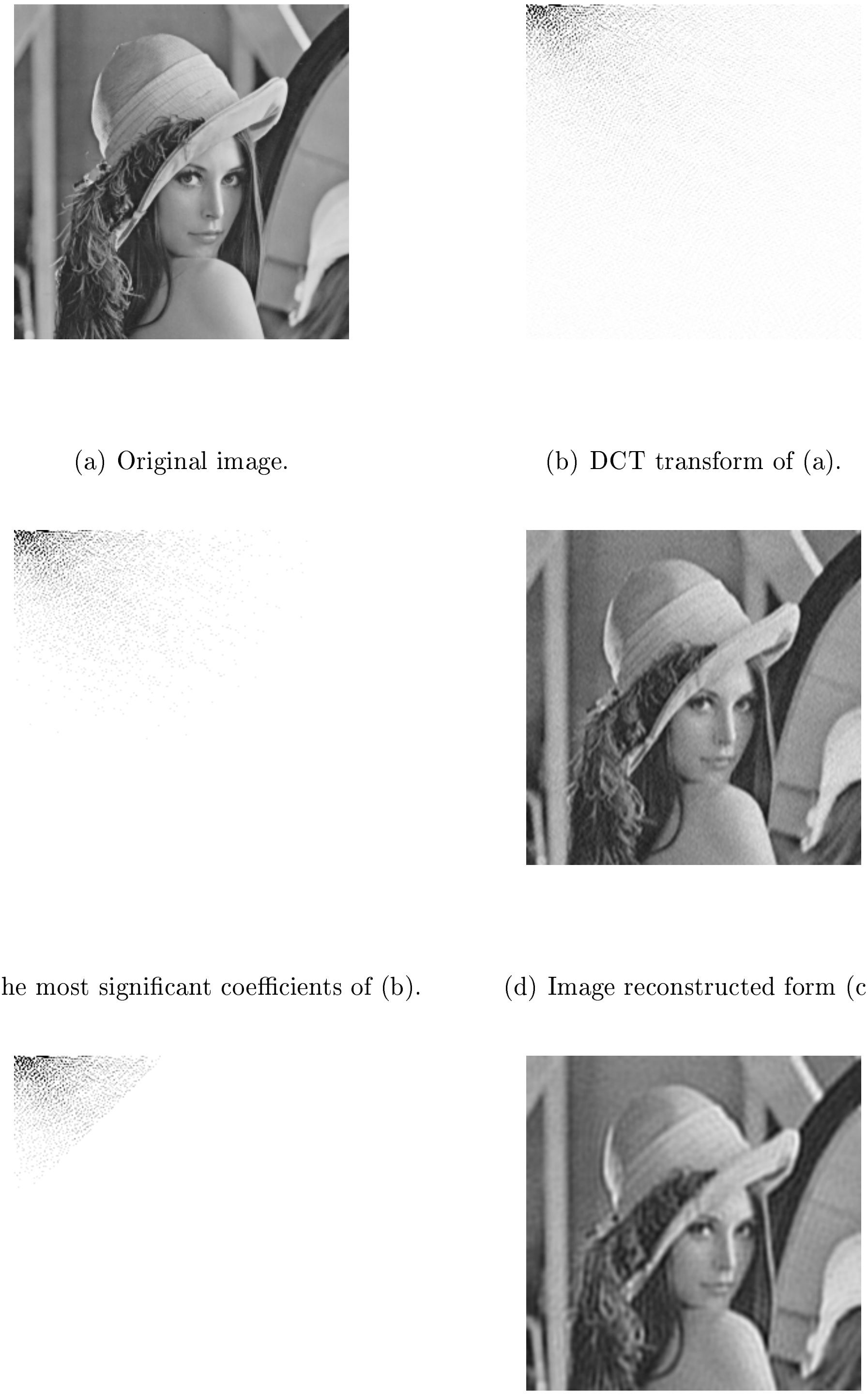


Figure B.19: Example of region of matrix that would be selected as being significant in a linear compression scheme. (Extracted from [1].)

frequency information might be lost and zero-valued coefficients would be unnecessarily stored. However this approach is interesting because the compression technique does not depend on the image, i.e., we do not need to know *a priori* where the significant coefficients are before we begin encoding. This is what is referred to in literature as linear compression. In other words, if $A$ and $B$ are images and $\hat{A}$ and $\hat{B}$ are their compressed forms, the compression of $A+B$ will result in $\hat{A}+\hat{B}$. In non-linear compression, however, the location of the significant coefficients must be known before the reconstruction can be accomplished and, therefore, the linearity does not hold.

In Figure B.20 we compare the reconstruction of image *lena* with 1 out of 10 coefficients using non-linear and linear DCT compression and are able o conclude that the latter scheme is much less efficient. In B.20(c) we set to zero the smallest values of the DCT transform and in B.20(e) we set to zero the DCT coefficients that are not on the upper-left corner of the transformed matrix. Images B.20(d) and B.20(f) are reconstructed by applying an inverse DCT to B.20(c) and B.20(e), respectively.



(a) Original image.

(b) DCT transform of (a).

(c) The most significant coefficients of (b).

(d) Image reconstructed form (c).

(e) Coefficients of (b) on the upper-left corner.

(f) Image reconstructed form (e).

Figure B.20: Example of image reconstructed with 1 out of 10 coefficients

# Appendix C

# Signal Representations

Representation is a key aspect in signal processing. It refers to describing a signal completely and unambiguously as a sequence of enumerable coefficients. The importance of this procedure can be associated with the continuous nature of existing signals, which has to be overcome before digital processing.

Discretization, however, is not the only benefit we are searching for. Good signal representations can enable a series of procedures as analysis, noise filtering and compression. The idea behind this is that depending on how we describe a signal some of its aspects can be highlighted, i.e., we can distribute the information of interest between specific components and therefore ease access to them [22].

In this appendix we will overview different ways of representing signals and analyze their basic characteristics and how signals can be reconstructed from them.

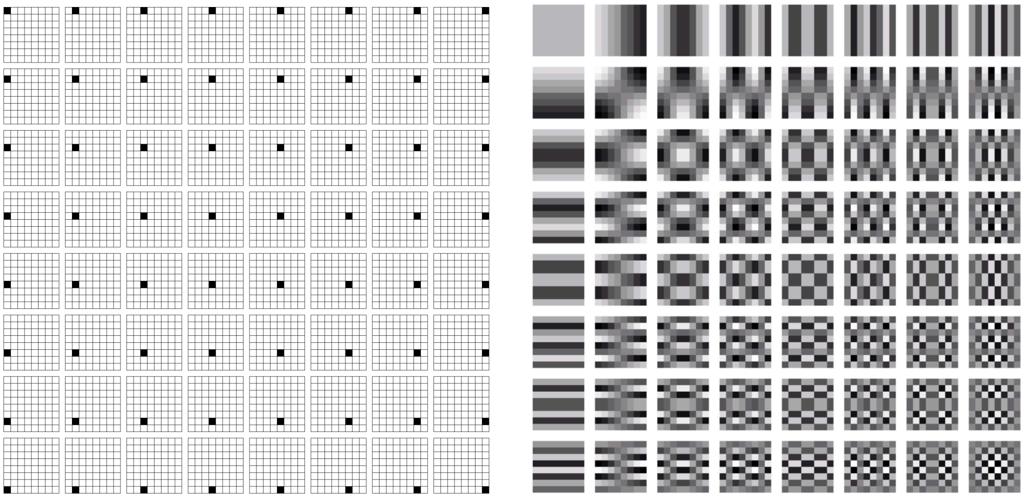
## C.1 Parallel with Image Compression

In the former appendix, we discussed transform coding as a method for compressing images by representing the same information in a smaller number of coefficients. It is interesting to point out, however, that when we exploit redundancy to map the image data to less correlated coefficients, we are actually choosing a new way to represent the signal.

We can interpret an $n \times n$ image block as a vector in $\mathbb{R}^N$, where $N = n^2$. In the bit-map representation, each of the $N$ canonical basis vectors would corespond to the information of a single pixel.

Since each orthonormal basis is a rotation of each other, the DCT transform is, therefore, no more than the rotation of this basis. Notice that the DCT expands the original image in sequence of cosines, i.e., the transformation is actually the projection in a new orthonormal basis.

The bit-map (canonical) basis is equivalent to *Dirac* functions in a two dimensional space, as shown in Figure C.1(a), while the DCT basis is illustraded in Figure C.1(b).



(a) $8 \times 8$ bit map basis.

(b) $8 \times 8$ DCT basis.

Figure C.1: Waveforms that compose the bit map and DCT bases.

Notice, however, that the DCT preserves many properties such as invertibility and orthogonality, which cannot be guaranteed for arbitrary representations. In the next section, we will, therefore, define such representations in a more abstract and generalized manner.

## C.2 Signal Decompositions

We define a signal representation [17] by a function $R : \mathbf{H} \rightarrow \mathcal{S}$ that maps a Hilbert space[1] $\mathbf{H}$ into a space of sequences. For a given signal, $x \in \mathbf{H}$, its representation $R(x)$ is a sequence:

$$R(x) = (s_1, s_2, s_3...) \in \mathcal{S}$$

where $s_n$ is a pair $(\alpha_n, g_{\gamma_n})$, the first representing a coefficient and the second a waveform.

Associated with $R$ is a set of functions $\mathcal{D} = (g_\lambda)_{\lambda \in \Gamma}$ called *dictionary.* Notice that the dictionary may be uncountable, however, the $(g_{\gamma_n})_{n \in \mathbb{Z}}$ used in the representation of a particular signal $X$ consists of a countable subset.

In some cases, the function $R$ is invertible and the signal $x$ will be perfectly reconstructed from its representation $R(x)$. We then say that the representation is *exact* and the original signal is reconstructed by the linear combination

$$x = \sum_{n \in \mathbb{Z}} \alpha_n g_{\gamma_n}$$

Nevertheless, when the representation is not exact, we make use of techniques to approximate the reconstruction of $x$.

The dimension $N$ of the signal space $\mathbf{H}$ is associated with the number of elements of the dictionary that are needed to span the space. A good representation scheme requires the use of a *complete* dictionary, i.e., any function in $\mathbf{H}$ can be expanded by a combination of the waveforms $(g_\lambda)_{\lambda \in \Gamma}$. It is noteworthy, however, that the size of the dictionary may be larger than N. In this case, we say that the dictionary is redundant because there is more than one way to represent the same signal. It is important to point out that, is some cases, we deal with infinite dimensions.

The key point in signal decompositions is thus to obtain the sequence of dictionary waveforms $(g_{\lambda_n})_{n \in \mathbb{Z}}$ and their corresponding coefficients $(\alpha_n)_{n \in \mathbb{Z}}$. There are

[1] A Hilbert space is an inner product space which, as a metric space, is complete, i.e., an abstract vector space in which distances and angles can be measured and which is complete, meaning that if a sequence of vectors approaches a limit, then that limit is guaranteed to be in the space as well.

many methods that do so, exploiting signal properties, as mentioned earlier. We will now distinguish between two representation models: basis and frames.

### C.2.1 Basis

A basis [23] is a set of linearly independent elements $(\phi_\lambda)_{\lambda \in \Gamma}$ that span the Hilbert space $\mathbf{H}$. By linear independence we mean that no function can be expressed as a linear combination of the others - this implies that the set is minimal.

#### Orthogonal Basis

We define an orthonormal basis as collection of functions $\{\phi_\lambda; \lambda \in \Gamma\}$ that are complete in the sense that they span $\mathbf{H}$ and satisfy:

$$\int_{-\infty}^{\infty} \phi_i(t)\bar{\phi}_j(t)dt = \delta(i-j), \quad \forall i, j \in \Gamma$$

where $\bar{\phi} = \mathrm{Re}\{\phi\} - j\mathrm{Im}\{\phi\}$ is de complex conjugate.

In this case, the representation is exact and the reconstruction is given by

$$x = \sum_{\lambda \in \Gamma} \langle x, \phi_\lambda \rangle \phi_\lambda$$

where the inner product $\langle x, \phi_\lambda \rangle = \int_{-\infty}^{\infty} x(t)\bar{\phi}_\lambda(t)dt$ is interpreted as the projection of the signal of interest in the base function $\phi_\lambda$.

### C.2.2 Frames

Frames [23] are a generalization of the concept of basis in a linear space. While a set of vectors forms a basis in $\mathbb{R}^M$ if they span $\mathbb{R}^M$ and are linearly independent, a set of $N \geq M$ vectors form a frame if they span $\mathbb{R}^M$.

More formally, a frame is a family of vectors $(\phi_\lambda)_{\lambda \in \Gamma}$ that characterizes any signal $x$ in a Hilbert space $\mathbf{H}$ from its inner product $\{\langle x, \phi_\lambda \rangle\}_{\lambda \in \Gamma}$, where the index set $\Gamma$ might be finite or infinite.

Frame Theory, developed by Duffin and Schaeffer, sets a condition for the frame to define a complete and stable signal representation:

**Definition 2.** *The sequence $(\phi_\lambda)_{\lambda\in\Gamma}$ is a frame of $\mathbf{H}$ if there exist two constants $A > 0$ and $B > 0$ such that for any $x \in \mathbf{H}$*

$$A\|x\|^2 \leq \sum_{\lambda\in\Gamma} |\langle x, \phi_\lambda\rangle|^2 \leq B\|x\|^2$$

*When $A = B$ the frame is said to be tight.*

It is noteworthy that a frame representation may be redundant, and, considering $\|\phi_\lambda\| = 1, \forall\lambda \in \Gamma$, this redundancy can be measured by the frame bounds A and B. The following example will be used to illustrate frame redundancy:

**Example 1.** *Let $(e_1, e_2)$ be an orthonormal basis of a two-dimensional plane $\mathbf{H}$. The three vectors:*

$$\phi_1 = e_1 \ , \ \phi_2 = -\frac{e_1}{2} + \frac{\sqrt{3}}{2}e_2 \ , \ \phi_3 = -\frac{e_1}{2} - \frac{\sqrt{3}}{2}e_2$$

*have equal angles of $\frac{2\pi}{3}$ between any two vectors. For any $x \in \mathbf{H}$*

$$\begin{aligned}
&\sum_{n\in\Gamma} |\langle x, \phi_n\rangle|^2 \\
&= |\langle x, e_1\rangle|^2 + |-\frac{1}{2}\langle x, e_1\rangle + \frac{\sqrt{3}}{2}\langle x, e_2\rangle|^2 + |-\frac{1}{2}\langle x, e_1\rangle - \frac{\sqrt{3}}{2}\langle x, e_2\rangle|^2 \\
&= \frac{3}{2}|\langle x, e_1\rangle + \langle x, e_2\rangle|^2 \\
&= \frac{3}{2}\|x\|^2
\end{aligned}$$

*These three vectors thus define a tight frame with $A = B = \frac{3}{2}$. The frame bound $\frac{3}{2}$ gives the redundancy ratio, i.e., three vectors in a two-dimensional space.*

## C.3 Uniform Point Sampling

In this section we will introduce the simplest method for representing a function and analyze some of its characteristics.

Point sampling discretizes a signal $x(t)$ by taking a partition $t_1 < t_2 < \cdots < t_N$ of the domain interval $I$. The subsequent representation is given by the vector:

$$x_n = (x(t_1), x(t_2), \ldots, x(t_N)) \in \mathbb{R}^N$$

This way, the space of real functions defined on the interval $I$ is represented by the Euclidean space $\mathbb{R}^N$. Point sampling is called uniform if $t_n = nt_s$, $\forall n$.

What remains to be investigated is if uniform point sampling is an exact representation and how can the original function $x$ be recovered from $x_n$.

The Shannon theorem guarantees that a band-limited signal can be perfectly reconstructed if the sampling rate is $1/(2\omega_0)$ seconds, where $\omega_0$ is the highest frequency in the original signal. We will not demonstrate this theorem here, but we will try to convince the reader with the following observations. Additional material regarding this theorem can be found in [24].

It is intuitive that sampling a signal in the time domain is equivalent to multiplying it by a *Dirac* comb. The Fourier transform of a *Dirac* comb is also a *Dirac* comb and therefore, in the frequency domain, the band-limited spectrum of the signal is being convolved by a *Dirac* comb, see Figure C.2.

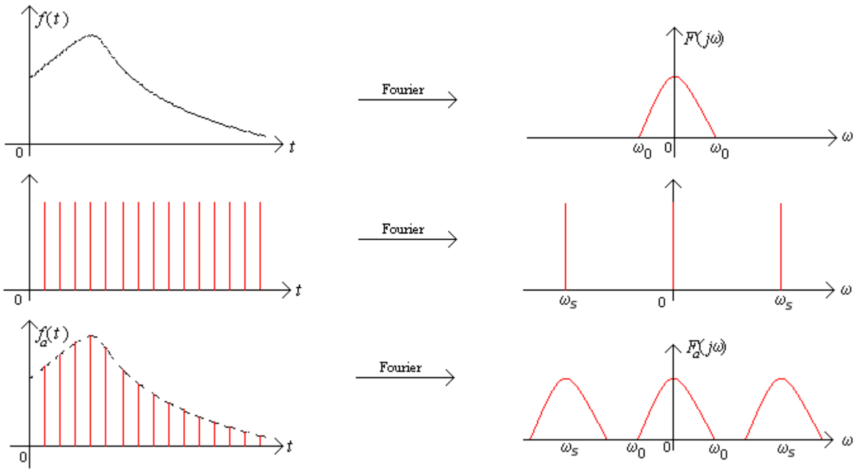


Figure C.2: Sampling in time and the consequences in the frequency domain.

By observing these pictures it is easy to see that if the sampling rate $\omega_s$ is greater than $2\omega_0$, then the signal in the frequency domain can be recovered by an ideal low pass filter, as shown in Figure C.3.

Since the Fourier transform of the Gate function is a *sinc* function, the reconstruction of the signal in the time domain is no more than an interpolation of the sampled vector by *sinc* functions.

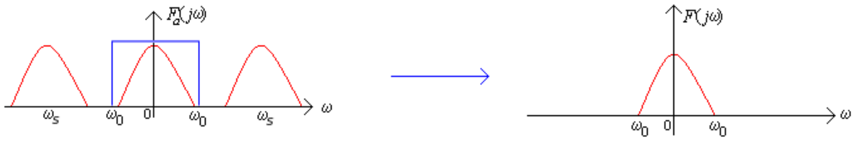


Figure C.3: Extracting the repeated spectrums.

On the other hand, if this limit of $2\omega_0$ , called the *Nyquist rate*, is not respected, then repeated spectrums will overlap and it will be impossible to recover the signal by a low pass filtering. This phenomenon is called *aliasing*, and is illustrated in Figure C.4.

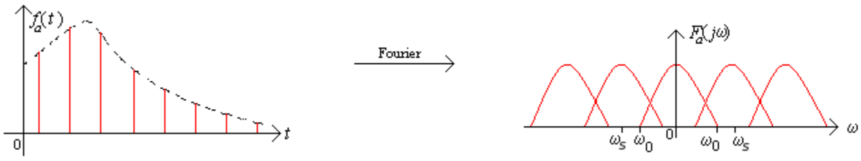


Figure C.4: Undersampling in time and the consequences in the frequency domain.

Notice that point sampling involves representing a signal as sequence of values

$$R(x) = (\alpha_n)_{n\in\mathbb{Z}}$$

where $\alpha_n$ is the projection of he signal on a delayed *Dirac*

$$\alpha_n = \langle x, \delta(t - nt_s)\rangle = \int_{-\infty}^{\infty} x\delta(t - nt_s) = x(nt_s).$$

This representation is an invertible function, once the original signal can be reconstructed by an interpolation of *sinc* functions. The exact reconstruction is then given by

$$x = \sum_{n\in\mathbb{Z}} \alpha_n h(t - nt_s)$$

where $h = \operatorname{sinc}\left(\frac{t}{t_s}\right)$ is a scaled *sinc* function.

This is a very interesting example, because the projection waveforms used for representation are different from the reconstruction waveforms (dictionary).

### C.3.1 Oversampling

If the sampling rate $\omega_s$ is greater than $2\omega_0$, we observe information redundancy, i.e., the number of samples is larger than it has to be to enable reconstruction of the signals. This can be usefull for many applications because it minimizes noise errors and allows the use of less complex anti-aliasing filters.

In this case, however, the scaled *sinc* functions that can be used to reconstruct this signal are not necessarily orthogonal. Note that

$$\langle h(t), h(t - nt_s)\rangle = \langle H(j\omega), H(j\omega)e^{-j\omega t_s}\rangle$$

where $H(j\omega)$ is a Gate function of badwidth $2/t_0$, $t_0 = 1/\omega_0$, and $t_s = 1/\omega_s$. Therefore, if $t_s = t_0/2$, then $\langle H(j\omega), H(j\omega)e^{-j\omega t_s}\rangle = 0$ and the basis is orthogonal.

However, when we oversample, this does not occur. Actually, the set $(h(t - nt_s))_{n\in\mathbb{Z}}$ becomes complete and redundant. In terms of what has been just described, this set is a frame.

### C.3.2 Undersampling

In many applications, however, the signal of interest is not band limited or it is necessary to sample in a rate smaller than the Nyquist limit. In this case, uniform sampling will undoubtedly produce aliasing.

In signal processing this problem is usually solved by applying an anti-aliasing filter. Since the periodic spectrum will overlap, to minimize the distortion effect, frequencies higher than $\omega_s$ are eliminated before sampling starts. This is accomplished by a low-pass filter known as the *anti-aliasing filter*. Figure C.5 illustrates this procedure.

Let us now analyze this problem using the concepts of representation and reconstruction. There are two problems with undersampling. The first is that high frequency information is lost and the second is that the low frequencies are distorted due to spectrum superpositions. Since the first problem cannot be solved using uniform sampling at such a low rate, we will focus on avoiding the second.

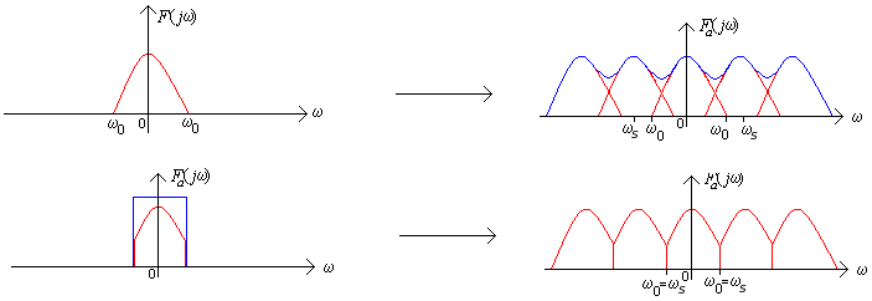


Figure C.5: Anti-aliasing filter.

The idea is to smoothen the signal before sampling, i.e., to extract high frequencies by applying a low-pass filter. Filtering the high frequency information and then projecting the result signal on a delayed *Dirac* function is equivalt to projecting the original signal on a small pulse waveform $v(t)$, as shown in Figure C.6.

It is interesting to point out that this kind of sampling is actually common and easier to implement than the *Dirac* comb. A camera, for instance, when acquiring an image, sets for each pixel an average of the surrounding values. This is not only a good procedure because it minimizes distortion effects, but also because it is easier to implement on hardware. Point sampling in a camera doesn't gather much light, and therefore the signal to noise ratio will be inadequate. Moreover, sampling by *Diracs* would require a very precise sensing mechanism, and usually electron beams have Gaussian intensity functions.

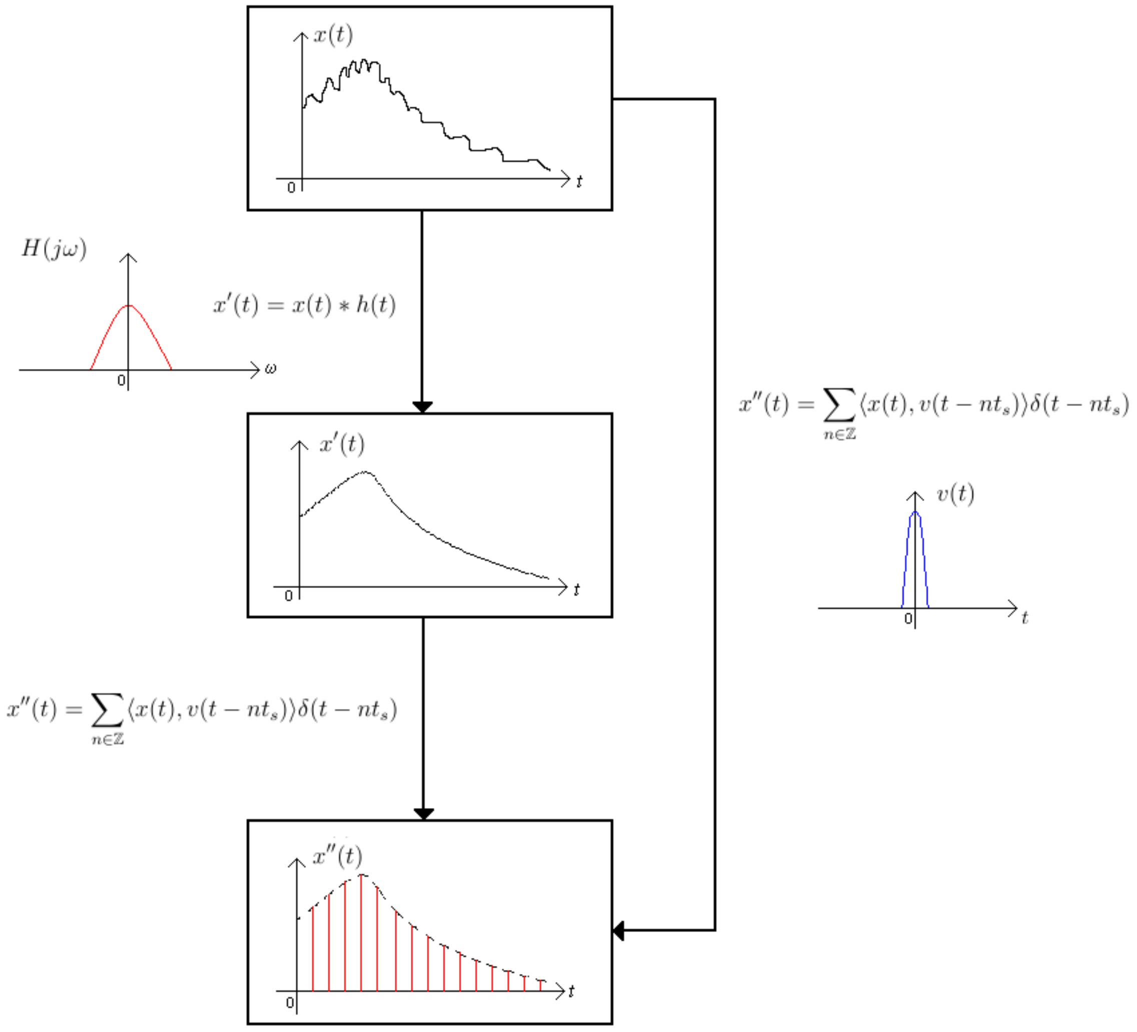
Consider that $v(t)$ is scaled *sinc* function. In this case, we are projecting the signal on a basis of delayed *sincs* $(v_n)_{n\in\mathbb{Z}}$, where

$$v_n(t) = \text{sinc}\left(\frac{t - nt_s}{t_s}\right)$$

This is, in fact, an orthogonal basis and, therefore, we can reconstruct the signal by

$$\hat{x} = \sum_{n\in\mathbb{Z}} \langle x, v_n \rangle v_n$$

If $t_s$ is such that the Nyquist limit is respected, then reconstruction is exact $(\hat{x} = x)$; however, if $t_s$ is large, then we are taking a signal of a Hilbert space and projecting it in the subspace spanned by $(e_n)_{n\in\mathbb{Z}}$. Notice that this projection is



$x(t)$
$t$
0
$H(j\omega)$
$\omega$
0
$x'(t) = x(t) * h(t)$
$x'(t)$
$t$
0
$x''(t) = \sum_{n\in\mathbb{Z}} \langle x(t), v(t-nt_s)\rangle \delta(t-nt_s)$
$v(t)$
$t$
0
$x''(t) = \sum_{n\in\mathbb{Z}} \langle x(t), v(t-nt_s)\rangle \delta(t-nt_s)$
$x''(t)$
$t$
0

Figure C.6: Undersampling.

taking a vector from a subspace of higher dimension and projecting it in a subspace of lower dimension and, therefore, this is a form of compression.

## C.4 Approximation Theory

Being able to represent signals using different bases is usefull in signal processing because it allows to approximate certain types of signals using just a few vectors.

In this section we will exploit in a more formal way what was just illustrated by the undersampling problem.

### C.4.1 Approximation on a Linear Basis

Given a signal $x$ and an orthogonal basis $\mathcal{B} = (\phi_\lambda)_{\lambda \in \Gamma}$, an approximation projects $x$ over $M$ basis vectors

$$x_M = \sum_{n \in I_M} \langle x, \phi_n \rangle \phi_n \tag{C.1}$$

The choice of the $M$ vectors can be done *a priori* or *a posteriori* (depending on the signal $x$). In the first case, the approximation is called linear and, in the second, non-linear.

Though linear approximations are simpler to implement, the distortion generated will highly depend on the original signal, whereas in the non-liner case we can adapt the projection vector to minimize the approximation error.

In this context, we can discuss DCT linear and non-linear compression studied in Section B.6. The DCT involves projecting the signal into a basis that makes it sparse and the run-length coding involves choosing from this new basis the most significant vectors. In this non-linear procedure, we need to save each coefficient value and its 'position', which refers to the vectors of this new basis that are most important to represent the signal. In linear compression, the significant vectors are known *a priori*, and we only need to store the coordinate values, which are the projections of the signal on each base vector.

### C.4.2 Approximation on Overcomplete Dictionaries

Linear expansion in a single basis is not always efficient because the information will be diluted across the whole basis. In overcomplete dictionaries [25], however, we are able to express the same signal using a smaller number of coefficients. Mallat illustrated this idea [3] by comparing signal representations to language vocabularies. While a small vocabulary may be sufficient to express any idea, it will sometimes require the use of full sentences to replace unavailable words otherwise available in large dictionaries.

Therefore, a good compression scheme involves finding the best representation of an image using a redundant dictionary. It is noteworthy that a trade-off considering the dictionary's size must be analyzed because, while a big dictionary guarantees a small number of values necessary to represent a given signal, it also demands a large number of bits to determine each coefficient.

Due to redundancy there are, however, innumerable ways to represent the same signal. The intention of most of the developed techniques is to find a representation which concentrates the energy in a small number of coefficients.

What we are looking for is a sparse representation, i.e., a representation with a larger number of zero coefficients. We can reduce this problem to the one of finding, for a given N-dimensional signal $x$, a $P$-sized dictionary $\mathcal{D} = \{g_1, g_2, \ldots, g_P\}$, and a value $M$, $M < N < P$, the representation

$$x_M = \sum_{m=0}^{M-1} \alpha_{p_m} g_{p_m} \tag{C.2}$$

that minimizes $\|x - x_M\|$.

This problem, however, is combinatorial and NP-hard. Thus, a series of pursuit methods were developed to reduce computational complexity by searching efficient but non-optimal approximations. To illustrate how the latter perform, we will overview two very popular algorithms.

### Basis Pursuits

Basis pursuits [26] consists in solving the following convex optimization problem with inequality constraints

$$\min \|\boldsymbol{\alpha}\|_1, \text{ subject to } \sum_{p=0}^{P-1} \alpha_p g_p = x$$

where $\boldsymbol{\alpha}$ is a vector of dimension $P$ containing the $\alpha_p$ coefficients.

This is more a principle than an algorithm, and there are many computational solutions to this problem, the most popular ones using linear programming.

The idea behind this technique is that the $l_1$-norm enhances sparsity, as will be discussed in Appendix D.

Therefore a good approximation strategy results from extracting the $M$ largest coefficients of the optimal $P$-sized $\boldsymbol{\alpha}$ vector.

### Matching Pursuits

Matching pursuit [3] is a greedy algorithm that decomposes a signal into a linear expansion of waveforms that are selected from a redundant dictionary.

At each step, the dictionary element that best matches the signal structure is chosen and the projection of the signal on it is stored. This process is repeated $M$ times using the residual which results from the subtraction.

The advantage of this technique is that it is less computationaly expensive than Basis Pursuits and very powerful in terms of performance. It also shares many interesting properties such as energy conservation and invertibility when $M = P$. However, since it maximizes the projection at each step without considering the overall signal structure, it is suboptimal.

# Appendix D

# Compressive Sensing: An Overview

Up until now we have been following the *sample-then-compress* framework, i.e., for a given image, we find a sparse representation and then encode the significant coefficients. The shortcomings of this approach are that before a compressing scheme can be applied, the encoder must:

- store a large number of samples;
- compute all the transform coefficients; and
- find the locations of the large coefficients.

This is what usually happens in popular image acquisition instruments. Common digital cameras sample at a large number of mega-pixels, but store the images in a compressed form, for example, the JPEG standard. This indicates that we only need a small percentage of the measured coefficients to reconstruct the signal and, therefore, efficiency is lost.

This suggests that a smarter and cheaper method could be used to improve performance. In this context, Compressive Sensing appears. It involves sampling the original signal in a rate smaller than the Nyquist limit and reconstructing it by means of an optimization procedure.

In this appendix we will study the principal concepts of this novel idea and how it first came to existence. We will leave a greater formalization of the theory involved for the next appendix.

## D.1 Essential Aspects

What we want is to build an acquisition scheme that captures the image already in its compressed form. Consider the DCT based compression scheme. If we knew *a priori* which were the most significant DCT coefficients (consider, for instance, a linear compression scheme), we could then simply measure their values without the need of exploiting each pixel information.

Note that the word *sample* here has a new meaning. It refers no longer to point samples, but rather to more general linear measurements of the signal. Each measurement $y_m$ in the acquisition system is an inner product of the signal $x$ against a different test function $\phi_m$ (for example, a row of the DCT transform matrix)

$$y_1 = \langle x, \phi_1 \rangle \,, \quad y_2 = \langle x, \phi_2 \rangle \,, \quad \dots \quad , \quad y_M = \langle x, \phi_M \rangle$$

where $M$ is the number of measurements.

However, as we have seen in the previous appendixes, linear approximations usually have performances that are far from optimal, illustrating that this *a priori* knowledge is hard to obtain. Accordingly, though it is true that $x$ is sparse in some domain, we can not know exactly which are the significant coefficients. Moreover, it is desirable to obtain a *nonadaptive* solution to the problem, so as to be able to use the same mechanism to capture information from any signal.

### D.1.1 The Algebraic Problem

Let $s$ be the signal represented in a sparse domain, i.e,

$$s = \Psi x$$

where $x$ is the original signal and $\Psi$ is a transformation that makes $s$ sparse, for example, the DCT.

To take a small number of measurements is to multiply $x$ by a fat[1] matrix $\Phi_\Omega$ as shown in Figure D.1, where each row is a measurement function $\phi_m$.

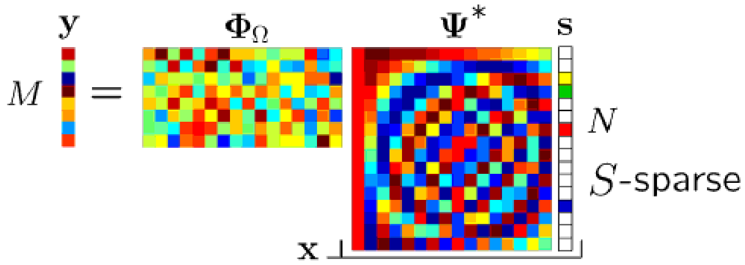


Figure D.1: The acquisition matrix. (Extracted from [4].)

$$y = \Phi_\Omega x$$
$$x = \Psi^* s \Longleftrightarrow s = \Psi x$$
$$y = \Theta_\Omega s, \text{ where } \Theta_\Omega = \Phi_\Omega \cdot \Psi^*$$

The reconstruction problem involves finding $x$ so that $y = \Phi_\Omega x$, or, analogously, $s$ so that $y = \Theta_\Omega s$. This problem, however, is ill posed because there is an infinite number of possible solutions. All the same, not all solutions satisfy the sparsity property of $s$ and, therefore, a simple choice would consist of searching among all possible solutions the one that makes $s$ the sparsest.

### D.1.2 Sparsity and the $l_1$ norm

Sparsity can be described by the $l_0$ norm

$$\|\alpha\|_{l_0} = \sharp\{i : \alpha(i) \neq 0\}$$

Hence, the solution we want is

$$\min_x \|\Psi x\|_{l_0} \quad \text{subject to} \quad \Phi_\Omega x = y$$

[1] We use the term *fat* to refer to a matrix where the number of rows exceeds the number of columns.

Or, alternatively

$$\min_s \|s\|_{l_0} \quad \text{subject to} \quad \Theta_\Omega s = y$$

Yet, this problem is combinatorial and NP-hard; however it can be proved that sparse signals have small $l_1$ norms relative to their energy. We will motivate the relation between the $l_0$ and the $l_1$ norm by the 2-dimensional example in Figure D.2.

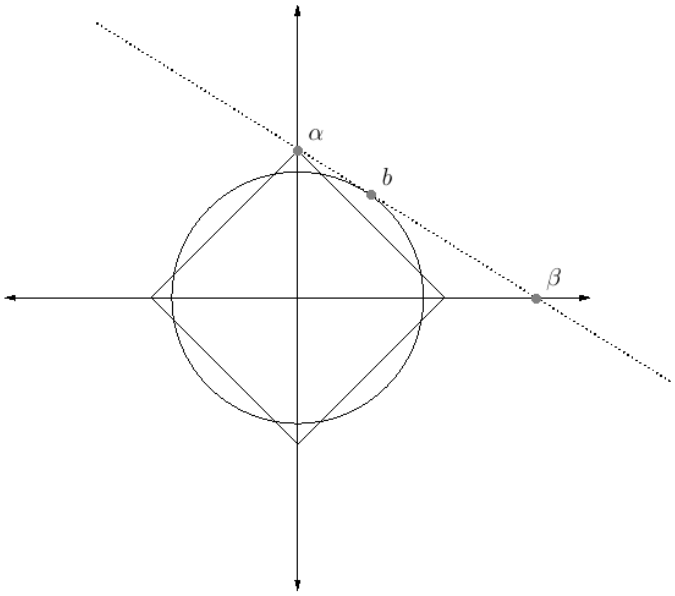


Figure D.2: Sparsity and the $l_1$ norm.

Suppose we wish to find the signal $s$ that has minimum $l_0$ norm, given that $s$ respects a linear equation that constrains its position in $\mathbb{R}^2$ to the dotted line. Note that if we minimize the $l_2$ norm the optimal solution will be given by $s = b$, which is not sparse and far from the $l_0$ solutions $\alpha$ and $\beta$. However, the $l_1$ minimization would result in $s = \alpha$, which is the exact solution we wanted.

The $l_1$ norm is convex, which makes optimization problem computationally tractable. Hence, all the following analyses and results will be given considering $l_1$ minimization.

### D.1.3 The Recovery Algorithm

We can now understand the idea of Compressive Sensing in terms of its recovery algorithm. This theory involves undersampling a signal and then recovering it by the convex optimization problem

$$\min_{s} \|s\|_{l_1} \quad \text{subject to} \quad \Theta_\Omega s = y$$

Though we have understood why this is a good procedure, we still have to analyze its efficiency. How can we know for sure that the sparsest solution is the one that reconstructs the original signal $s$? What do we need to assume about the sensing matrix and the number of samples? What kind of results can we guarantee?

A series of theorems and definitions have been proposed to formalize this idea and specify sufficient conditions that guarantee good results. These will be studied with some care in the following appendix. We will, nevertheless, take some time to introduce the first theorem proposed in this field. Though it is much weaker than the ones that will be considered in the future, it sheds light to many interesting ideas, as well as how the researchers first came up with CS.

## D.2 The Fourier Sampling Theorem

### D.2.1 The Magnetic Resonance Imaging Problem

The classical tomography problem consists in reconstructing a 2D image $x$ from samples of its Fourier transform $\hat{x}(\omega)$ on the star shaped domain $\Omega$ illustrated by Figure D.3.

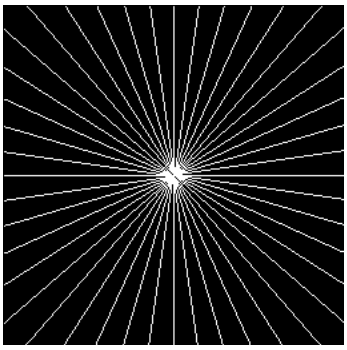
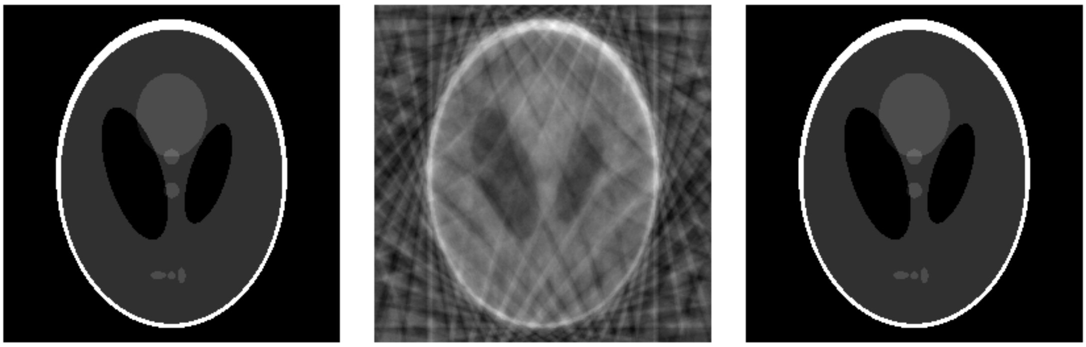


Figure D.3: Sampling domain $\Omega$ in the frequency plane. (Extracted from [12].)

The most common algorithm, called *filtered backprojection*, assumes the non-sampled Fourier coefficients to be zero, in this way reconstructing the image with minimal energy. An image reconstructed by this procedure is shown in Figure D.4 and illustrates how this mechanism has a bad performance.



(a) Original image.

(b) Reconstruction with filtered backprojection.

(c) Reconstruction using convex optimization.

Figure D.4: First CS experiment applied to the Logan-Shepp phantom test image. (Extracted from [12].)

The solution proposed by [12] involves guessing the missing Fourier coefficients by means of a convex optimization based on the total-variation norm [2]

$$\min_y \|y\|_{TV} \quad \text{subject to} \quad \hat{y}(\omega) = \hat{x}(\omega), \forall \omega \in \Omega$$

This was implemented with some numerical constants and resulted in the *exact* reconstruction of the original image. This surprising result led the researches to formalize a new sampling theorem.

## D.2.2 New Sampling Theorem

**Theorem 1** (Fourier Sampling Theorem [12]). *Assume that $x \in \mathbb{R}^N$ is S-sparse and that we are given M Fourier coefficients with frequencies selected uniformly at*

[2] The total-variation (TV) norm can be interpreted as the $l_1$-norm of the (appropriately discretized) gradient.

*random*[3]. *Suppose that the number of measurements*[4] *obeys*

$$M \geq C \cdot S \cdot \log N$$

*where $C$ is a relatively small constant. Then minimizing*

$$\min_{s} \|s\|_{l_1} \quad \textit{subject to} \quad \Theta_\Omega s = y$$

*reconstructs $x$ exactly with overwhelming probability.*

This theorem differs from usual constraint specifications because it involves probabilistic results. The reason for this rather unorthodox approach is that we cannot obtain powerful results if we consider all measurable sets of size $M$, as there are some special sparse signals that vanish nearly everywhere in the Fourier domain.

To illustrate this, consider the discrete *Dirac* comb in $\mathbb{R}^N$, where $N$ is a perfect square and the signal spikes are equally spaced by $\sqrt{N}$, as shown in Figure D.5.

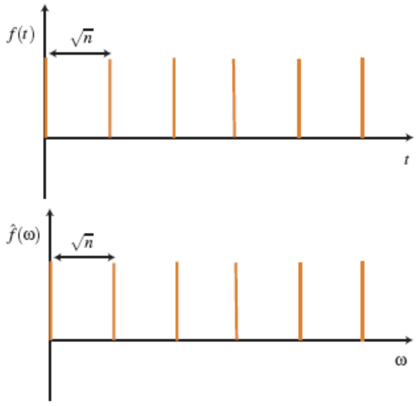


Figure D.5: Comb filter. (Extracted from [15].)

Let $\Omega$ be the set of all frequencies but the multiples of $\sqrt{N}$. Then the observed signal in the Fourier domain is equal to zero and the reconstruction is identically zero. Note that the problem here does not really have anything to do with $l_1$ minimization

[3] In this case, we denote by $\Phi$ the $N \times N$ Fourier transform matrix and by $\Phi_\Omega$ the fat matrix created by extracting $N$ rows of $\Phi$.

[4] It is common in literature to denote the set that supports the signal by $T$ and the sampling set by $\Omega$. Therefore, $S = |T|$ and $M = |\Omega|$.

once the signal cannot be reconstructed from its Fourier samples using any possible method.

Another interesting point to analyze is whether it would be possible to recover an arbitrary signal from less than $CS \log N$ samples using another algorithm. To motivate that this solution is tight we will use the same example of the *Dirac* comb. If $x$ is as shown in Figure D.5, to be able to recover it from $\hat{x}$, the observation set $\Omega$ must contain at least on spike. Supposing that

$$|T| < |\Omega| < \frac{N}{2} \Longleftrightarrow \sqrt{N} < M < \frac{N}{2}$$

and choosing $\Omega$ uniformly at random, the probability that no spike is chosen is given by [12]

$$P = \frac{\binom{N-\sqrt{N}}{M}}{\binom{N}{M}} \geq \left(1 - \frac{2M}{N}\right)^{\sqrt{N}}$$

Therefore, for the probability of unsuccessful recovery to be smaller that $N^{-\delta}$, it must be true that

$$\sqrt{N} \cdot \log\left(1 - \frac{2M}{N}\right) \leq -\delta \log N$$

Since $M < \frac{N}{2}$, $\log\left(1 - \frac{2M}{N}\right) \approx -\frac{2M}{N}$ and we obtain the solution

$$M \geq Const \cdot \delta \cdot \sqrt{N} \cdot \log N$$

Hence, we conclude that the above theorem identifies a fundamental limit, and thus no recovery can be successfully achieved with significantly fewer observations.

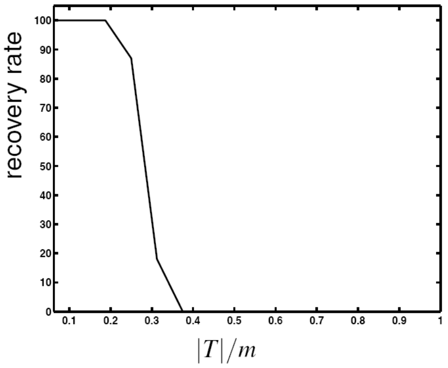
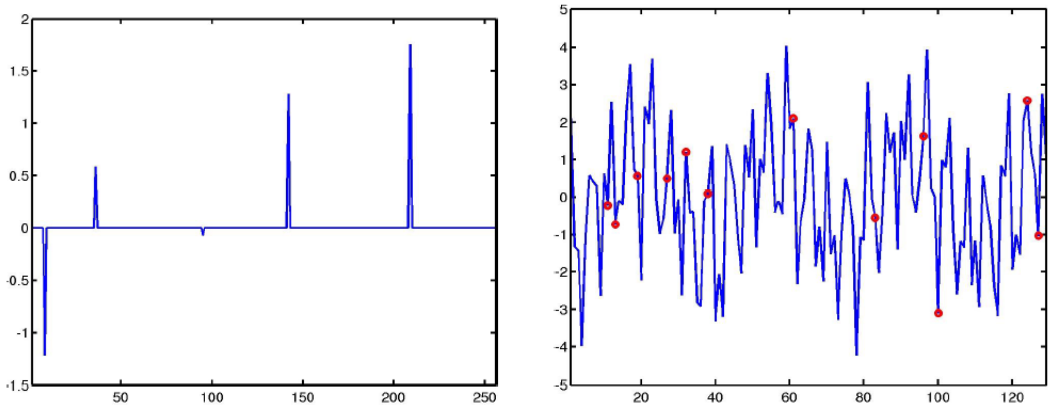


Figure D.6: Numerical example. (Extracted from [15].)

A final illustration is given in Figure D.6, which shows how the recovery rate decreases when the number of samples decreases in relation to the set that supports the signal. To build this graph signals of size $n = 1024$ were used and $|T|$ spikes were randomly placed.

### D.2.3 Relashionship with Nyquist Sampling Theorem



(a) Sparse signal in the time domain.

(b) Fourier transform of (a) and CS measurements (dots).

Figure D.7: CS intrepolation ploblem. (Extracted from [15].)

Consider the signal in Figure D.7(a). To follow the Nyquist sampling scheme, we would have to consider the size of the signal band in the frequency domain and sample it at twice that rate. In CS theory, on the other hand, we don't have to consider the signal band at all. All that is relevant is the number of nonzero coefficients which, multiplied by a log factor, gives us the sensing rate.

When sampling in the Fourier domain, the measurements are as shown by the red dots of Figure D.7(b), and reconstruction involves an interpolation procedure that returns the blue curve. Notice, however, that this problem cannot be solved by a simple interpolation formula, as is done in the Nyquist sampling theorem with the *sinc* function. Instead, we reach the interpolated result by means of a convex optimization procedure that minimizes the $l_1$ norm of the sparse signal.

This problem was solved by [15] and the recovery is exact.

## D.3 Uncertainty Principles

Though CS may seem like a great breakthrough, the basic principles around it have been known for quite some time. In fact, we can consider this novel idea as an extension of the theory about uncertainty principles.

We have already mentioned in our study of the Wavelet transform in Section B.2.3 that a function and its Fourier transform cannot both be highly concentrated. We can extend this uncertainty principle to functions $x$ that are not concentrated in an *interval*. Instead, if $x$ is practically zero outside a measurable set $T$ and its Fourier transform $\hat{x}$ is practically zero outside a measurable set $\Omega$, then

$$|T| \cdot |\Omega| \geq 1 - \delta$$

where $\delta$ is an oscillation parameter related to the *practically zero* definition.

In the discrete case, if $x \in \mathbb{R}^N$ has $N_t$ nonzero components and $\hat{x}$ is not zero at $N_\omega$, the uncertainty principle states that

$$N_t \cdot N_\omega \geq N$$

where the lower bound $N_t N_\omega = N$ is reached in the case where $x$ is a *Dirac* comb. Note that this happens in the example shown in Figure D.5, where $N_t = \sqrt{N}$ and $N_\omega = \sqrt{N}$.

In most common studies, uncertainty principles are used to prove that certain things are impossible, for example, obtaing good resolutions simultaneously in the time and frequency domains. However, in this approach, we make use of this theorem to allow recovery of signals despite amounts of missing information.

Donoho and Stark showed in [27] that it is possible to recover a bandlimited signal when sampled with missing elements. Consider that the signal $x$, where $\hat{x} \in \Omega$, is observed in the time domain but a subset $T^c$ of the information is lost. Then the observed signal $r(t)$ is such that

$$r(t) = \begin{cases} x(t) + n(t), & \text{if } t \in T \\ 0, & \text{if } t \in T^c \end{cases}$$

where $n(t)$ is a noise signal.

It can be demonstrated that $x$ can be recovered from $r$, provided that $|T^c||\Omega| < 1$.

Intuitively, consider the signal $h$, $\hat{h} \in \Omega$, completely concentrated on $T^c$. The problem of reconstructing $s$ from $r$ derives from the fact that $x$ and $x + h$ cannot be distinguished and therefore the reconstruction error can be arbitrary large. However, such function $h$ cannot exist because if it did the uncertainty principle would require $|T^c||\Omega| \geq 1$. Hence, a stable reconstruction to the above problem can be achieved.

## D.4 Extensions

The practical relevance of Theorem 1 has two limitations. The first one is that it restricts the sampling domain to Fourier and we are not always at liberty to choose the types of measurements we use to acquire a signal. The second is that completely unstructured measurement systems are computationally hard.

In view of these shortcomings, a significant amount of effort has been given to make CS theory useful for practical applications. Not only have researches expanded this result, but they also described conditions that guarantee good performances in adverse situations.

# Appendix E

# Compressive Sensing: Theoretical Aspects

In the previous appendix we introduced a sampling theory that allows compression. We will now provide some key mathematical insights underlying this new argument.

Two different approaches will be used:

- Basic CS - theory that stipulates constraints for the exact recovery of sparse signals.
- Robust CS - expansion of the former approach to allow CS to be used in applications where the signal is not exactly sparse or the measurements are corrupted by noise.

This appendix also includes some important considerations for the design of efficient sensing matrices.

## E.1 Basic CS

Basic CS deals with analyzing the constraints that guarantee perfect reconstruction by means of an $l_1$ optimization, considering that there exists a domain in which the signal $x$ is $S$-sparse and that the acquired measurements are not corrupted by noise.

The first concept that needs to be extended from the discussed *Fourier Sampling Theorem* is that the domain where $x$ is sparse and the domain where the samples are taken may vary in different applications, not necessarily being time and frequency. Therefore, it is of utmost importance to develop a way of determining if a sampling domain is efficient, given that the signal is sparse after it is multiplied by $\Psi$, where $\Psi$ is, for example, a wavelet transform. [1]

### E.1.1 Incoherence

Coherence [7] is a measurement of the correlation between the sensing waveforms $\phi_k$ and the waveforms where the signal is supposed to be sparse $\psi_k$. Assuming both have unit $l_2$ norm, the definition is as follows.

**Definition 3** (Coherence between $\Psi$ and $\Phi$ [5]).

$$\mu(\Phi, \Psi) = \sqrt{N} \max_{i,j} |\langle \phi_i, \psi_j \rangle| \quad , \quad \|\phi_i\|_{l_2} \quad \|\psi_i\|_{l_2} = 1$$

Note that $\mu(\Phi, \Psi)$ measures the minimum angle between the sensing waveforms

[1] Notation review:

We use $x$ to refer to an input signal and $s$ to denote its $S$-sparse representation. $T$ is the set that supports $s$ and is of size $|T| = S$ and $\Omega$ is the random measurement subset of size $|\Omega| = M$.

We denote by $\Phi$ the matrix that spans $\mathbb{R}^N$, where each row is a measurement function $\phi_m$ to be applied to the signal $x$. Therefore, the sensing problem is

$$y = \Phi_\Omega x$$

where $\Phi_\Omega$ is a fat matrix created by randomly selecting $M$ rows of $\Phi$. Since $x$ is sparse in the $\Psi$ domain, the sparse representation of $x$ is given by

$$s = \Psi x$$

And therefore, since $\Psi$ is unitary (orthonormal transform),

$$y = \Phi_\Omega \Psi^* s$$
$$\Rightarrow y = \Theta_\Omega s, \text{ where } \Theta_\Omega = \Phi_\Omega \Psi^*$$

We also denote $\Theta = \Phi\Psi^*$ and $\Theta_{\Omega T}$ is the submatrix created by extracting the columns of $\Theta_\Omega$ corresponding to the indexes of $T$. Note that $\Theta$ is $N \times N$, $\Theta_\Omega$ is $M \times N$, and $\Theta_{\Omega T}$ is $M \times S$.

and the sparsity waveforms. Therefore, if we look at the waveforms as vectors in $R^N$, then high incoherencies mean that these vectors are far apart, i.e., nearly orthogonal.

From linear algebra we get

$$1 \leq \mu(\Phi, \Psi) \leq \sqrt{N}$$

**Demostration:** The upper bound comes from the Cauchy-Schwarz inequality

$$|\langle \phi_i, \psi_j \rangle|^2 \leq \|\phi_i\|^2 \cdot \|\phi_j\|^2 \ \Rightarrow \ \mu(\Phi, \Psi) \leq \sqrt{N}$$

and the lower bound can be derived if we consider that $\Psi$ is an orthogonal basis

$$\sum_j |\langle \phi_i, \psi_j \rangle|^2 = 1 \ \Rightarrow \ \max_j |\langle \phi_i, \psi_j \rangle| \geq \frac{1}{\sqrt{N}} \ \Rightarrow \ \mu(\Phi, \Psi) \geq 1$$

□

Therefore, the time and the frequency domains are maximally incoherent, since the Fourier basis $\psi_k(t) = \frac{1}{\sqrt{N}} e^{\frac{2\pi jk}{N}}$ and the canonical basis $\phi_k(t) = \delta(t-k)$ yield $\mu = 1$. This is very good because better results are achieved when coherence is small, i.e., when both domains are poorly correlated.

We can perceive this observation if we notice that sampling in the sparse domain directly returns many zero-valued coefficients. The advantage of incoherence is that if we measure a series of random combinations of the entries, we learn something new about the sparse vector with every measurement.

We can also define incoherence based on the matrix $\Theta$.

**Definition 4** (Mutual Coherence [6]).

$$\mu(\Theta) = \sqrt{N} \max_{i,j} |\Theta_{i,j}|$$

Notice that this is equivalent to Definition 3

$$\Theta = \begin{bmatrix} \phi_1^T \\ \vdots \\ \phi_N^T \end{bmatrix} \begin{bmatrix} \psi_1^* & \dots & \psi_N^* \end{bmatrix} = \begin{bmatrix} \phi_1^T \psi_1^* & \dots & \phi_1^T \psi_N^* \\ \vdots & \ddots & \vdots \\ \phi_N^T \psi_1^* & \dots & \phi_N^T \psi_N^* \end{bmatrix}$$

And, since each row (or column) of $\Theta$ has necessarily an unitary $l_2$-norm [2], $\mu$ will take a value between 1 and $\sqrt{N}$.

In terms of the matrix $\Theta$, $\mu$ can be interpreted as a rough measure of how concentrated the rows of $\Theta$ are. From the above comment we notice that if there is a coincident vector $\phi_i$ and $\psi_j$, the $i^{th}$ row of $\Theta$ will be maximally concentrated, i.e., $\Theta_{i,j} = 1$ and $\Theta_{i,k} = 0, \forall k \neq i$. On the other hand, the best recovery possibility occurs if $\phi_i$ is spread out in the $\Psi$ domain, i.e., when the row is diluted: $\Theta_{i,k} = \frac{1}{\sqrt{N}}, \forall k$.

### E.1.2 Result Theorem

**Theorem 2** ([7]). *Let $\Theta$ be an $N \times N$ orthogonal matrix and $\mu(\Theta)$ be as defined previously. Fix a subset $T$ of the signal domain. Choose a subset $\Omega$ of the measurement domain of size $M$, and a sign sequence $z$ on $T$ uniformly at random. Suppose that*

$$M \geq C_0 \cdot |T| \cdot \mu^2(\Theta) \cdot \log(N)$$

*for some fixed numerical constants $C_0$. Then for every signal $s$ supported on $T$ with signs matching $z$, the recovery from $y = \Theta_\Omega s$ by solving*

$$\hat{s} = \min_{s^*} \|s^*\|_{l_1} \quad \textit{subject to} \quad \Theta_\Omega s^* = y$$

*Is exact ($\hat{s} = s$) with overwhelming probability.*

Theorem 2 extends the previous *Fourier Sampling Theorem* with the exception that the latter holds for each sign sequence. The need to randomize the signs comes from an artifact that was used to demonstrate the thesis. It is highly probable that it still holds without this constraint, however researchers have not been able to prove this up until now [15].

We will not demonstrate this theorem here, but we will give two examples that serve as insights to its tightness.

[2] The rows have unitary $l_2$-norm if we consider $\Psi$ orthonormal and the columns have unitary $l_2$-norm if we consider $\Phi$ orthonormal.

To show this is a fundamental limit, consider $\Psi$ the time and $\Phi$ the frequency domain. Then, $\mu = 1$ and the above theorem results in the *Fourier Sampling Theorem*, which we have proven to be tight.

On the other hand, consider that $\Phi$ and $\Psi$ are the same, i.e., $\mu^2(\Phi, \Psi) = N$ and we want to recover a signal that is 1-sparse. The theorem says that we actually need to measure every coefficient to guarantee recovery. This is intuitive because since each measurement informs only one of the $\psi_k$ coefficients, unless we measure the nonzero coefficient, the information will vanish. Therefore, to reconstruct $x$ with probability greater than $1 - \delta$, we need to see all $\phi_k$ components.

The latter result is maintained without the need to assume $\Phi = \Psi$, as long as we consider both orthogonal. In fact, if there exists two coefficients $i$ and $j$, such that $|\langle \phi_i, \psi_j \rangle| = 1$, then $\mu(\Phi, \Psi) = \sqrt{N}$ and the number of measurements needed to recover a 1-sparse signal $x$ is $N$. To see this result intuitively, note that $\theta_{i,j} = 1$, $\theta_{i,k} = 0, \forall k \neq j$ and $\theta_{k,j} = 0, \forall k \neq j$. Therefore, $y = \Theta s$ can be rewritten as:

$$y = \begin{bmatrix} * & \dots & * & 0 & * & \dots & * \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ * & \dots & * & 0 & * & \dots & * \\ 0 & \dots & 0 & 1 & 0 & \dots & 0 \\ * & \dots & * & 0 & * & \dots & * \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ * & \dots & * & 0 & * & \dots & * \end{bmatrix} \begin{bmatrix} 0 \\ \vdots \\ 0 \\ * \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

Notice that unless $\phi_j$ is chosen, i.e., unless $j \in \Omega$ we will not obtain any information because $\Theta_\Omega s = 0$. Therefore, to guarantee recovery we must sample with the hole matrix $\Theta_\Omega = \Theta$.

## E.2 Restricted Isometries

In this section, we will define strict conditions that when imposed in the matrix $\Theta$ guarantee that CS is efficient.

### E.2.1 An Uncertainty Principle

Below is an intermediate result that follows directly from incoherence.

**Theorem 3** ([15]). *Let $\Theta$, $T$, and $\Omega$ be as in Theorem 2. Suppose that the number of measurements $M$ obeys*

$$M \geq \cdot|T| \cdot \mu^2(\Theta) \cdot max\left(C_1 log|T|, C_2 \log\left(3/\delta\right)\right),$$

*for some positive constants $C_1$, $C_2$. Then*

$$P\left(\left\|\frac{N}{M}\Theta^*_{\Omega T}\Theta_{\Omega T} - I\right\| \geq 1/2\right) \leq \delta$$

The above equation means that all the eigenvalues of $\frac{N}{M}\Theta^*_{\Omega T}\Theta_{\Omega T}$ are between $\frac{1}{2}$ and $\frac{3}{2}$. To see that this is an uncertainty principle, let $s \in \mathbb{R}^N$ be a sequence supported on $T$, and suppose that $\|\frac{N}{M}\Theta^*_{\Omega T}\Theta_{\Omega T} - I\| \leq 1/2$ (which is very likely the case). It follows that

$$\frac{1}{2}\cdot\frac{M}{N}\cdot\|s\|^2_{l_2} \quad\leq\quad \|\Theta_\Omega s\|^2_{l_2} \quad\leq\quad \frac{3}{2}\cdot\frac{M}{N}\cdot\|s\|^2_{l_2}$$

This asserts that the portion of the energy of $s$ that will be concentrated on the set $\Omega$ is essentially proportional to $M$. Notice that $\|s\|^2_{l_2} = \|\Theta s\|^2_{l_2}$ and, therefore, we can rewrite the equation as

$$\frac{1}{2}\cdot\frac{M}{N}\cdot\|\bar{s}\|^2_{l_2} \quad\leq\quad \|\bar{s}_\Omega\|^2_{l_2} \quad\leq\quad \frac{3}{2}\cdot\frac{M}{N}\cdot\|\bar{s}\|^2_{l_2}$$

where $\bar{s} = \Theta s$ and $\bar{s}_\Omega$ is $\bar{s}$ restricted to set $\Omega$, $\bar{s}_\Omega = \Theta_\Omega s$.

Hence, the relation says that the energy of the signal restricted of the set $\Omega$ is much smaller than the energy of the signal. This is an uncertainty relation because it means that if a signal is $S$-sparse (if the signal is concentrated on $T$), then it cannot be concentrated on the set $\Omega$. If fact, this relation is quantized because there is a fixed value $M/N$ to which the concentration in each domain is proportional.

Though usually uncertainty principles are considered bad, this one actually makes recovery possible. We can only take less measurements because the energy is diluted in the $\Phi$ domain and, thus, by taking random measurements, we are able to obtain a considerate amount of information about the signal.

### E.2.2 The Restricted Isometry Property

Based on the intermediate result presented in Section E.2.1, Candès and Tao defined in [6] the restricted isometry property. A refined approach appears in [8].

**Definition 5** (Restricted Isometry Constant [8])**.** *For each integer $S = 1, 2, \ldots, N$ we define the $S$-restricted isometry constant $\delta_S$ of a matrix $\Theta_\Omega$ as the smallest number such that*

$$(1-\delta_S)\|s\|_{l_2}^2 \leq \|\Theta_{\Omega T} s\|_{l_2}^2 \leq (1+\delta_S)\|s\|_{l_2}^2$$

*for all $S$-sparse vectors.*

The restricted isometry is a property of the measurement matrix $\Theta_\Omega$ that refers to the existence and boundary of $\delta_S$. The RIP establishes a condition which, if obeyed by $\Theta_\Omega$, guarantees recovery of sparse vectors. Notice that the constant $\delta_S$ is intrinsic to the structure of $\Theta_\Omega$ and, therefore, by setting constraints to its size, we can quantify the efficiency of the sensing matrix.

The reason we call this RIP is straightforward: the energy of the signal restricted to the set $\Omega$ is proportional to the size of $\Omega$. Nevertheless, some authors describe this as an Uniform Uncertainty principle (UUP). The relation to the uncertainty principles has already been established in Section E.2.1 and involves the guarantee that the signal cannot be concentrated simultaneously on both sets. This condition, however, is stronger than Theorem 3 because it is valid for every set $T$ (every S-sparse vector). Hence, it is called *uniform.*

We will now try to illustrate what this property means in terms of linear algebra. By undersampling we get an ill posed problem and, from the infinite number of solutions, we are going to choose the one that makes $s$ the sparsest. However, how can we know for sure that this solution is unique? How can we force that there will be no other solution that is as sparse as $s$ or sparser? As mentioned earlier, we can only guarantee this if we have incoherent measurements, i.e., if the sensing matrix has some properties.

First of all, note that if $\Theta_\Omega$ has linear dependent columns, two different sparse vectors can result in the same measurement.

**Demostration:**

$$\Theta_\Omega \cdot c = \sum_{j=1}^{N} c_j \cdot v_j, \quad \text{where } v_j \text{ is a column of } \Theta_\Omega$$

Let $c \neq 0$ be a vector such that $\sum_{j=1}^{N} c_j \cdot v_j = 0$ (this is always possible because the columns are l.d.). Then, if we partition the set of indexes $I = \{1, 2, \ldots, N\}$ into two disjoint sets $I_1 \cup I_2 = I$, it results that

$$\Theta_\Omega \cdot c = \sum_{j \in I_1} c_j \cdot v_j = \sum_{j \in I_2} -c_j \cdot v_j$$

And we measure the vectors $a$ and $b$ defined as follows

$$a = \begin{cases} a_j = c_j, \text{ if } j \in I_1 \\ a_j = 0, \text{ if } j \in I_2 \end{cases} \qquad b = \begin{cases} b_j = -c_j, \text{ if } j \in I_2 \\ b_j = 0, \text{ if } j \in I_1 \end{cases}$$

by $\Theta_\Omega$, we obtain the same result $y = \Theta_\Omega a = \Theta_\Omega b$. □

Hence, we conclude that the existence of linear dependent columns lead to equivalent measurements for two different input signals and, therefore, recovery can only be guaranteed if the columns are linear independent. However, we cannot impose linear independence because the matrix is fat, i.e., the number of columns is larger than the number of rows. Here again sparsity comes to the rescue. All we need is that the columns of $\Theta_\Omega$ behave like an l.i. system for sparse linear combinations involving no more than $S$ vectors. That is exactly what the RIP gives us, it says that for every $T$ of size no bigger than $S$, $\Theta_{\Omega T}$ is approximately orthogonal.

It can be easily shown that, if $\delta_{2S} < 1$ for $S \geq 1$, for any $T$ such that $|T| \leq S$, there is a unique $s$ with $\|s\|_{l_0} \leq S$ and obeying $y = \Theta_\Omega s$.

**Demostration:** Suppose for contradiction that there are two $S$-sparse signals $s_1$ and $s_2$ such that $\Theta_\Omega s_1 = \Theta_\Omega s_2 = y$. Then, let $h$ be such that $h = s_1 - s_2$. It is clear that $h$ is $2S$-sparse and that

$$\Theta_\Omega h = \Theta_\Omega (s_1 - s_2) = \Theta_\Omega s_1 - \Theta_\Omega s_2 = 0.$$

The RIP states that

$$(1 - \delta_{2S}) \|h\|^2 \leq \|\Theta_{\Omega T} y\|^2 = 0$$

Since $\delta_{2S} < 1$, $(1 - \delta_{2S}) > 0$ and, therefore we must have $\|h\|^2 = 0$ contradicting the hypothesis that $s_1$ and $s_2$ were distinct. □

We should point out that these results are general in the sense that they are not considering that the recovery algorithm is based on the $l_1$ norm.

### E.2.3 Result for Basic CS

**Theorem 4** ([8, 9])**.** *Let $s$ be an $S$-sparse signal supported on $T$ and measured by $\Theta_\Omega$. Assume that the restricted isometry constant for the matrix $\Theta_{\Omega T}$ is such that $\delta_{2S} < \sqrt{2} - 1$. Then the solution $\hat{s}$ to*

$$\hat{s} = \min_{s^*} \|s^*\|_{l_1} \quad \textit{subject to} \quad \Theta s^* = y$$

*is exact, i.e., $\hat{s} = s$.*

This result is deterministic, not involving a non-zero probability of failure and is also universal in the sense that all sufficiently sparse vectors are exactly reconstructed from $\Theta_\Omega s$.

We can interpret this result as a slightly stronger condition that is related to the $l_1$ norm reconstruction strategy. In fact, it can be shown that for

- $\delta_{2S} < 1$ solution to the $l_0$ norm is unique; and
- $\delta_{2S} < \sqrt{2} - 1$ solution to the $l_0$ norm and the $l_1$ are unique and the same.

## E.3 Robust CS

Most signals are not usually sparse; they can be approximately sparse or have an exponential decay. Moreover, measurements are not usually perfect and some level of noise is added to them. For CS to be suitable for real application it must be robust to these kinds of inaccuracies. Therefore, a lot of effort was made to set conditions and theorems to expand the CS theory.

In this section, we will present theorems that make CS robust to applications when:

- the signal is not exactly sparse; or
- measurements are corrupted by noise.

### E.3.1 Signals that are not Exactly Sparse

In general we cannot assume that images are sparse in a specific domain. However, they are compressible in the sense that, after the DCT or Wavelet transform, the coefficient decay rapidly, typically like a power law.

In this case, if $x$ if an image, $s = \Psi x$ is only approximately sparse, and, therefore, we denote by $s_S$ the best $S$-sparse approximation of $s$, i.e., the result obtained when we force the $N - S$ smallest coefficients of $s$ to be zero.

The following theorem evaluates the performance of CS in this scenario.

**Theorem 5** ([9]). *Assume that $s$ is approximately sparse and let $s_S$ be as defined above. Then if $\delta_{2S} < \sqrt{2} - 1$, the solution $\hat{s}$ to*

$$\hat{s} = \min_{s^*} \|s^*\|_{l_1} \quad \textit{subject to} \quad \Theta_\Omega s^* = y$$

*obeys*

$$\|\hat{s} - s\|_{l_1} \leq C \cdot \|\hat{s} - s_S\|_{l_1}$$

*and*

$$\|\hat{s} - s\|_{l_2} \leq C_0 s^{-1/2} \cdot \|\hat{s} - s_S\|_{l_1}$$

*for reasonable values of the constant $C_0$.*

Roughly speaking, the theorem says that CS recovers the $S$ largest entries of $s$. Notice that, in the particular case when $s$ is S-sparse, $\|\hat{s} - s_S\| = 0$ and the recovery is exact.

This result has the following desired properties:

- it is a deterministic statement and there is no probability of failure;
- it is universal in that it holds for all signals; and
- it holds for a wide range of values of $S$.

Again, the demonstration of the above theorem does not lead us to the objective of this section and, therefore, will not be presented here. For the interested reader, we recommend [9, 28].

### E.3.2 Signals that are Corrupted by Noise

Another very import and realistic scenario to consider is when the acquired data is corrupted with noise, i.e.,

$$y = \Phi x + n$$

where $n$ is an unknown noise contribution bounded by a known amount $\|n\|_{l_2} \leq \epsilon$.

The property that will allow the method to be applicable is *stability* [28]: small changes in the observations should result in small changes in recovery. Hence, considering the undersampling problem, the best result we can hope for is a reconstruction error proportional to $\epsilon$.

**Demostration:** [28] Consider the best possible condition in which we know *a priori* the support $T$ of $s_S$. In this case, we can reconstruct $\hat{s}$ by a Least-Squares method, for example:

$$\hat{s} = \begin{cases} (\Theta_{\Omega T}^* \Theta_{\Omega T})^{-1} \Theta_{\Omega T}^* y & \text{on } T \\ 0 & \text{elsewhere} \end{cases}$$

and suppose that no other method would exhibit a fundamentally better performance. Therefore,

$$\hat{s} - s_S = (\Theta_{\Omega T}^* \Theta_{\Omega T})^{-1} \Theta_{\Omega T}^* n$$

and if the eigenvalues of $\Theta_{\Omega T}^* \Theta_{\Omega T}$ are well behaved, then

$$\|\hat{s} - s_S\|_{l_2} \approx \|\Theta_{\Omega T} n\|_{l_2} \approx \epsilon.$$

□

Therefore, the result we are searching for is a bounding for $\Theta$ that guarantees that the reconstructed $\hat{s}$ obeys

$$\|\hat{s} - s_S\|_{l_2} \leq C_1 \epsilon \tag{E.1}$$

for a rather small constant $C_1$.

This can be achieved by minimizing the $l_1$ norm and considering the constraint $\|\Theta_\Omega s - y\| \leq \epsilon$

**Theorem 6** ([9]). *Assume that $y = \Theta_\Omega s + n$ where $\|n\|_{l_2} \leq \epsilon$. Then if $\delta_{2S} < \sqrt{2} - 1$, the solution $\hat{s}$ to*

$$\hat{s} = \min_s \|s\|_{l_1} \quad \textit{subject to} \quad \|\Theta_\Omega s - y\|_{l_2} \leq \epsilon$$

*obeys*

$$\|\hat{s} - s\|_{l_2} \leq C_0 s^{-1/2} \cdot \|\hat{s} - s_S\|_{l_1} + C_1 \epsilon$$

*for reasonable values of the constant $C_0$ and $C_1$.*

It is noteworthy that the reconstruction error is a superposition of two factors: the errors that yield from sparsity approximation and the error that results from the additive noise.

For the reader interested in the proofs of Theorems 5 and 6 we recommend [6, 9].

## E.4 Design of Efficient Sensing Matrices

It is, of course, of great importance to have matrices that preserve the RIP. Given a sensing matrix $\Phi$, the calculus of the associated restricted isometry constant is NP hard and thus testing this property at each acquisition is unfeasible. We can, however, determine some measurement ensembles where the RIP holds.

The actual problem is to design a fat sensing matrix $\Theta_\Omega$, so that any subset of columns of size $S$ be approximately orthogonal. Here, randomness re-enters the picture because setting a deterministic $\Theta_\Omega$ may be a very difficult task (especially considering large values of $S$ ), but it can be easily shown [6] that trivial random structures perform quite well.

Interestingly, the high dimensionality of the usually handled signals also gives a positive contribution. It can be shown [29] that if $N$ is large, a small set of randomly selected vectors in $\mathbb{R}^N$ will be approximately orthogonal.

The following results obtained by [6, 28] provide several examples of matrices that obey RIP.

**Theorem 7** (Gaussian Matrices)**.** *Let the entries of $\Theta_\Omega$ be i.i.d., Gaussian with mean zero and variance $1/N$. Then the RIP holds with overwhelming probability if*

$$S \leq C \cdot N/\log(M/N)$$

*for a relatively small constant $C$.*

**Theorem 8** (Random Projections)**.** *Let $\Theta_\Omega$ be a random Gaussian matrix whose rows were orthonormalized. Then the RIP holds with overwhelming probability if*

$$S \leq C \cdot N/\log(M/N)$$

*for a relatively small constant $C$.*

A measurement using this matrix involves projecting the signal on an orthogonal subspace which was chosen uniformly at random. Notice that the result of Theorem 7 is the same as Theorem 8 because, essentially, we have the same Gaussian matrix.

**Theorem 9** (Binary Matrices)**.** *Let the entries of $\Theta_\Omega$ be independent taking values $\pm 1/\sqrt{N}$ with equal probability. Then the RIP holds with overwhelming probability if*

$$S \leq C \cdot N/\log(M/N)$$

*for a relatively small constant $C$.*

This case is also very similar to Theorem 7. However, it measures the correlation between the signal and random sign sequences instead of the correlation between the signal and white noise.

Theorems 7, 8 and 9 can be extended to several other distributions, but we will not present them here. Instead, we will focus on a much stronger result.

**Theorem 10** (General Orthogonal Measurement Ensembles)**.** *Let $\Theta$ be an orthogonal matrix and $\Theta_\Omega$ be obtain by selecting $M$ rows from $\Theta$ uniformly at random. Then the RIP holds with overwhelming probability if*

$$S \leq C \cdot \frac{1}{\mu^2} \cdot \frac{N}{(\log M)^6}$$

*for a relatively small constant $C$.*

Theorem 10 is very significant because, as we have mentioned before, in many applications the signal is not sparse in the time domain, but rather in a fixed orthonormal basis $\Psi$. Therefore, this theorem guaranties that if we can determine an orthogonal matrix $\Phi$ such that $\mu(\Phi, \Psi)$ is small[3], then recovery is exact when the measurements are taken with $\Phi_{\Omega}$.

This result is not trivial and certainly not optimal, but researchers have been unable to improve it up until now [15].

[3] This is equivalent to setting $\Theta = \Phi\Psi^*$ and forcing $\mu(\Theta)$ to be small.

# Appendix F

# Results

In this appendix we will verify CS theory by means of examples.

Though the images are already stored in the computer as a matrix of pixels, we will simulate acquisition by means of measurements that involve linear combinations of these coefficients.

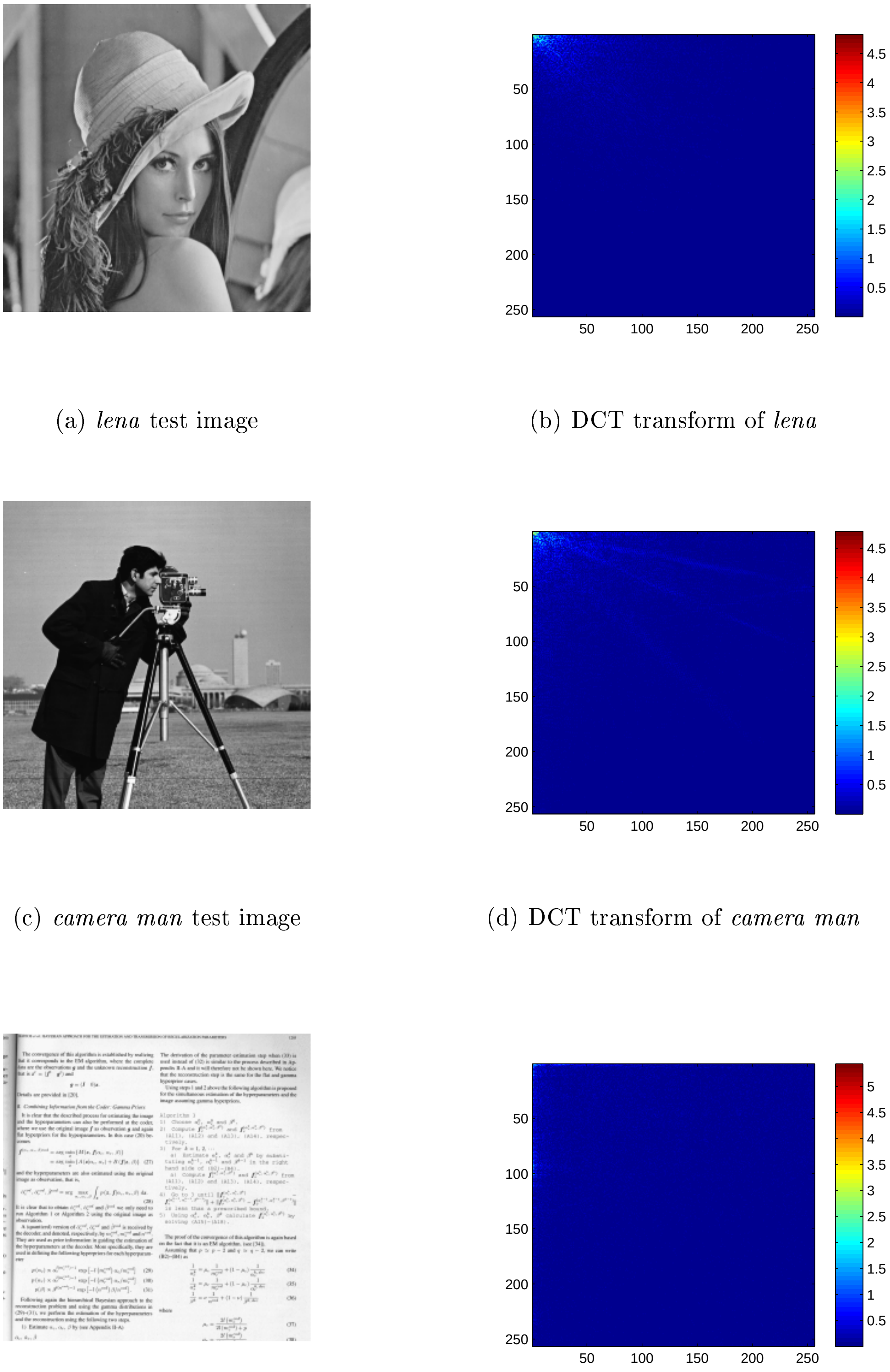
The different acquisition approaches will be evaluated in terms of their peak signal to noise ratios (PSNR) for different amounts of measurements, $M$.

The procedure was based on the results obtained by [10] and the optimization algorithms used were downloaded from http://www.acm.caltech.edu/l1magic [11].

## F.1 Images that are Sparse in the DCT Domain

To explore CS, we used three different images of size $N = 256 \times 256 = 65536$: *lena*, *camera man* and *text*, which differ in terms of energy distribution in the DCT domain. From Figure F.1 we observe that while *lena* is smoother, the highest DCT coefficients appear on the upper left matrix corner. On the other hand, since *text* is an image with abrupt intensity variations, its energy is spread along almost all the DCT basis. Middling, *camera man* has an intermediate energy spread, displaying strong intensities at some DCT diagonals which correspond to the sharp image lines.

To evaluate applications on image compression for Basic CS, it was necessary to force sparsity in the DCT representation of the images. Therefore, for $S = 3.5k$, $6k$,



(a) *lena* test image

(b) DCT transform of *lena*

(c) *camera man* test image

(d) DCT transform of *camera man*

(e) *text* test image

(f) DCT transform of *text*

Figure F.1: Test images and their DCT transform.

$10k$, and $14k$ (where $k = 10^3$) we selected the $N - S$ smallest DCT coefficients of each image and set them to zero. Figures F.2, F.3, and F.4 illustrate the distribution in DCT domain of the non-zero values. [1]

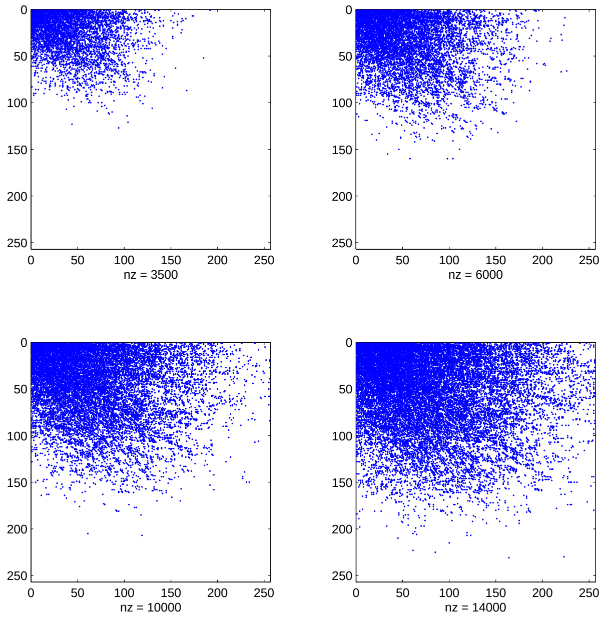


Figure F.2: Visualization of sparsity pattern in test image *lena*.

### F.1.1 Acquisition Strategies

We considered the following acquisition strategies[2]:

1. Linear DCT measurements followed by inverse transform recovery;

2. Random Noiselet measurements followed by $l_1$ minimization recovery; and;

[1] Notation: we will continue to use the variables $s$ and $x$ to denote representations in the time and DCT domain, respectively; and we will use $x_0$ and $s_0$ to refer to the former descriptions after sparsity is forced.

[2] Let $x_0$ be the image represented in terms of a vector of pixels of size $65536 \times 1$ and $s_0$ be the vector that represents the 2D-DCT transform of the image. Since the sparse domain is the DCT, in the current notation, $\Psi$ is the 2D-DCT transform, but the measurement matrix $\Phi_\Omega$ is chosen in different fashions. We will denote $\Phi_{\Omega j}$ the measurement matrix that is applied in strategy $j$ and $\hat{x}_j$ and $\hat{s}_j$ the recovered signals.

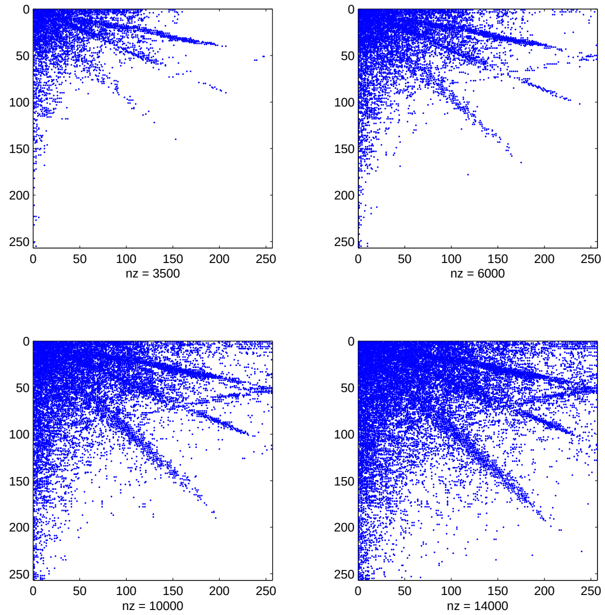


Figure F.3: Visualization of sparsity pattern in test image *camera man*.

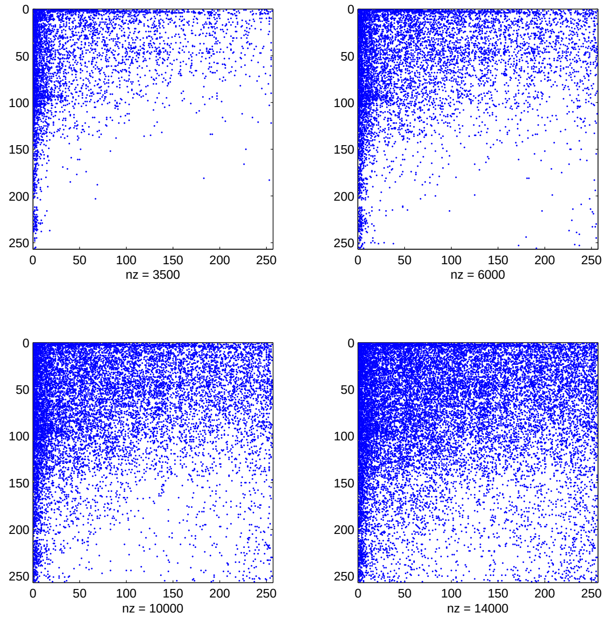


Figure F.4: Visualization of sparsity pattern in test image *text*.

3. $1k$ linear DCT and $M - 1k$ random Noiselet measurements followed by $l_1$ minimization recovery.

In the first strategy, we implement a linear DCT compression scheme to which we will compare CS results. Measurements are taken by obtaining the first $M$ DCT coefficients (according to the diagonal zigzag scanning pattern described in Section B.5.1). Therefore $\Phi_1$ is a fat matrix created by stacking the rows of $\Psi$ that correspond to the linear DCT measurements and recovery is done by setting to zero the unknown values and then applying the inverse DCT transform.

The other two approaches are classic compressive sampling schemes. In the second procedure, the sensing matrix $\Phi_{\Omega 2}$ is built by choosing at random $M$ waveforms of an $N \times N$ Noiselet transform $\Phi$[3]. Therefore, measurements are taken as

$$y_2 = \Phi_{\Omega 2} x_0$$

and the following linear program is solved

$$\hat{s}_2 = \min_s \|s\|_{l_1} \quad \text{subject to} \quad y_2 = \Phi_{\Omega 2} \Psi^* s \tag{F.1}$$

From $\hat{s}_2$, we can reconstruct $\hat{x}_2$ as

$$\hat{x}_2 = \Psi^* \hat{s}_2$$

The third method is analogous to the second one, only differing in terms of the sensing matrix. This time, $\Phi_{\Omega 3}$ is constructed by staking the first thousand linear

[3] This transform is described in [30] and we used the `MATLAB` code downloaded from http://users.ece.gatech.edu/∽justin/spmag to generate it [10]. This transform was chosen not only because it is highly incoherent with the DCT and Wavelets but also because the matrix created is orthogonal and self-adjoint, thus being easy to manipulate. Below one can see an illustration of $\Phi$ for $N = 4$.

$$\Phi = \frac{1}{2} \cdot \begin{bmatrix} 1 & -1 & 1 & 1 \\ -1 & 1 & 1 & 1 \\ 1 & 1 & -1 & 1 \\ 1 & 1 & 1 & -1 \end{bmatrix}$$

DCT coefficients (i.e., the most important ones according to the zigzag scanning pattern) and $M - 1k$ Noiselet waveform (chosen at random from the same $N \times N$ Noiselet transform used in strategy 2).

**Computational Errors**

Due to computational errors, we were unable to use the function that solves Equation F.1 in the `l1- Magic` package. Instead, we solved the convex optimization problem, also included in `l1-Magic`,

$$\hat{s} = \min_s \|s\|_{l_1} \quad \text{subject to} \quad \|y - \Theta_\Omega s\|_{l_2} \leq \epsilon \tag{F.2}$$

for $\epsilon = 10^{-3}\|y\|_{l_2}$.

It is noteworthy that, by changing the parameter $\epsilon$, different results can be obtained and, therefore, a further analysis of this problem should involve testing recovery for varying values of $\epsilon$. However, due to computational efforts [4], it would be impossible to do this in the present project without jeopardizing other interesting analyses of CS theory. From a few tests, however, we were able to induce that, as we minimize $\epsilon$, the results improve in the sense that higher PSNRs are reached, but the curve format stays the same[5].

### F.1.2 Results

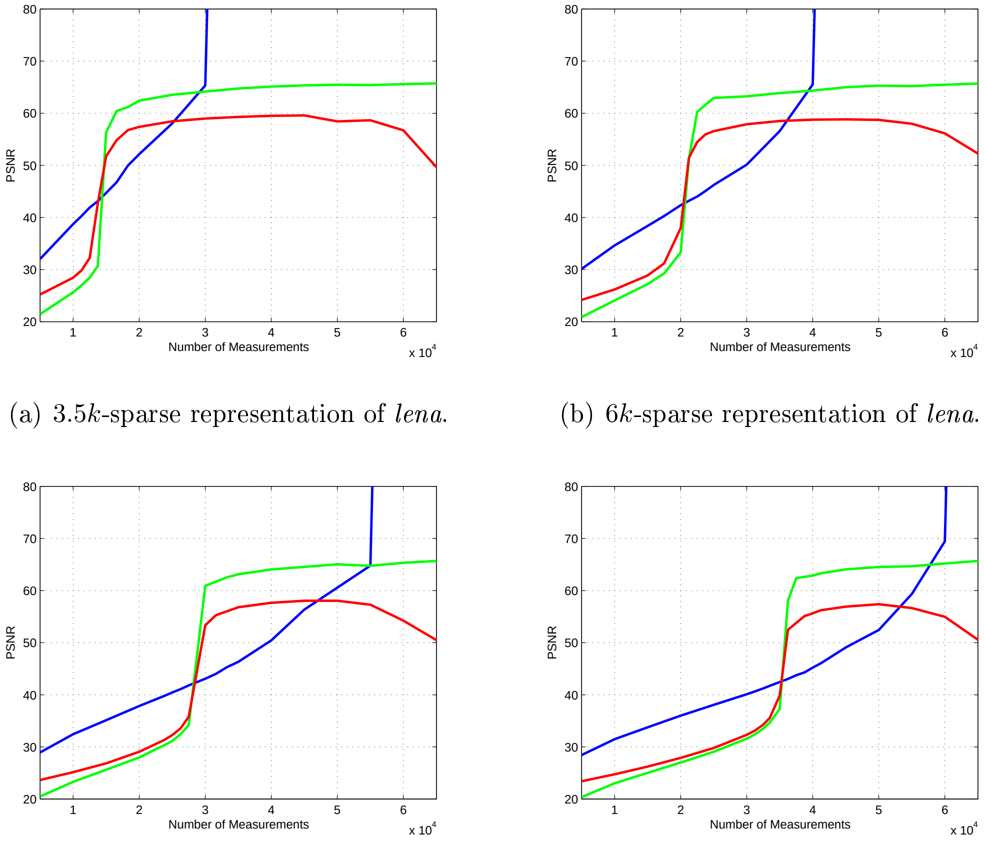
Figures F.5, F.6 and F.7 show the results obtained for each image, respectively.

### F.1.3 Analysis

The first meaningful characteristic that we observe is that compressive sampling routines start to have good performances after a specific number of measurements are taken. This threshold can be associated with the number of samples set by

[4] The necessary calculations executed for each of the 222 considered measurements involve approximately 2 hours of computational processing.

[5] This observation refers to the case where the signal is sparse and the measurements are uncorrupted by noise. Latter in this appendix, we will discuss how $\epsilon$ behaves in other scenarios.



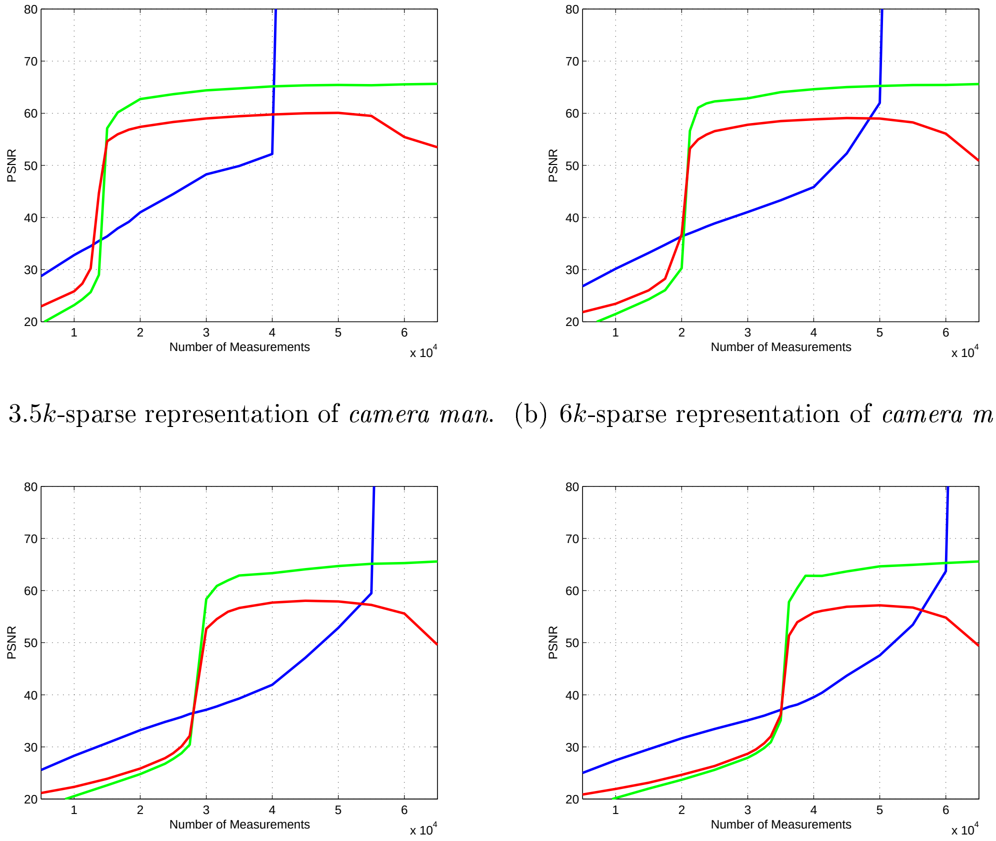
(a) $3.5k$-sparse representation of *lena*.

(b) $6k$-sparse representation of *lena*.

(c) $10k$-sparse representation of *lena*.

(d) $14k$-sparse representation of *lena*.

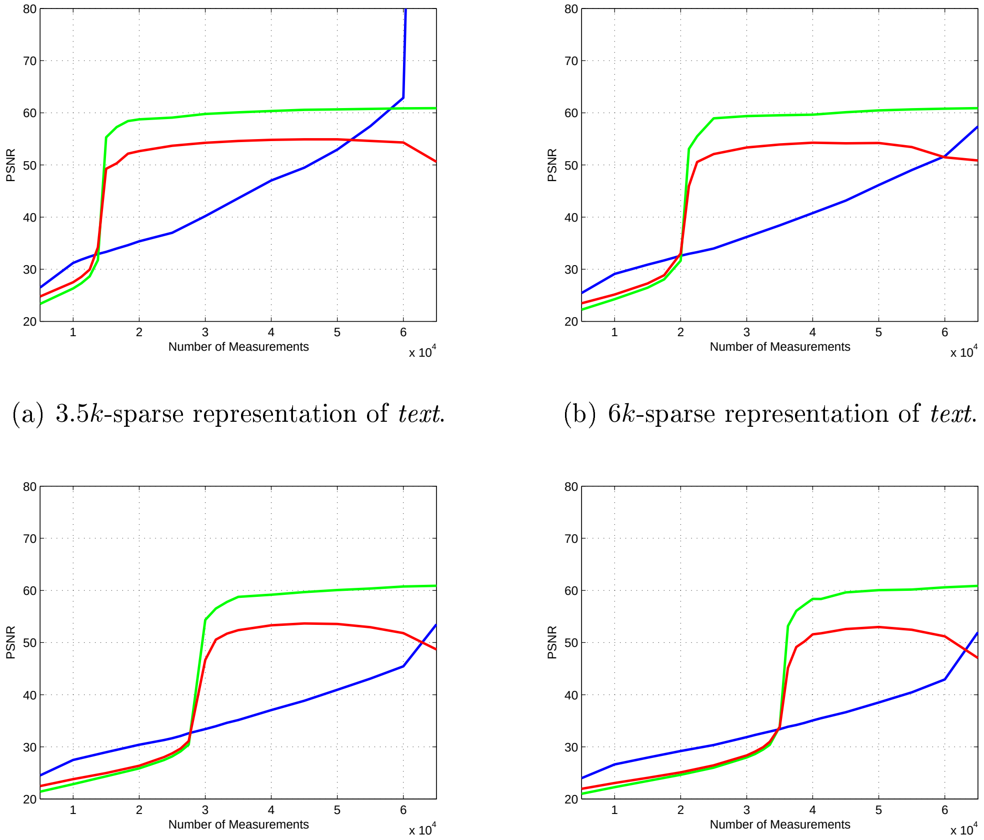
Figure F.5: Results for applications of CS scheme in sparse versions of image *lena*. In each graph is shown the PSNR (peak signal-to-noise ratio between the sparse version of the image and the compressed reconstruction) versus the number of measurements: (blue) linear DCT acquisition, (green) CS using only Noiselet measurements, and (red) CS using Noiselet measurements and the first thousand linear DCT coefficients.



(a) 3.5$k$-sparse representation of *camera man*. (b) 6$k$-sparse representation of *camera man*.

(c) 10$k$-sparse representation of *camera man*. (d) 14$k$-sparse representation of *camera man*.

Figure F.6: Results for applications of CS scheme in sparse versions of image *camera man*. In each graph is shown the PSNR (peak signal-to-noise ratio between the sparse version of the image and the compressed reconstruction) versus the number of measurements: (blue) linear DCT acquisition, (green) CS using only Noiselet measurements, and (red) CS using Noiselet measurements and the first thousand linear DCT coefficients.



(a) $3.5k$-sparse representation of *text*.

(b) $6k$-sparse representation of *text*.

(c) $10k$-sparse representation of *text*.

(d) $14k$-sparse representation of *text*.

Figure F.7: Results for applications of CS scheme in sparse versions of image *text*. In each graph is shown the PSNR (peak signal-to-noise ratio between the sparse version of the image and the compressed reconstruction) versus the number of measurements: (blue) linear DCT acquisition, (green) CS using only Noiselet measurements, and (red) CS using Noiselet measurements and the first thousand linear DCT coefficients.

Theorem 2. Notice that this borderline depends linearly on the sparsity of the signal. Since we imposed the same level of sparsity for all three cases, it is not surprising that the threshold does not differ between the test images.

Linear DCT acquisition, however, is best in *lena* and worst in *text*. This was already expected since *lena* has the best energy concentration along the DCT coefficients.

We calculated the coherence by

$$\mu(\Theta) = \sqrt{N} \max_{i,j} |\Theta_{i,j}|$$

and obtained $\mu(\Theta_2) = 2.82$, while $\mu(\Theta_3) = \sqrt{N} = 256$.

Therefore, although the threshold for strategies 1 and 2 are essentially the same, $\mu(\Theta_3)$ is almost a hundred times larger than $\mu(\Theta_2)$. This may strike the reader as a contradiction to the tightness of Theorem 2. Notice, however, that $\Theta_3$ is not orthogonal and thus the theorem cannot be applied in this particular example.

It is also relevant to point out that before the boundary, strategy 3 performs better than 2 and this tendency is not sustained when CS theory start to operate. This result can be interpreted by the fact that when taking a small number of samples the knowledge of the low frequency coefficients adds more information to the signal than random measurements. In fact, the best acquisition strategy in this region is the linear DCT.

A last but very important comment is that, although it may seem that for $M$ higher than the threshold $\Phi_{\Omega 2}$ behaves better than $\Phi_{\Omega 3}$, this is not true. We should consider that after the threshold the signal is perfectly reconstructed and what we see are measurement errors. To illustrate this point, we plotted in Figure F.8 the recovery of the $10k$-sparse image *lena* for very small values of $\epsilon$. Notice that the oscillation for high values of $M$ confirm the hypothesis of additional computational errors.

To further convince the reader, Figure F.9 shows the difference between the two acquisition schemes for the $10k$-sparse representation of *lena*. We took $M = 45k$ measurements and reconstructed the image using Equation F.2 with $\epsilon = 10^{-3}\|y\|_{l_2}$.

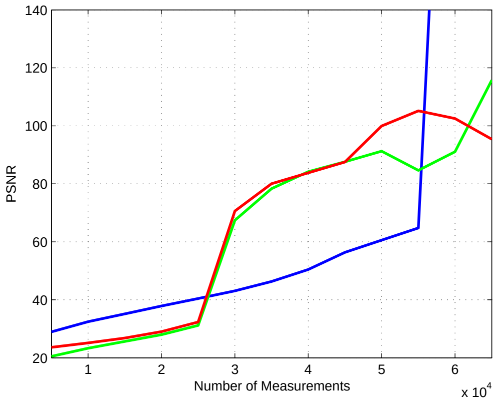


Figure F.8: Recovery of the $10k$-sparse respresentation of *lena* with $\epsilon = 0.001$ for $\Phi_{\Omega 2}$ and $\epsilon = 0.1$ for $\Phi_{\Omega 3}$.

## F.2 Recovery Considering that the Image is not Exactly Sparse

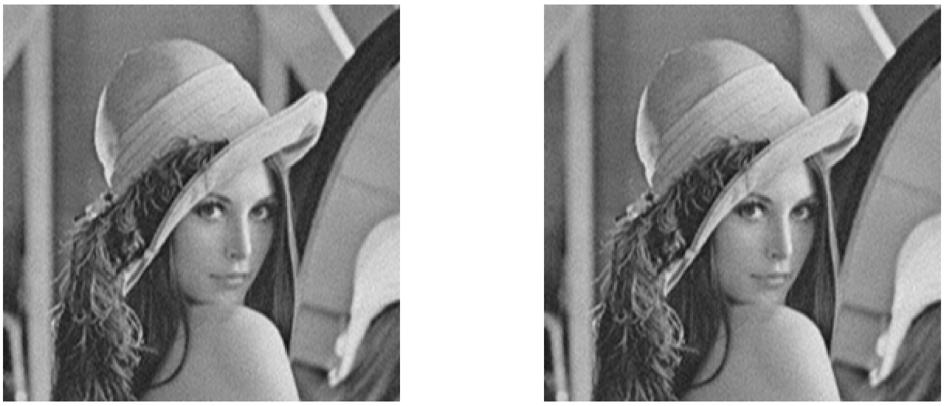
In this section, we repeat the acquisition schemes used in the previous one without imposing sparsity to the test images. The same three strategies are reproduced and the results are shown in Figure F.10.

In this case, we also used Equation F.2 instead of Equation F.1 and $\epsilon = 10^{-3}\|y\|_{l_2}$. A few tests were made varying $\epsilon$ but, differently from the sparse example, minimizing $\epsilon$ lead to no significant improvement. This phenomenon can be explained by the distortion provoked by the absence of sparsity which overcomes the computational errors, making the adjustment of $\epsilon$ ineffective.

### F.2.1 Analysis

From the results, we conclude that CS performs very poorly when we do not force sparsity to the input image, in fact it is worse than the linear DCT compression scheme, even considering large number of measurements. The explanation to this disappointing outcome is that the images are not sparse in the DCT domain.

An inadvertent reader, by coming across Figures F.5, F.6, F.7, and F.10, could easily conclude that, for the same $M$, we achieve better results if we force sparsity



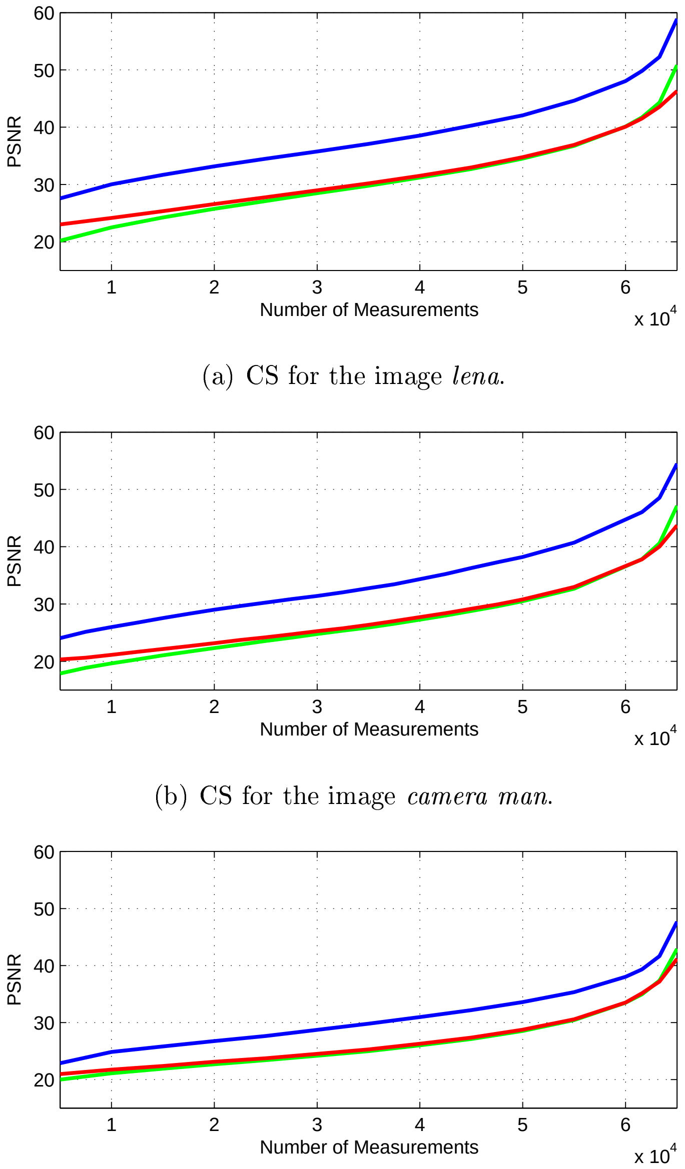
(a) measuring $y = \Phi_{\Omega 2} x_0$ (b) measuring $y = \Phi_{\Omega 3} x_0$.

Figure F.9: Test image *lena* reconstructed from $M = 45k$ measurements.

before applying CS (contradicting Theorem 5 that states that if we need $M$ measurements to recover an $S$-sparse signal, then if the signal is not sparse, we would recover the $S$ largest coefficients with this number of samples).

Notice, however, that to develop Figures F.5, F.6, and F.7 we calculated PSNR by comparing the recovered data and the *sparse representation* of the original image. Were we to compare results from Section F.1 with the original test images, we would conclude that, save a relatively small error, the maximum PSNR in Figures F.5, F.6, and F.7 is the same one obtained by taking the number of measurements set by the threshold in those Figures and sensing the original image.

Figures F.11, F.12, and F.13 formalize this observation. From F.11(b), we can make out that $20k$ measurements are needed to recover the $3.5k$-sparse representation of *lena* and, therefore, Theorem 5 guaranties that $20k$ measurements recover the $3.5k$ most significant coefficients of the original image. Notice that, compared to the original image, the reconstruction of the $3.5k$-sparse representation results in PSNR $= 28.8$ and the reconstruction of the original image when $20k$ measurements are taken results in PSNR $= 26.6$, as shown in F.11(b). The same analysis can be made on the other figures and Table F.1 compares the different PSNR calculated when we compare, to the original image, the results obtained when sparsity is or

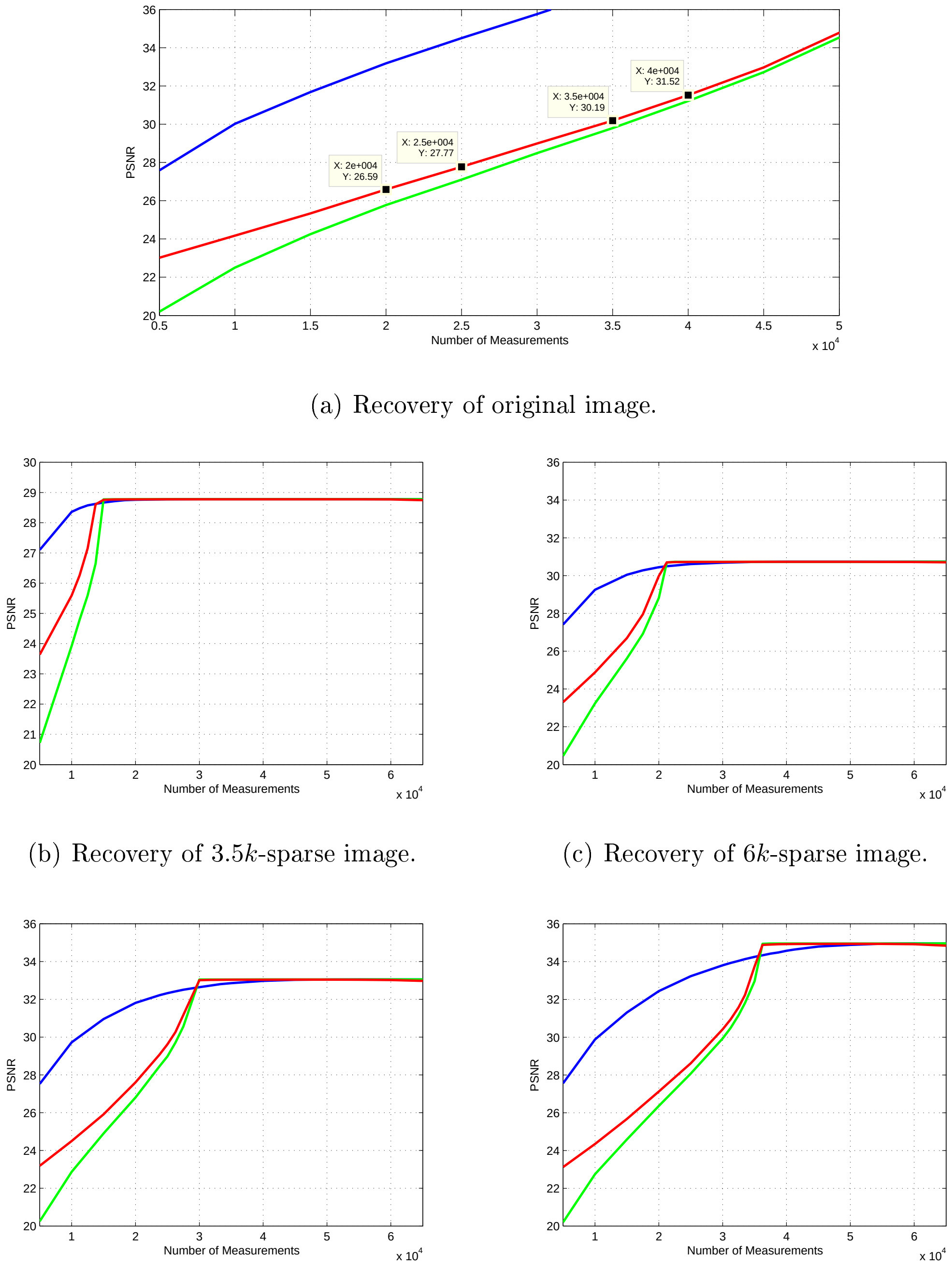


(a) CS for the image *lena*.

(b) CS for the image *camera man*.

(c) CS for the image *text*.

Figure F.10: Results for aplications of CS scheme in the original (only approximately sparse) version of images *lena*, *camera man*, and *text*. In each graph is shown the PSNR versus the number of measurements: (blue) linear DCT acquisition, (green) CS using only noiselet measurements, and (red) CS using noiselet measurements and the first thousand linear DCT coefficients.



(a) Recovery of original image.

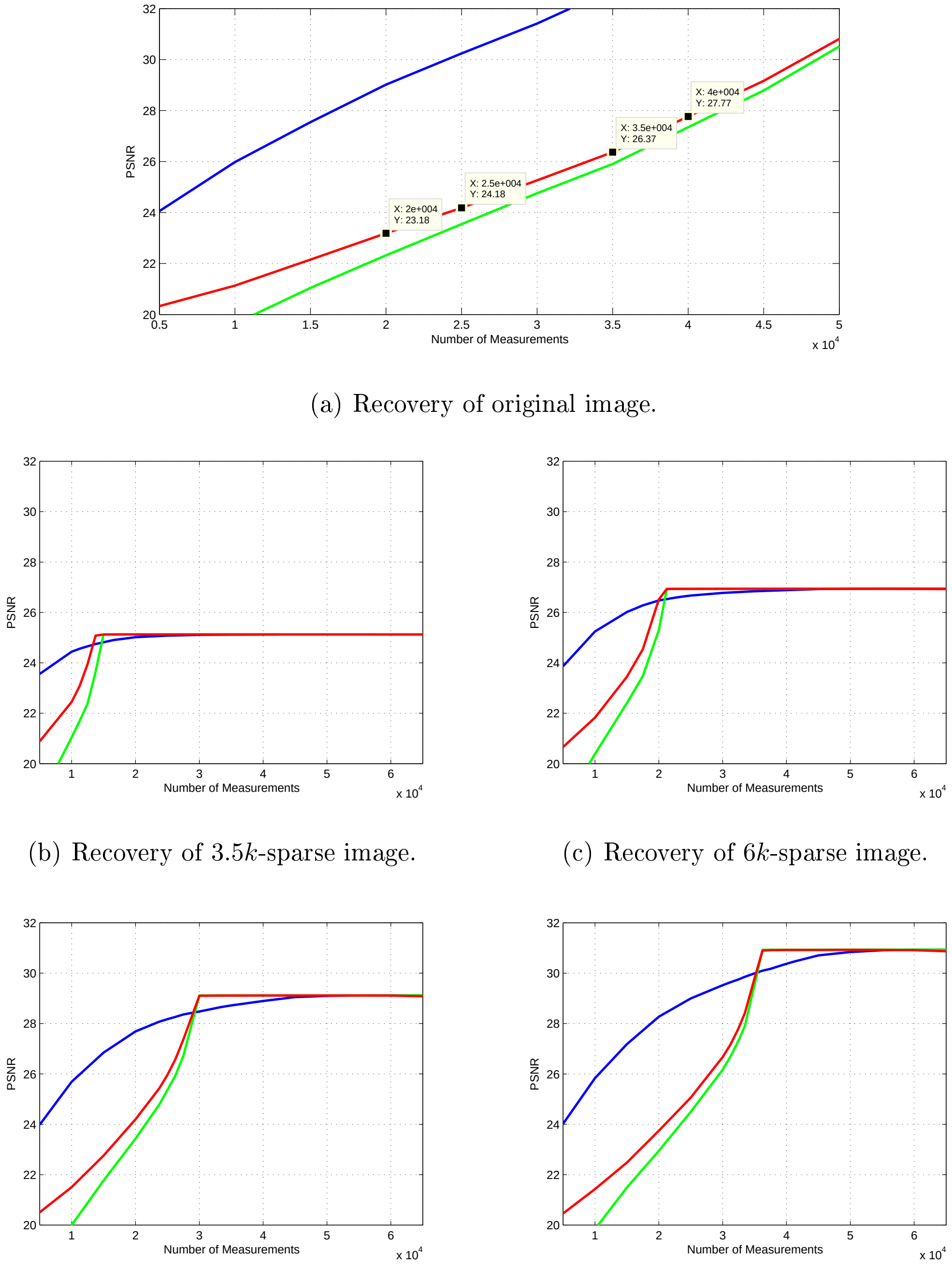
(b) Recovery of 3.5$k$-sparse image.

(c) Recovery of 6$k$-sparse image.

(d) Recovery of 10$k$-sparse image.

(e) Recovery of 14$k$-sparse image.

Figure F.11: Comparing CS acquisition when forcing or not sparsity to the input image *lena*; in all images PSNR is calculated by comparing the reconstructed image with the *original* test image.



(a) Recovery of original image.

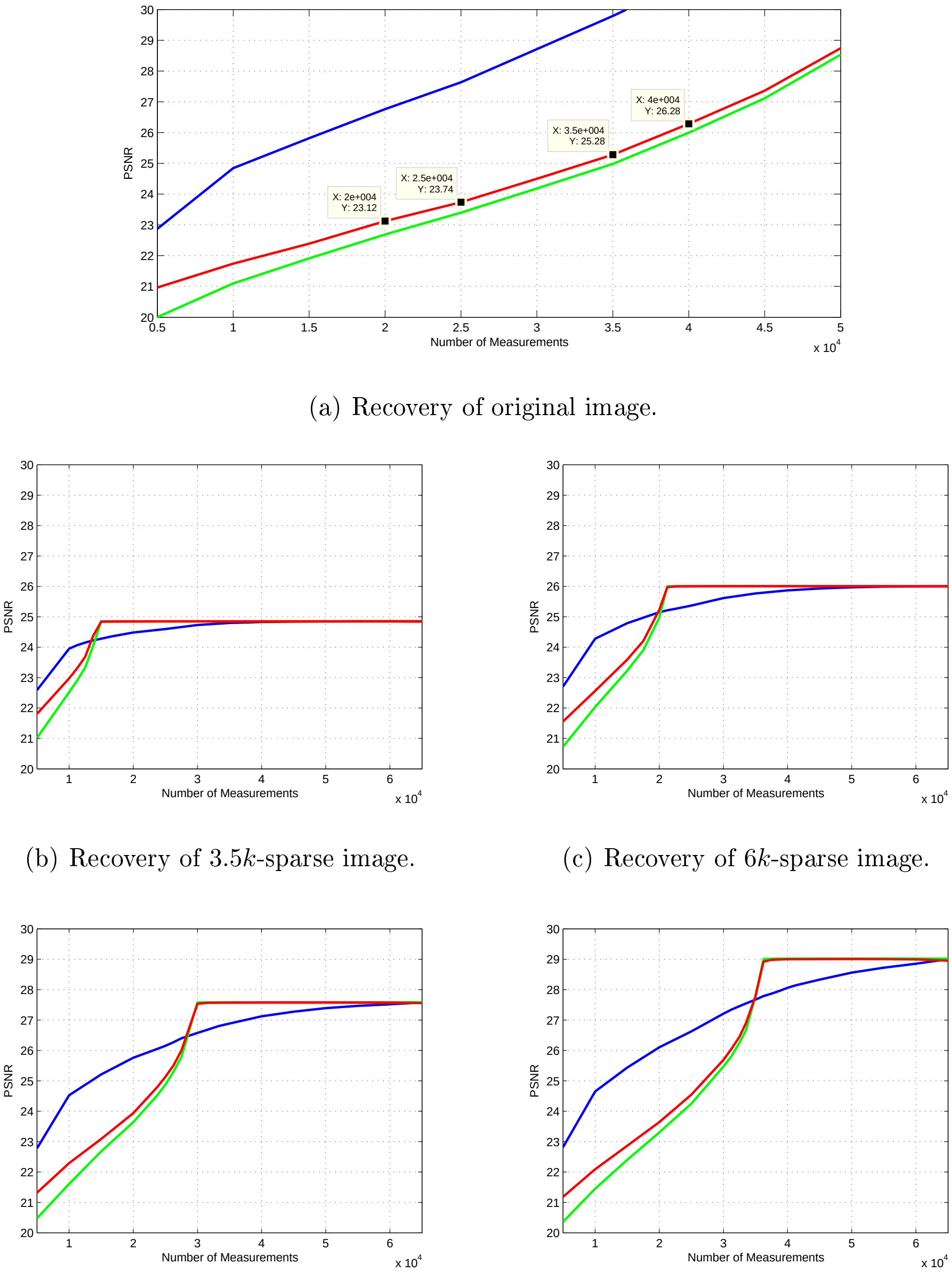
(b) Recovery of 3.5$k$-sparse image.

(c) Recovery of 6$k$-sparse image.

(d) Recovery of 10$k$-sparse image.

(e) Recovery of 14$k$-sparse image.

Figure F.12: Comparing CS acquisition when forcing or not sparsity to the input image *camera man*; in all images PSNR is calculated by comparing the reconstructed image with the *original* test image.



(a) Recovery of original image.

(b) Recovery of 3.5$k$-sparse image.

(c) Recovery of 6$k$-sparse image.

(d) Recovery of 10$k$-sparse image.

(e) Recovery of 14$k$-sparse image.

Figure F.13: Comparing CS acquisition when forcing or not sparsity to the input image *text*; in all images PSNR is calculated by comparing the reconstructed image with the *original* test image.

not forced before CS measurements are applied. The variations can be associated with the constant $C_0$ of Theorem 5.

Table F.1: Different PSNR calculated when we compare, to the original image, the results obtained when sparsity is or not forced before CS measurements are applied.

| Test image *lena* | | |
|---|---|---|
| Measurements | Sparsity is forced | Sparsity is **not** forced |
| $M = 20k$ | PSNR = 28.8 | PSNR = 26.6 |
| $M = 25k$ | PSNR = 30.7 | PSNR = 27.8 |
| $M = 35k$ | PSNR = 33.0 | PSNR = 30.2 |
| $M = 40k$ | PSNR = 34.9 | PSNR = 31.5 |
| Test image *camera man* | | |
| Measurements | Sparsity is forced | Sparsity is **not** forced |
| $M = 20k$ | PSNR = 25.1 | PSNR = 23.2 |
| $M = 25k$ | PSNR = 26.9 | PSNR = 24.2 |
| $M = 35k$ | PSNR = 29.1 | PSNR = 26.4 |
| $M = 40k$ | PSNR = 30.9 | PSNR = 27.8 |
| Test image *text* | | |
| Measurements | Sparsity is forced | Sparsity is **not** forced |
| $M = 20k$ | PSNR = 24.8 | PSNR = 23.1 |
| $M = 25k$ | PSNR = 26.0 | PSNR = 23.7 |
| $M = 35k$ | PSNR = 27.6 | PSNR = 25.3 |
| $M = 40k$ | PSNR = 29.0 | PSNR = 26.3 |

## F.3 The Wavelet Domain

### F.3.1 Signal that are Only Approximately Sparse

Since in the former section we observed that the DCT domain does not assure a proper sparse representation of images, we intent to improve CS recovery by applying a Wavelet transform. To generate the Wavelet basis, we used the `MATLAB` package `WAVELAB` downloaded from http://www-stat.stanford.edu/∼wavelab/. We used an orthornormal Wavelet basis to better relate to the Theorems stated in the previous appendix and to simplify implementation. The Coiflet of 4 vanishing moments was chosen because it performed quite well in the tested images.

The sparsity transform $\Psi$ is, therefore, the Wavelet and the sensing matrix $\Phi$ is built by choosing at random $M$ waveforms of an $N \times N$ Noiselet transform (analogous to the second strategy of Section F.1). The sampled signals are the original images to which we can only guarantee sparsity approximation. Figure F.14 presents the obtained results.

As expected, the result for reconstruction considering that the signal is sparse in the Wavelet domain is much better than for the DCT. It is also interesting to point out that for images *text* and *camera man*, which are less sparse in the DCT domain, the gain in terms of signal to noise ratio is considerably higher.

### F.3.2 Sparse Signals

Considering that substituting DCT for Wavelets resulted in much better performances, we were motivated to analyze Wavelets in the conditions described in Section F.1, i.e., for sparse signals.

Again $\Psi$ is formed by Wavelets and $\Phi$ by Noiselets and the signals are forced to be sparse by setting the smallest coefficients (of the Wavelet domain) to zero. Figures F.15, F.16 and F.17 show the results for each of the three considered images. We plotted the Wavelets on top of the results obtained for the DCT to facilitate analysis.

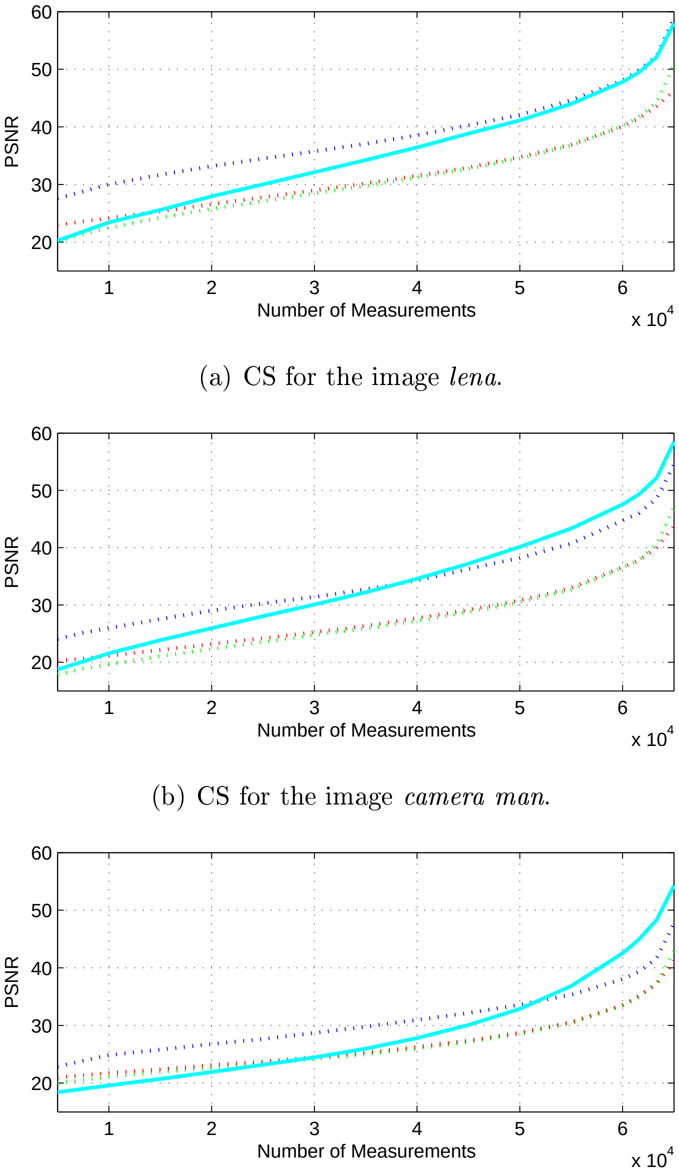
We observe that, in this case, performance is practically the same for both DCT and Wavelet bases. This happens because the signal is as sparse in both domains and the incoherence is very similar: when $\Psi$ is the DCT $\mu(\Theta_1) = 2.82$, while when $\Psi$ is the DWT $\mu(\Theta_1) = 4.81$.

## F.4 Noisy Measurements

In this section, we consider that the acquired measurements are corrupted by an independent white Gaussian noise. We consider the image of *lena*, modified to be $10k$-sparse in the Wavelet domain, and take Noiselet measurements. Therefore, we get

$$y = \Phi_\Omega x_0 + n$$

where $n$ is a random variable with normal distribution and variance $\sigma^2$.

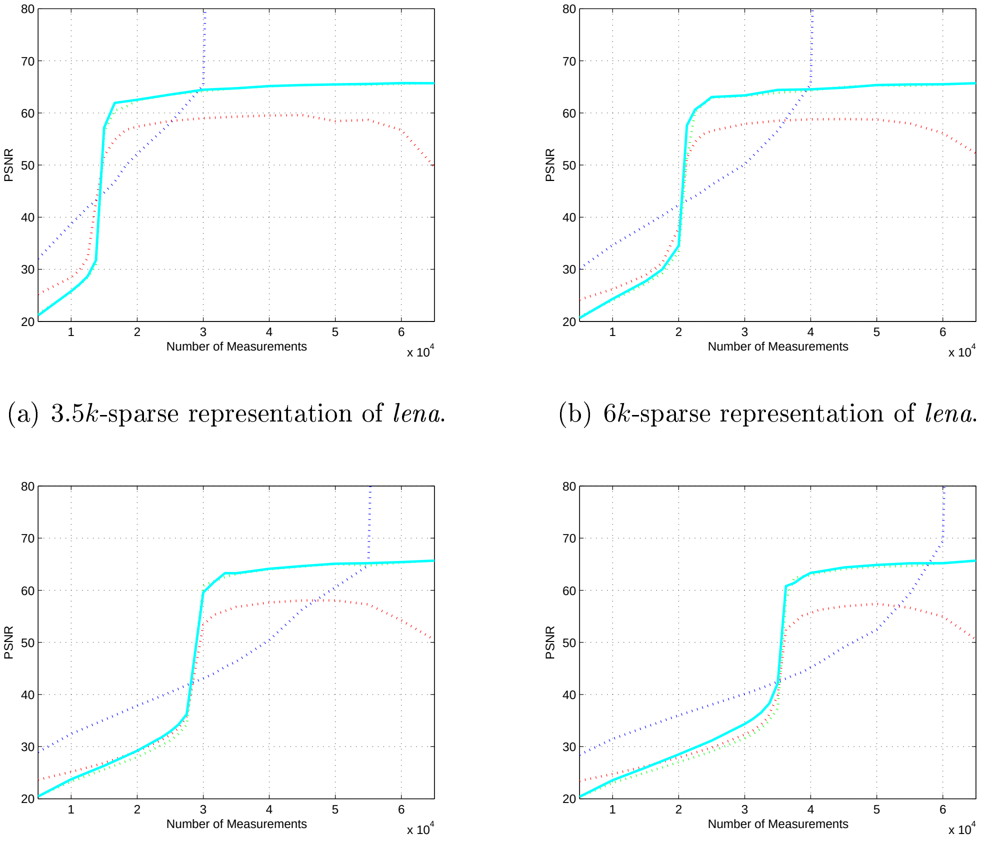


(a) CS for the image *lena*.

(b) CS for the image *camera man*.

(c) CS for the image *text*.

Figure F.14: Results for applications of CS scheme in the original (only approximately sparse) version of images *lena*, *camera man*, and *text*. To assist analysis, we plotted the result for taking Noiselet measurements and assuming that the image is sparse in the Wavelet domain (cian) on top of the results obtained for the DCT in Figure F.10.



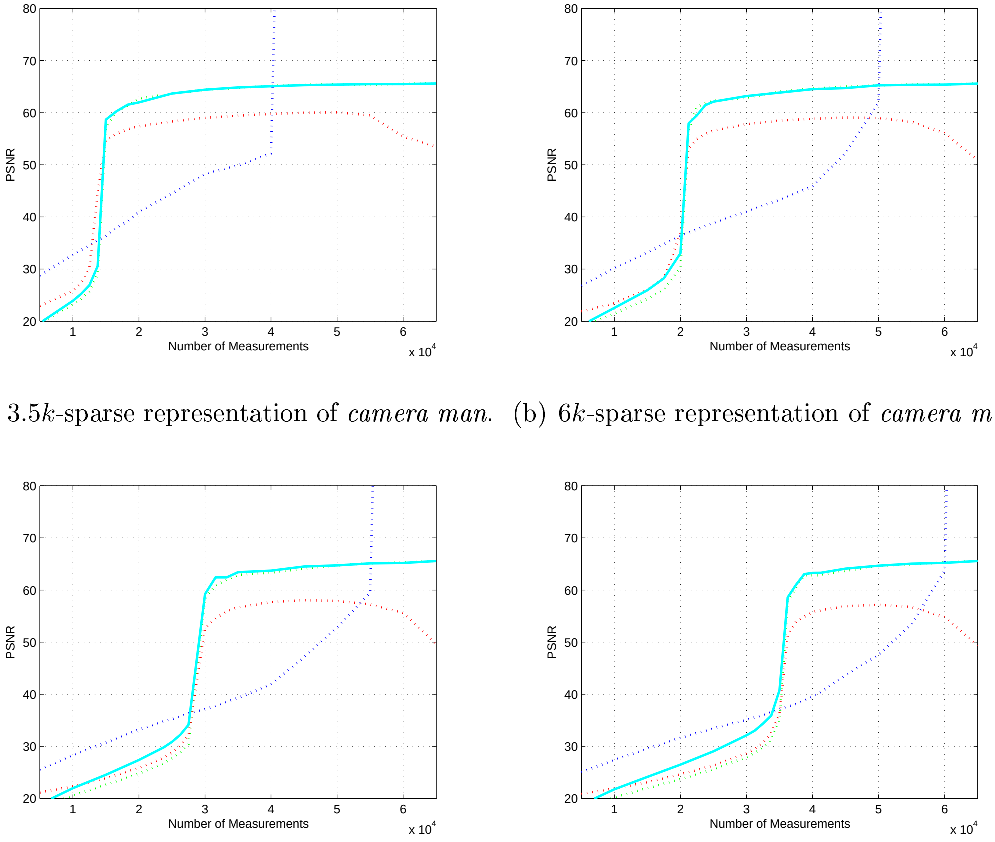
(a) 3.5$k$-sparse representation of *lena*.

(b) 6$k$-sparse representation of *lena*.

(c) 10$k$-sparse representation of *lena*.

(d) 14$k$-sparse representation of *lena*.

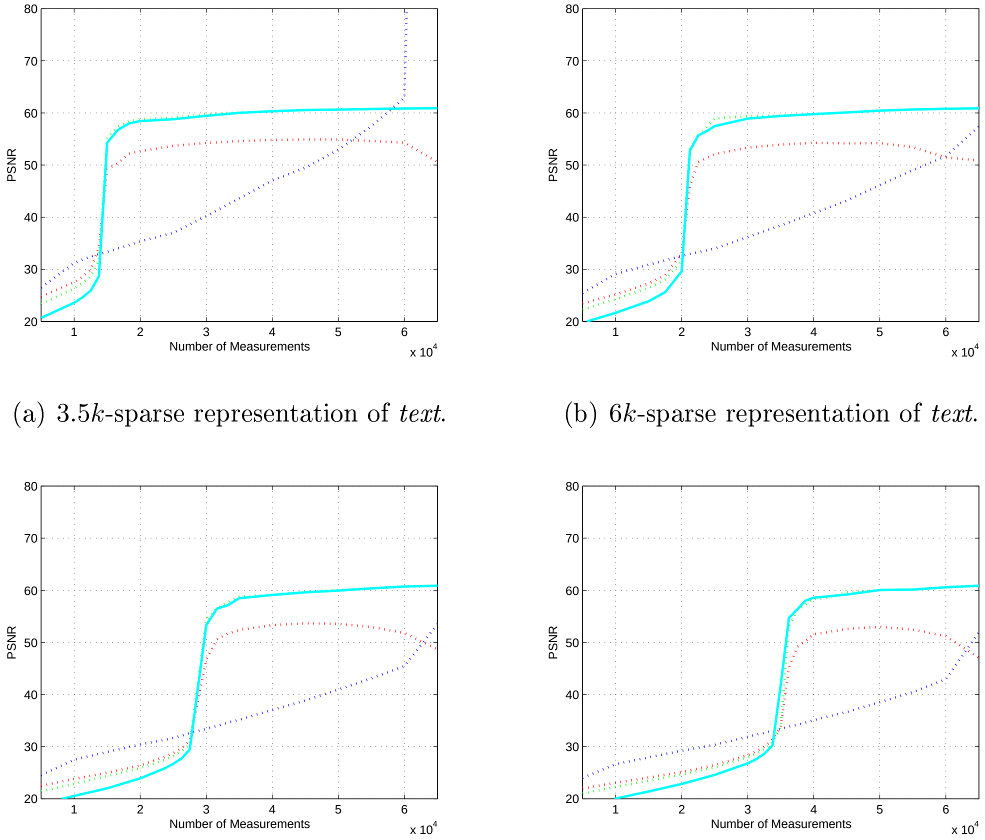
Figure F.15: Results for applications of CS scheme in sparse versions of image *lena*. To assist analysis, we plotted the result for taking Noiselet measurements and assuming that the image is sparse in the Wavelet domain (cian) on top of the results obtained for the DCT in Figure F.5.



(a) $3.5k$-sparse representation of *camera man*. (b) $6k$-sparse representation of *camera man*.

(c) $10k$-sparse representation of *camera man*. (d) $14k$-sparse representation of *camera man*.

Figure F.16: Results for applications of CS scheme in sparse versions of image *camera man*. To assist analysis, we plotted the result for taking Noiselet measurements and assuming that the image is sparse in the Wavelet domain (cian) on top of the results obtained for the DCT in Figure F.6.



(a) 3.5$k$-sparse representation of *text*.

(b) 6$k$-sparse representation of *text*.

(c) 10$k$-sparse representation of *text*.

(d) 14$k$-sparse representation of *text*.

Figure F.17: Results for applications of CS scheme in sparse versions of image *text*. To assist analysis, we plotted the result for taking Noiselet measurements and assuming that the image is sparse in the Wavelet domain (cian) on top of the results obtained for the DCT in Figure F.7.

Figure F.18 shows the result obtained for $\sigma^2 = 0, 0.1, 1, 2, 3, 4, 5, 10$.

In this section, the choice of Equation F.2 is not only due to computational errors, but the theory itself stipulates $\epsilon$ proportional to a bounding of the noise contribution. Hence, given a fixed number of measurements $M$, there is an optimal parameter $\epsilon$ under which the best solution is outside the convex set limitated by the constraints and above which the solution is less exact [6].

The parameter $\epsilon$ was chosen according to a series of experiments and varies according to $\sigma^2$.

## F.5 Quantization

In general, measurements cannot be taken with arbitrary large precision, and a round-off error is added to the acquired data. This quantization process is very important to our study because we are interested in compressing the signal. As seen in Appendix B, the size of the quantization step is extremely relevant to determine the compression rate which, in turn, is used to evaluate compression efficiency based on the rate-distortion criteria.

Unlike the Gaussian noise, the quantization error is deterministic and signal-dependent. Therefore, a great contribution to CS theory consists in verifying how it performs in the presence of quantization errors and, then, plot the Rate × Distortion function.

### F.5.1 Quantization of Sparse Signals

We used a representation of *lena*, forced to be $10k$-sparse in the Wavelet domain. A uniform quantizer of varying step sizes was applied to the Noiselet measurements and reconstruction was implemented based on Equation F.2.

Again, the parameter $\epsilon$ was chosen according to a series of experiments and varies according to the quantization error. To illustrate the calculus of the optimal value for

[6] Note that this is only true for $\sigma^2 \neq 0$. When $\sigma^2 = 0$ the smallest $\epsilon$ the better, the only reason we are not allowed to make $\epsilon = 0$ is due to computational errors

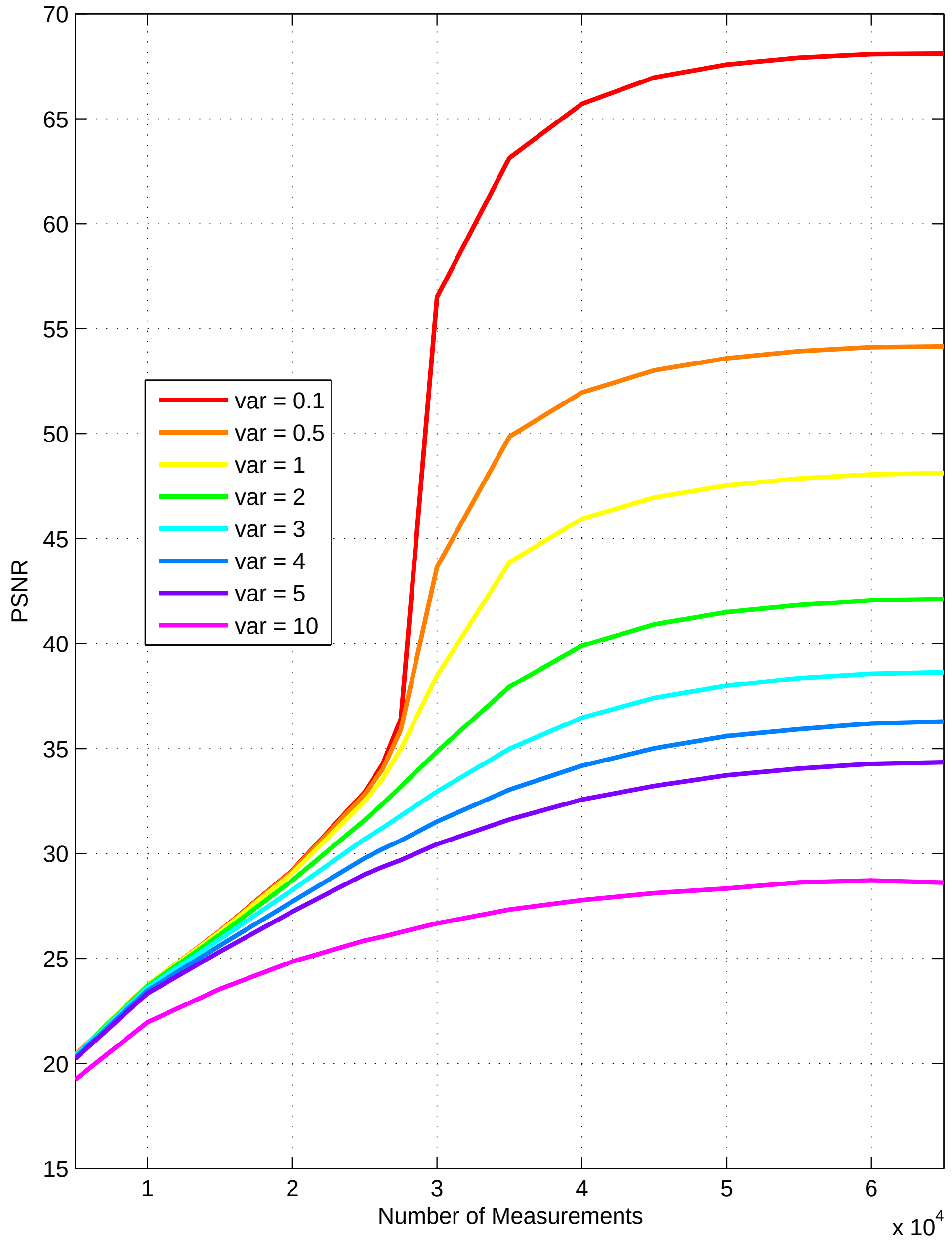


Figure F.18: Results for applications of CS scheme to noisy versions of the $10k$-sparse representation of image *lena*. The strategy involves taking Noiselet measurements and assuming that the image is sparse in the Wavelet domain.

$\epsilon$, which was repeated in the previous and the following sections, we present in Table F.2 the variations of the PSNR according to parameter $\epsilon$ for different quantization steps and a fixed number of measurements, $M = 45k$. We marked the chosen $\epsilon$ for each quantization step.

Notice that the optimal $\epsilon$ increases and diminishes proportionally to the quantization step (that reflects the error size) and that there is an optimal value for each step size, as explained in the previous section. From Table F.2, however, we observe that both of these behaviors as not exact. This is also due to computational errors that are visible since the PSNR variations are small.

For each fixed quantization step we varied the number of measurements and plotted the Rate × PSNR curve, as shown in Figure F.19.

The rate was calculated, as follows

$$Rate = \frac{M \cdot E_y}{256^2}$$

where $E_y$ is the entropy of the measured data $y$.

To calculate $E_y$ we built an histogram based on the minimum and maximum values assumed by $y$ $(y_{\min}, y_{\max})$ and the quantization step, $q_s$. Hence, we obtain a vector $v_y$ of size

$$N = \frac{y_{\max} - y_{\min}}{q_s}$$

where $v_y(n)$ indicates the number of coefficients of $y$ that range between $\{y_{\min} + (n-1)q_s, y_{\min} + nq_s\}$. To disregard the cases when a quantized value does not occur, we add 1 to every coefficient of $v_y$,

$$v'_y(n) = v_y(n) + 1, \forall n \in \{1, 2, \ldots, N\}$$

Hence, the probability of occurrence of each symbol is given by

$$p_y(n) = \frac{v'_y(n)}{\sum_{i=1}^{N} v'_y(i)}$$

and $E_y$ is calculated as in Equation B.2.

We observe a threshold, related to the transition point, where CS theory starts to operate. As we increase the quantization step, the curve approaches the $y$ axis but the threshold diminishes. This was expected because, by boosting quantization effects we minimize rate but create higher distortions.

### F.5.2 Quantization of Signals that are Only Approximately Sparse.

To formally evaluate performance of CS, we have to consider the real case, where quantization errors are added to images that are only approximately sparse.

Therefore, in this section we repeated the acquisition method used in the previous one on the original version of *lena*. Figure F.20 shows the result.

We observe that, since the image is no longer sparse, the threshold for each quantization step vanishes, which was already expected based on the previous results. It is also noteworthy, that there is little efficiency loss when sparsity is not forced. This is because both errors (from lack of sparsity and noise addition) add up and, hence, the former takes smaller importance. This observation also indicates that the Wavelet transform is an appropriate representation of the image.

The rate × distortion curve for CS acquisition is the concave hull of the various plotted functions and can be compared to the rate × distortion curve for the JPEG2000 standard plotted in black.

We observe that, in terms of compression, CS is much less efficient. However, it has the advantage of taking fewer measurements and, therefore, can be more appropriate for certain applications. Moreover, many of the used parameters can still be improved. We can expect a better performance for other values of $\Phi$ and $\Psi$.

For a final illustration, we present in Figures F.21, F.22, and F.23 the recovered image *lena* (not sparse) for random DCT measurements (Section F.2), Wavelet measurements (Section F.3.1) and quantized Wavelet measurements (Section F.5.2).

Table F.2: PSNR values (dB) for $M = 45k$ and several values of $\epsilon$ and $q_s$ (quantization step).

| $\mathbf{q_s = 0.01}$ | | $\mathbf{q_s = 0.1}$ | | $\mathbf{q_s = 0.2}$ | |
|---|---|---|---|---|---|
| $\boldsymbol{\epsilon = 0.001}$ | **87.85** | $\epsilon = 0.001$ | 76.78 | $\epsilon = 0.001$ | 71.73 |
| $\epsilon = 0.005$ | 87.84 | $\epsilon = 0.005$ | 76.78 | $\epsilon = 0.005$ | 71.69 |
| $\epsilon = 0.010$ | 87.83 | $\boldsymbol{\epsilon = 0.01}$ | **76.78** | $\boldsymbol{\epsilon = 0.010}$ | **71.67** |
| $\epsilon = 0.050$ | 86.60 | $\epsilon = 0.050$ | 76.78 | $\epsilon = 0.050$ | 71.62 |
| $\epsilon = 0.100$ | 86.48 | $\epsilon = 0.100$ | 76.78 | $\epsilon = 0.100$ | 71.79 |
| $\epsilon = 0.500$ | 84.21 | $\epsilon = 0.500$ | 76.07 | $\epsilon = 0.500$ | 71.45 |
| $\epsilon = 1.000$ | 83.14 | $\epsilon = 1.000$ | 76.78 | $\epsilon = 1.000$ | 71.19 |
| $\mathbf{q_s = 0.5}$ | | $\mathbf{q_s = 1}$ | | $\mathbf{q_s = 2}$ | |
| $\epsilon = 0.001$ | 63.88 | $\epsilon = 0.001$ | 57.84 | $\epsilon = 0.010$ | 51.82 |
| $\epsilon = 0.005$ | 63.88 | $\epsilon = 0.005$ | 57.84 | $\epsilon = 0.050$ | 51.82 |
| $\epsilon = 0.010$ | 63.87 | $\epsilon = 0.010$ | 57.84 | $\epsilon = 0.100$ | 51.83 |
| $\boldsymbol{\epsilon = 0.050}$ | **63.87** | $\epsilon = 0.050$ | 57.84 | $\boldsymbol{\epsilon = 0.500}$ | **51.84** |
| $\epsilon = 0.100$ | 63.87 | $\boldsymbol{\epsilon = 0.100}$ | **57.85** | $\epsilon = 1.000$ | 51.83 |
| $\epsilon = 1.000$ | 63.83 | $\epsilon = 1.000$ | 57.84 | $\epsilon = 5.000$ | 51.81 |
| $\epsilon = 5.000$ | 63.42 | $\epsilon = 5.000$ | 57.73 | $\epsilon = 10.00$ | 51.69 |
| $\epsilon = 10.000$ | 62.81 | $\epsilon = 10.00$ | 57.42 | $\epsilon = 20.00$ | 51.49 |
| $\mathbf{q_s = 3}$ | | $\mathbf{q_s = 4}$ | | $\mathbf{q_s = 5}$ | |
| $\epsilon = 0.001$ | 48.25 | $\epsilon = 0.001$ | 45.75 | $\epsilon = 0.001$ | 43.77 |
| $\epsilon = 0.005$ | 48.25 | $\epsilon = 0.010$ | 45.75 | $\epsilon = 0.010$ | 43.77 |
| $\epsilon = 0.010$ | 48.25 | $\epsilon = 0.050$ | 45.75 | $\epsilon = 0.050$ | 43.77 |
| $\epsilon = 0.050$ | 48.25 | $\epsilon = 0.100$ | 45.75 | $\epsilon = 0.100$ | 43.77 |
| $\epsilon = 0.100$ | 48.25 | $\boldsymbol{\epsilon = 0.500}$ | **45.76** | $\boldsymbol{\epsilon = 0.500}$ | **43.77** |
| $\boldsymbol{\epsilon = 0.500}$ | **48.27** | $\epsilon = 1.000$ | 45.76 | $\epsilon = 1.000$ | 43.77 |
| $\epsilon = 1.000$ | 48.26 | $\epsilon = 5.000$ | 45.75 | $\epsilon = 5.000$ | 43.77 |
| $\epsilon = 5.000$ | 48.24 | $\epsilon = 10.00$ | 45.72 | $\epsilon = 10.00$ | 43.75 |
| $\epsilon = 10.00$ | 48.18 | $\epsilon = 50.00$ | 45.42 | $\epsilon = 50.00$ | 43.55 |
| $\epsilon = 20.00$ | 48.08 | $\epsilon = 100.0$ | 44.89 | $\epsilon = 100.0$ | 43.17 |
| $\epsilon = 50.00$ | 47.67 | $\epsilon = 200.0$ | 43.47 | $\epsilon = 200.0$ | 42.13 |
| $\mathbf{q_s = 10}$ | | $\mathbf{q_s = 50}$ | | $\mathbf{q_s = 100}$ | |
| $\epsilon = 0.100$ | 37.78 | $\epsilon = 1.000$ | 24.77 | $\epsilon = 500.0$ | 19.76 |
| $\epsilon = 0.500$ | 37.79 | $\epsilon = 10.00$ | 24.79 | $\epsilon = 800.0$ | 20.02 |
| $\epsilon = 1.000$ | 37.79 | $\epsilon = 50.00$ | 24.87 | $\epsilon = 1000$ | 20.17 |
| $\epsilon = 5.000$ | 37.80 | $\epsilon = 200.0$ | 25.06 | $\epsilon = 1500$ | 19.83 |
| $\boldsymbol{\epsilon = 10.00}$ | **37.80** | $\epsilon = 500.0$ | 25.34 | $\boldsymbol{\epsilon = 2000}$ | **20.50** |
| $\epsilon = 50.00$ | 37.78 | $\boldsymbol{\epsilon = 800.0}$ | **25.56** | $\epsilon = 2500$ | 20.37 |
| $\epsilon = 100.0$ | 37.72 | $\epsilon = 1000$ | 25.52 | $\epsilon = 3000$ | 20.07 |
| $\epsilon = 250.0$ | 37.19 | $\epsilon = 2000$ | 24.59 | $\epsilon = 5000$ | 17.93 |
| $\epsilon = 500.0$ | 35.90 | $\epsilon = 5000$ | 19.95 | | |

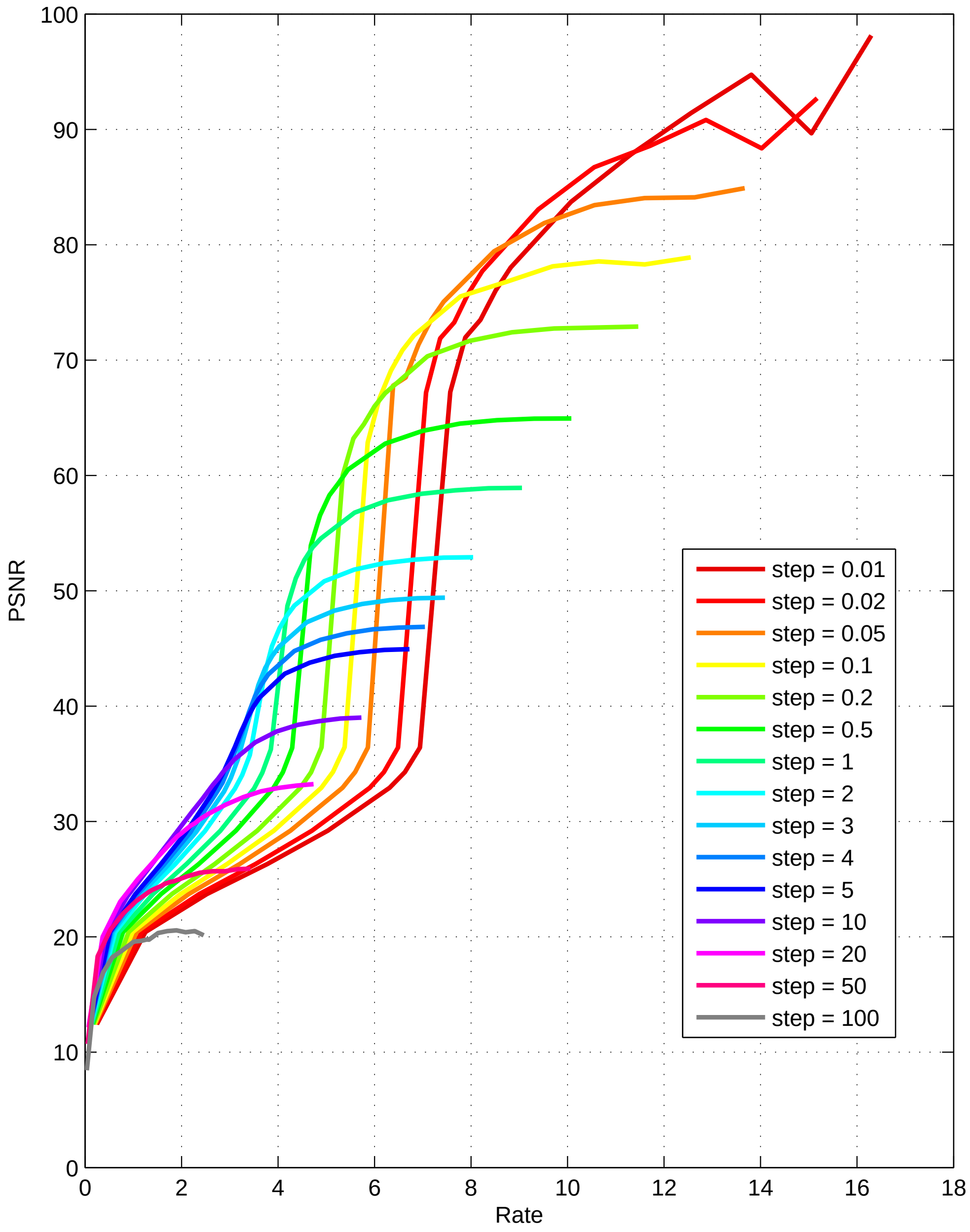


Figure F.19: Results for applications of CS scheme to quantized versions of the $10k$-sparse representation of image *lena*. The strategy involves taking Noiselet measurements and assuming that the image is sparse in the Wavelet domain. Here we plotted PSNR × Rate, where the rate is calculated based on the signal's entropy.

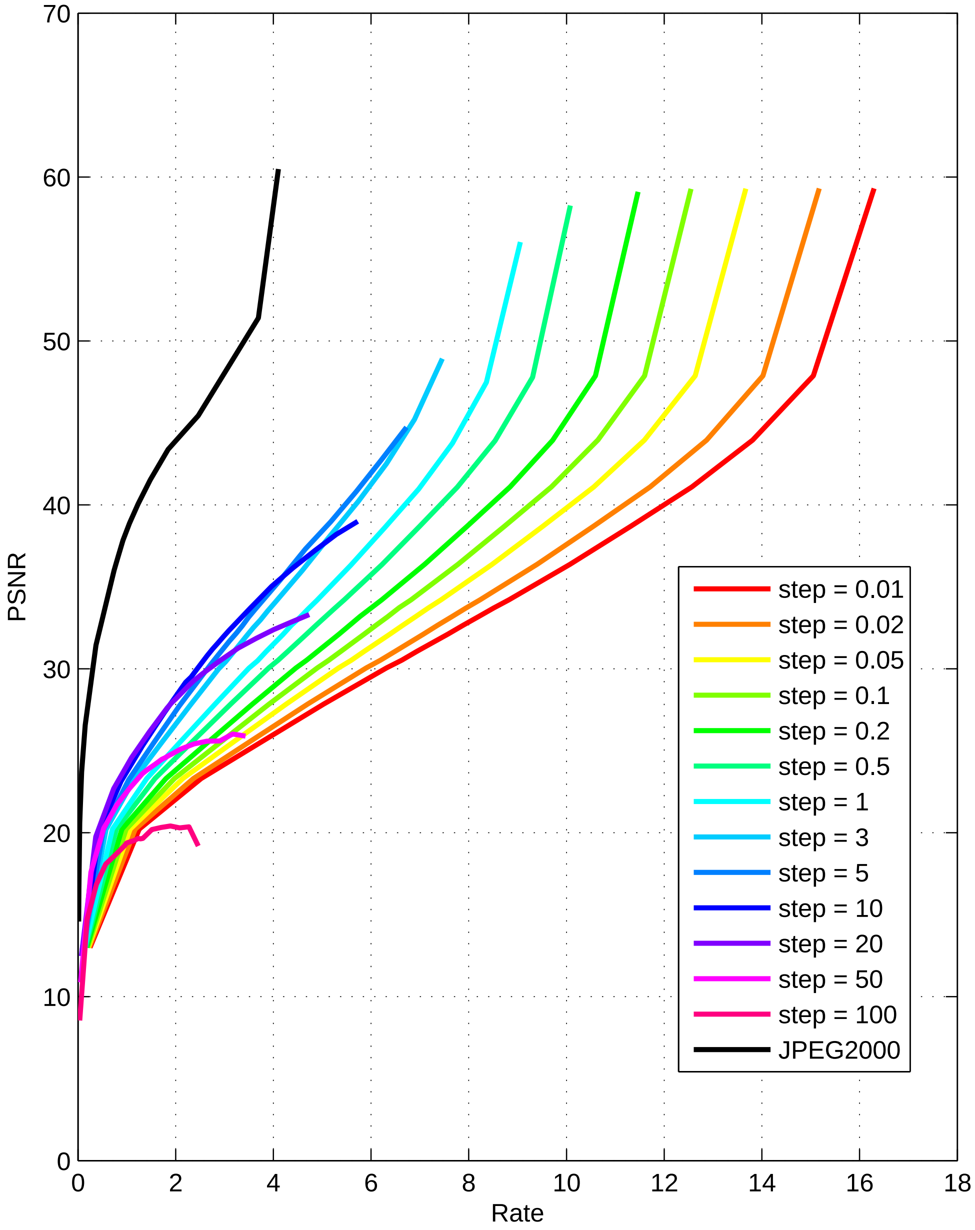
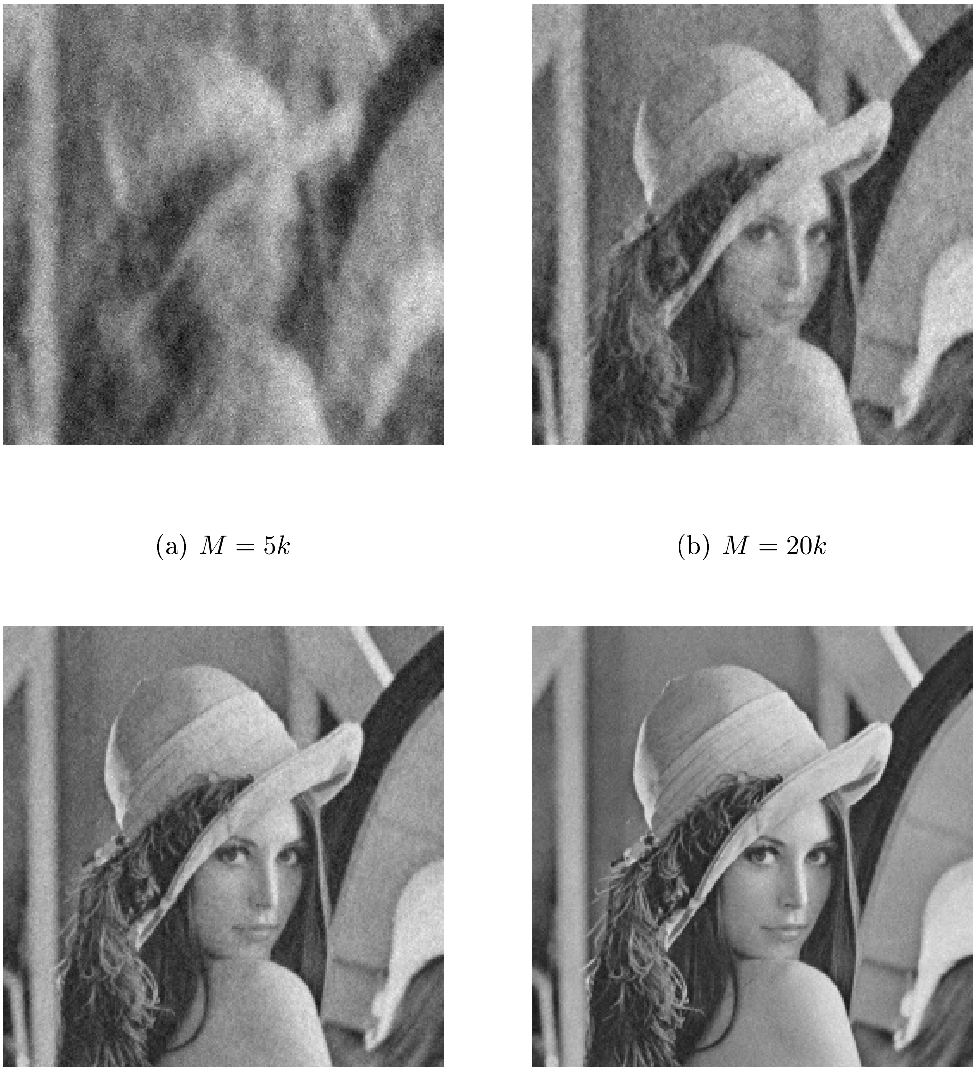


Figure F.20: Results for applications of CS scheme to quantized versions of the original (only approximately sparse) image *lena*. The strategy involves taking Noiselet measurements and assuming that the image is sparse in the Wavelet domain. Here we plotted PSNR × Rate, where the rate is calculated based on the signal's entropy.



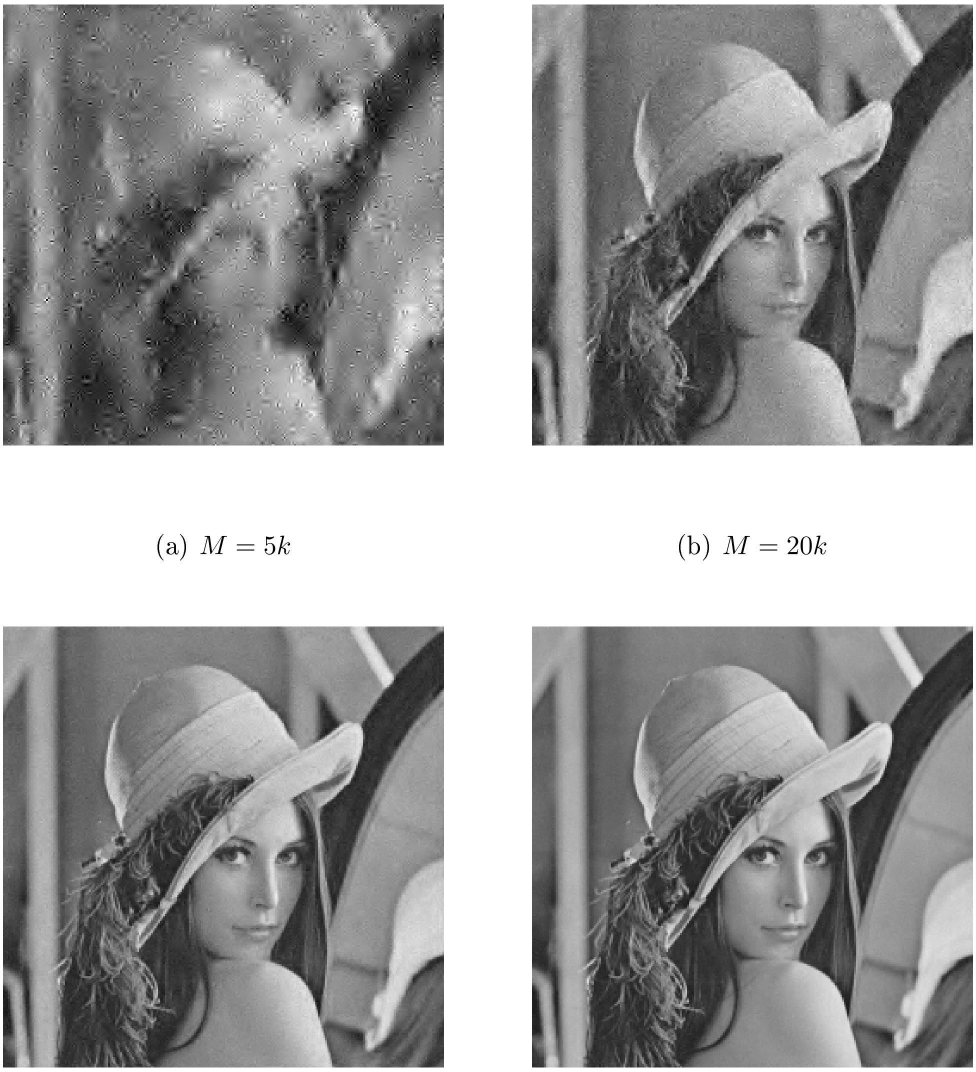
(a) $M = 5k$

(b) $M = 20k$

(c) $M = 35k$

(d) $M = 50k$

Figure F.21: Recovered image *lena* for acquisition based on Noiselet measurements and considering that the signal is approximately sparse in the DCT domain.



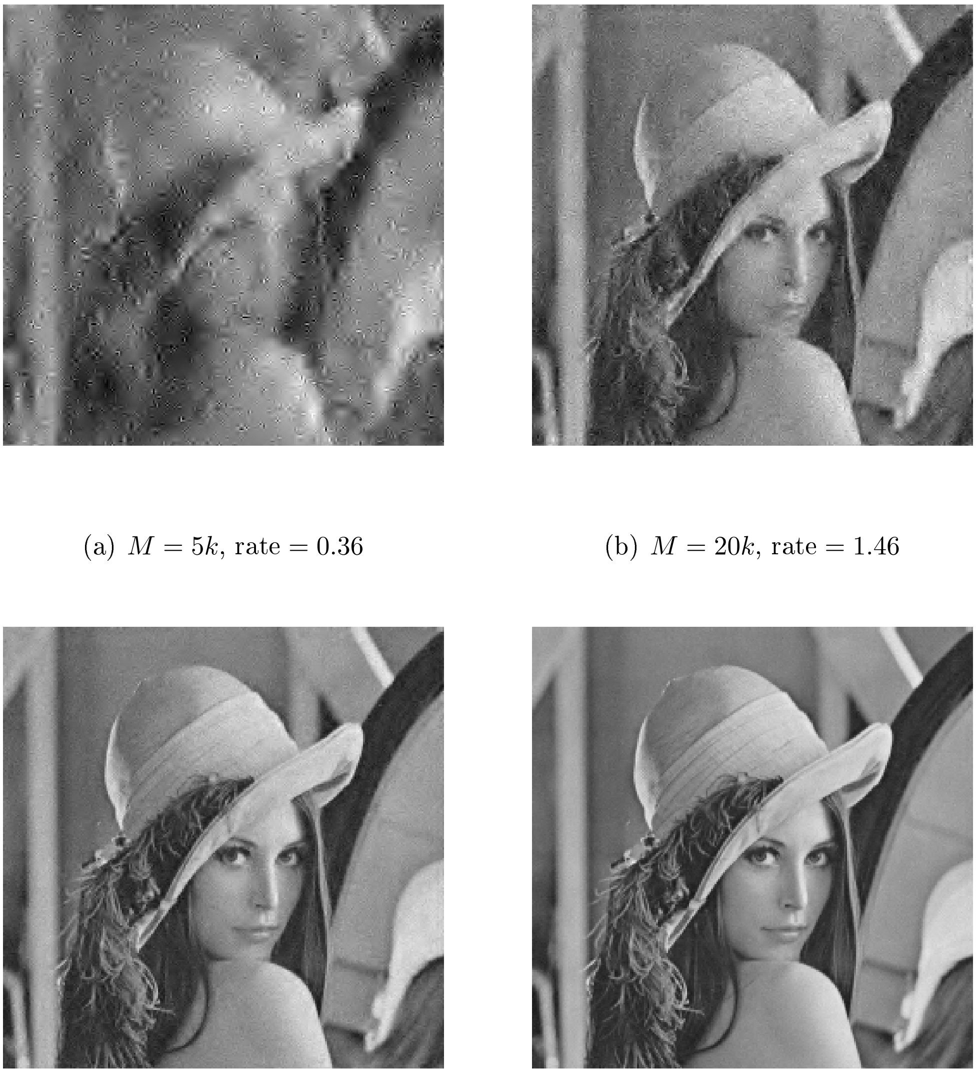
(a) $M = 5k$

(b) $M = 20k$

(c) $M = 35k$

(d) $M = 50k$

Figure F.22: Recovered image *lena* for acquisition based on Noiselet measurements and considering that the signal is approximately sparse in the Wavelet domain.



(a) $M = 5k$, rate $= 0.36$

(b) $M = 20k$, rate $= 1.46$

(c) $M = 35k$, rate $= 3.09$

(d) $M = 50k$, rate $= 5.17$

Figure F.23: Recovered image *lena* for acquisition based on Noiselet measurements corrupted by quantization noise and considering that the signal is approximately sparse in the Wavelet domain.

# Appendix G

# Conclusions

During the course of this work we introduced Compressive Sensing as a new paradigm for image acquisition. Our study involved an overview of standard procedures for sensing and compression in order to familiarize the reader and motivate applications.

We discussed the most important definitions and theorems that formalize CS theory and stated some relevant arguments that justify its efficiency. Finally, we produced examples related to image compression and were able to evaluate performance in a series of different scenarios.

We observe that several adjustments must be made in order to allow CS to be applicable in modern acquisition conditions, once the compression rate is significantly smaller than standard compression schemes and the recovery strategy is computationally more expensive.

However, it is noteworthy that this theory has a lot of potential, once it rises against the common knowledge of the filed and, therefore, allows us to look at data acquisition from a different point of view. This suggests that applications in different circumstances can and should be experimented.

## G.1 Future Work

In many recent publications [12, 13, 14], researchers have used the total-variation norm instead of the $l_1$-norm. In view of the relation between the TV-norm and the gradient of the $l_1$ norm, there is a conjecture commonly found in the literature that the theorems are still valid under this condition [15]. Applied to images, the TV-norm minimization suggest a certain smoothness that is usually found in natural and manmade pictures and is, therefore, very efficient. An extention of this study should, hence, consider this approach.

It would also be interesting to experiment alternatives to Noiselet measurements. In the future, we intend to experiment acquisition with random Gaussian matrices and Whash-Hadamard functions.

The choice of orthogonal Wavelets came from the structure of the recovery algorithm, that needs as input the matrix $\Theta$ and its transpose. Though biorthogonal Wavelets are more adequate to enhance sparsity, they do not lead to auto-adjoint transform matrices, making the implementation rather difficult. In the future, more careful analysis of the $l_1$-Magic toolbox will allow the use on non auto-adjoint matrices.

It will also be interesting to verify how CS behaves when block partitioning is adopted. We notice that the number of samples grows with a $\log N$ factor. Though researchers claim that we can expect to recovery most signals 50% of the time if $M \geq 4S$ [12], it would be interesting to consider very large images and compare acquisitions schemes with and without block partitioning.

english

# Referências Bibliográficas

[1] Eduardo da Silva; Lisandro Lovisolo. *TV Digital - Notas de aula.* DEL/Poli/UFRJ, 2007.

[2] Khalid Sayood. *Introduction to data compression.* Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2000.

[3] Stèphane Mallat; Zhifeng Zhang. Matching pursuits with time-frequency dictionaries. Technical report, New York, NY, USA, 1993.

[4] Richard Baraniuk. Compressive sensing. *IEEE Signal Processing Magazine*, 24(4), July 2007.

[5] Emmanuel Candès; Michael Wakin. An introduction to compressive sampling. *IEEE Signal Processing Magazine*, 25(2), March 2008.

[6] Emmanuel Candès; Terence Tao. Near optimal signal recovery from random projections: Universal encoding strategies? *IEEE Trans. on Information Theory*, 52(12), December 2006.

[7] Emmanuel Candès; Justin Romberg. Sparsity and incoherence in compressive sampling. *Inverse Problems*, 23(3):969–985, 2007.

[8] Emmanuel Candès; Terence Tao. Decoding by linear programming. *IEEE Trans. on Information Theory*, 51(12), December 2005.

[9] Emmanuel Candès. The restricted isometry property and its implications for compressed sensing. *Compte Rendus de l'Academie des Sciences, Series*, 346:589–590, 2008.

[10] Justin Romberg. Imaging via compressive sampling. *IEEE Signal Processing Magazine*, 25(2), March 2008.

[11] Emmanuel Candès; Justin Romberg. *L1–Magic: Recovery of Sparse Signals via Convex Programming.* 2006.

[12] Emmanuel Candès; Justin Romberg; Terence Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. on Information Theory*, 52(2), February 2006.

[13] Dharmpal Takhar; Jason N. Laska; Michael B. Wakin; Marco F. Duarte; Dror Baron Shriram Sarvotham; Kevin F. Kelly; Richard G. Baraniuk. A new compressive imaging camera architecture using optical-domain compression. In *Computational Imaging IV at SPIE Electronic Imaging*, San Jose, California, January 2006.

[14] D. L. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52(4).

[15] The Institute for Mathematics and its Applications (IMA). *Lectures on compressive sampling and frontiers in signal processing*, University of Minnesota, June 2007.

[16] Anil K. Jain. *Fundamentals of digital image processing.* Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1989.

[17] Jonas Gomes; Luiz Velho. *From Fourier Analysis to Wavelets.* SIGGRAPH'99 Course Notes 5, SIGGRAPH-ACM publication, Los Angeles, California, USA, August 1999.

[18] Eduardo A. B. da Silva. *Wavelet Transforms for Image Coding.* PhD thesis, Essex University - Colshester, UK, June 1995.

[19] Eduardo da Silva; Gelson Mendonca. *Digital Image Processing*, chapter VII.4, pages 891–910. The Electrical Engineering Handbook. Wai-Kai Chen, Elsevier - Academic Press, 2005.

[20] Simon Haykin. *Sistemas de comunicações analógicas e digitais.* Bookman, São Paulo, SP, Brasil, 2004.

[21] Majid Rabbani; Rajan Joshi. An overview of the jpeg 2000 still image compression standard. *Signal Processing: Image Communication*, 17:3–48, 2002.

[22] Jonas Gomes; Luiz Velho. *Image Processing for Computer Graphics.* Springer Verlag, 1997.

[23] Stèphane Mallat. *A Wavelet Tour of Signal Processing.* Academic Press, San Diego, CA, USA, second edition edition, 1999.

[24] Paulo A. R. Diniz; Eduardo A. B. da Silva; Sergio L. Netto. *Processamento Digital de Sinais - Projeto e Análise de Sistemas.* Bookman, Porto Alegre, 2004.

[25] Rogério Caetano. *Video Coding using Generalized Bit-planes.* PhD thesis, COPPE/UFRJ, March 2004.

[26] Scott Shaobing Chen; David L. Donoho; Michael A. Saunders. Atomic decomposition by basis pursuit. *SIAM J. Sci. Comput.*, 20(1):33–61, 1998.

[27] David Donoho; Philip Stark. Uncertainty principles and signal recovery. *SIAM Journal on Applied Mathematics*, 49(3):906–931, 1989.

[28] Emmanuel Candès; Justin Romberg; Terence Tao. Stable signal recovery from incomplete and inaccurate measurements. *Communications on Pure and Applied Mathematics*, 59(8), August 2006.

[29] Emmanuel Candès. Compressive sampling. *Int. Congress of Mathematics*, 3:1433–1452, 2006.

[30] R. Coifman; F. Geshwind; Y. Meyer. Noiselets. *Appl. Comp. Harmon.Anal.*, 10(1):27–44, 2001.