
Projeto de Iniciação Científica

**Aspectos Matemáticos e Computacionais
da Compressão de Imagens**

Estudante: Leonardo Arazo de Oliveira Araújo

*Bacharelado em Ciência e Tecnologia - ICT - UNIFESP - Campus São
José dos Campos*

Orientador: Prof. Dr. Thadeu Alves Senne

ICT - UNIFESP - Campus São José dos Campos

Sumário

1	Resumo	3
2	Introdução e Justificativa	3
3	Objetivos	5
4	Material e Método	5
4.1	Decomposição em Valores Singulares	6
4.2	<i>Compressive Sensing</i>	7
5	Forma de Análise dos Resultados	10
6	Plano de Trabalho e Cronograma de Atividades	10
	Referências	11

1 Resumo

Nos dias atuais, lidamos com uma quantidade enorme de dados, que necessitam ser analisados e tratados. Embora a capacidade de armazenamento e de processamento dos computadores tenha crescido expressivamente ao longo das últimas décadas, é imperativo lançar mão de técnicas que sejam capazes de reduzir de forma significativa a quantidade de informações necessárias para processar e interpretar os dados, sem comprometer sua qualidade. Isso pode ser feito por meio de uma reconstrução dos dados, utilizando uma quantidade relativamente pequena de amostras. Levando isso em consideração, este projeto visa o estudo da fundamentação matemática e a implementação computacional de duas das principais técnicas de compressão de dados (Decomposição em Valores Singulares e *Compressive Sensing*), que certamente poderá contribuir para futuros avanços no desenvolvimento desta área.

2 Introdução e Justificativa

Nas últimas décadas, temos observado um expressivo aumento da necessidade de processar, armazenar e transmitir volumes gigantescos de dados, em uma velocidade cada vez maior. Para que estas informações sejam acessadas de forma eficiente e rápida, há uma intensa atividade de pesquisa científica referente ao desenvolvimento de técnicas de compressão de dados, com o intuito de ocupar cada vez menos memória dos computadores e facilitar a transmissão, com uma deterioração mínima desses dados.

Um dos objetivos principais das técnicas de compressão de dados é reduzir consideravelmente a quantidade de informações necessárias para exibir uma imagem digital, mas sem perder nenhuma informação essencial. Para isso, uma imagem digital em preto e branco pode ser armazenada como uma matriz, em que cada entrada corresponde a um *pixel* (que corresponde à menor unidade que compõe uma imagem digital), de modo que o número 0 representa a cor preta (ausência de intensidade), e o número 255 indica a cor branca (intensidade máxima), totalizando 256 tons de cinza diferentes. Logo, as entradas da matriz em questão são números inteiros entre 0 e 255.

Neste trabalho, serão estudados e implementados dois métodos bastante empregados nas técnicas de compressão e de reconstrução de imagens digitais: a Decomposição em Valores Singulares – SVD, do Inglês *Singular Value Decomposition* (Golub & Van Loan [4], Meyer [11], Watkins [14]), e o *Compressive Sensing* (Donoho [3]), que serão descritos matematicamente na Seção 3. É importante mencionar que, na literatura científica, não é de costume traduzir a expressão *Compressive Sensing* para o Português.

A ideia da utilização da SVD para compressão de imagens digitais é a seguinte: dada uma matriz \mathbf{A} de ordem $m \times n$ que representa esta imagem, encontramos seus valores singulares $\sigma_1, \sigma_2, \dots, \sigma_r$ (com $r \leq \min\{m, n\}$) ordenados em uma sequência decrescente. Em seguida, construímos uma matriz \mathbf{A}_k que aproxima a matriz original \mathbf{A} ,

utilizando os seus k maiores valores singulares (com $k < r$), de tal maneira que a matriz \mathbf{A}_k contenha uma quantidade consideravelmente reduzida de entradas não nulas em relação à matriz \mathbf{A} .

No trabalho de Pradhan *et al.* [12], foi desenvolvido um algoritmo baseado no uso da SVD para o reconhecimento de expressões faciais e para a detecção de sinais de sonolência ou fadiga em motoristas, por meio de rastreamento ocular. Em Ahmadi *et al.* [1] e Zermi *et al.* [15], foram propostas técnicas de geração de marcas d'água digitais, baseada na combinação da Transformada *Wavelet* Discreta com a SVD, que podem ser aplicadas à proteção de direitos autorais e de imagens médicas, por exemplo. Tripathi & Garg [13] abordam o uso da SVD para reconstruir imagens hiperespectrais terrestres ou lunares (ou seja, imagens da Terra ou da Lua que são capturadas com um grande número de comprimentos de onda).

A técnica de *Compressive Sensing*, também conhecida como *Compressed Sensing*, consiste na recuperação de uma imagem (que é considerada como um *signal*), feita por meio da coleta de um número pequeno de amostras (isto é, de partes) da imagem original, de tal forma que a imagem recuperada seja um *signal esparsa*, ou seja, ela passa a ser representada com poucos dados, que são efetivamente relevantes. A *esparsidade* de um *signal* pode ser quantificada por meio de algumas métricas.

Dentre as aplicações de diversas variantes do *Compressive Sensing*, destacamos aquelas empregadas na Medicina, mais especificamente na obtenção de diagnósticos via ressonância magnética. Neste contexto, Mardani *et al.* [9] propuseram um método que usa redes adversárias generativas para reduzir ruídos gerados pela compressão das imagens. Um modelo baseado em redes neurais convolucionais foi desenvolvido por Hashimoto *et al.* [5], e uma comparação do desempenho entre algumas técnicas de *Compressive Sensing* foi realizada por Herguedas-Alonso *et al.* [6]. Monika *et al.* [10] desenvolveram um método de *Compressive Sensing* adaptativo por blocos com avaliação de coeficientes de limiar mistos, para reconstruir imagens médicas com poucas amostras e menos distorções.

Tendo em vista as aplicações mencionadas acima, notamos que o estudo matemático e computacional de técnicas de compressão de imagens desempenha um papel fundamental para o desenvolvimento científico, tecnológico, econômico e social. Além disso, certamente poderá abrir caminhos para aprimorar os métodos já existentes e até mesmo inspirar a criação de outras técnicas na área. Tudo isso motiva esta proposta de projeto de Iniciação Científica.

3 Objetivos

Os objetivos deste projeto são:

- Estudar os fundamentos matemáticos necessários para uma bom entendimento do problema de compressão de imagens.
- Compreender a modelagem matemática dos problemas de compressão de imagens por meio das técnicas da Decomposição em Valores Singulares e de *Compressive Sensing*.
- Identificar, se possível, as vantagens e desvantagens de cada técnica estudada, por meio de testes dos programas que serão implementados pelo candidato, utilizando a linguagem de programação *Julia* (para mais informações, acesse <https://julialang.org/downloads/>).

Mais especificamente, para cumprir estes objetivos, espera-se que o candidato cumpra as etapas descritas no Plano de Trabalho e no Cronograma de Atividades, que serão apresentados na Seção 6.

4 Material e Método

Nesta seção, apresentaremos uma breve descrição sobre os métodos matemáticos que serão estudados neste projeto e que serão aplicados ao problema de compressão de imagens: a Decomposição em Valores Singulares, e o *Compressive Sensing*.

Antes de prosseguirmos, é importante mencionarmos duas observações sobre a notação matemática empregada ao longo do texto.

1. Um vetor $\mathbf{z} \in \mathbb{R}^n$, cujas componentes são os números reais z_1, z_2, \dots, z_n , será representado por uma matriz-coluna $n \times 1$, isto é,

$$\mathbf{z} = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_n \end{bmatrix},$$

ou, de forma equivalente,

$$\mathbf{z} = [z_1 \ z_2 \ \cdots \ z_n]^T,$$

pois a transposta de uma matriz-linha $1 \times n$ é uma matriz-coluna $n \times 1$. Logo, temos

$$\mathbf{z}^T = [z_1 \ z_2 \ \cdots \ z_n].$$

2. Considere dois vetores quaisquer $\mathbf{a} = [a_1 \ a_2 \ \dots \ a_n]^T$ e $\mathbf{b} = [b_1 \ b_2 \ \dots \ b_n]^T$ de \mathbb{R}^n . O *produto interno usual* de \mathbb{R}^n é definido por

$$\langle \mathbf{a}, \mathbf{b} \rangle = a_1 b_1 + a_2 b_2 + \cdots + a_n b_n = \sum_{i=1}^n a_i b_i = \mathbf{a}^T \mathbf{b}.$$

4.1 Decomposição em Valores Singulares

Seja $\mathbf{A} \in \mathbb{R}^{m \times n}$ uma matriz qualquer. A **Decomposição em Valores Singulares (SVD)** (veja [4, 11, 14]) desta matriz \mathbf{A} é dada por

$$\mathbf{A} = \mathbf{U} \mathbf{S} \mathbf{V}^T, \quad (1)$$

em que $\mathbf{U} \in \mathbb{R}^{m \times m}$ e $\mathbf{V} \in \mathbb{R}^{n \times n}$ são matrizes ortogonais e $\mathbf{S} \in \mathbb{R}^{m \times n}$ é uma matriz da forma

$$\mathbf{S} = \begin{bmatrix} \hat{\mathbf{S}}_{r \times r} & \mathbf{O}_{r \times (n-r)} \\ \mathbf{O}_{(m-r) \times r} & \mathbf{O}_{(m-r) \times (n-r)} \end{bmatrix},$$

sendo $\hat{\mathbf{S}} = \text{diag}(\sigma_1, \dots, \sigma_r)$ uma matriz diagonal de ordem $r \times r$ com $\sigma_i > 0$ para todo $i = 1, \dots, r$, e $\mathbf{O}_{p \times q}$ indica a matriz nula de ordem $p \times q$. As entradas $\sigma_1, \dots, \sigma_r$ da diagonal da matriz $\hat{\mathbf{S}}$ são os chamados **valores singulares** de \mathbf{A} , e são ordenados em uma sequência decrescente, ou seja, $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r > 0$.

Pode-se provar (veja [14]) que a SVD da matriz \mathbf{A} dada em (1) pode ser expressa como

$$\mathbf{A} = \sigma_1 \mathbf{u}_1 \mathbf{v}_1^T + \sigma_2 \mathbf{u}_2 \mathbf{v}_2^T + \dots + \sigma_r \mathbf{u}_r \mathbf{v}_r^T = \sum_{i=1}^r \sigma_i \mathbf{u}_i \mathbf{v}_i^T, \quad (2)$$

em que \mathbf{u}_i e \mathbf{v}_i denotam, respectivamente, as i -ésimas colunas das matrizes \mathbf{U} e \mathbf{V} .

Dada uma matriz \mathbf{A} e sua SVD na forma (2), para cada $k = 1, \dots, r$, definimos a matriz

$$\mathbf{A}_k = \sum_{i=1}^k \sigma_i \mathbf{u}_i \mathbf{v}_i^T. \quad (3)$$

Observando (2) e (3), note que $\mathbf{A}_r = \mathbf{A}$. Além disso, pode-se provar (veja [14]) que, se $\text{posto}(\mathbf{A}) = r$, então \mathbf{A} possui exatamente r valores singulares positivos, e a matriz \mathbf{A}_k dada em (3) é a matriz de posto k que está mais próxima de \mathbf{A} . Tendo este resultado em mãos, a ideia fundamental do esquema de compressão de imagens baseada na SVD é selecionar um número adequado k de valores singulares da matriz original \mathbf{A} , para obter a matriz \mathbf{A}_k dada em (3), que aproxima a matriz \mathbf{A} .

Uma vez que cada coluna de \mathbf{U} possui m entradas e cada coluna de \mathbf{V} contém n entradas, para armazenar a matriz \mathbf{A}_k dada em (3), precisamos ter disponíveis as k primeiras colunas de \mathbf{U} , as k primeiras colunas de \mathbf{V} , e os k primeiros valores singulares. Logo, para obter \mathbf{A}_k , a quantidade de números que teremos de armazenar será

$$k + k \cdot m + k \cdot n = k(1 + m + n).$$

Se todas as mn entradas de uma matriz $\mathbf{A} \in \mathbb{R}^{m \times n}$ que corresponde a uma imagem fossem armazenadas, estaríamos representando a imagem original. Suponha que esta matriz \mathbf{A} tenha r valores singulares positivos, com $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r > 0$. Para algum

k inteiro com $1 < k < r$, suponha que $\sigma_{k+1}, \dots, \sigma_r$ sejam suficientemente pequenos (ou seja, muito próximos de zero). Neste caso, ao desprezar as parcelas correspondentes em (2), obtemos uma matriz \mathbf{A}_k na forma (3), cuja imagem resultante está associada a uma versão comprimida da imagem original.

Para ilustrarmos brevemente o processo acima, considere uma imagem de tamanho 1920×1080 *pixels*, que corresponde a uma imagem exibida em uma tela com resolução *full HD*. Neste caso, a matriz \mathbf{A} correspondente a essa imagem é de ordem 1920×1080 , de modo que $m = 1920$ e $n = 1080$. Se trabalhássemos com todos os dados referentes a esta imagem, teríamos de armazenar $mn = 1920 \times 1080 = 2073600$ números. Se quisermos, por exemplo, realizar a compressão desta imagem utilizando somente os $k = 50$ maiores valores singulares de \mathbf{A} , armazenaremos apenas

$$k(1 + m + n) = 50(1 + 1920 + 1080) = 150050$$

números. Logo, somente

$$100 \left(\frac{k(1 + m + n)}{mn} \right) = 100 \left(\frac{150050}{2073600} \right) \approx 7,24\%$$

dos dados originais serão utilizados para reconstruir a imagem, o que resulta numa compressão de $100\% - 7,24\% = 92,76\%$ desses dados.

4.2 Compressive Sensing

Seja $\mathbf{x} = [x_1 \ x_2 \ \dots \ x_n]^T \in \mathbb{R}^n$ um sinal real, finito e discreto no tempo. Dada uma base ortonormal $\mathcal{B} = \{\boldsymbol{\psi}_1, \boldsymbol{\psi}_2, \dots, \boldsymbol{\psi}_n\}$ de \mathbb{R}^n , este sinal pode ser expresso como combinação linear dos vetores de \mathcal{B} , ou seja, existem escalares reais s_1, s_2, \dots, s_n tais que

$$\mathbf{x} = s_1 \boldsymbol{\psi}_1 + s_2 \boldsymbol{\psi}_2 + \dots + s_n \boldsymbol{\psi}_n = \sum_{i=1}^n s_i \boldsymbol{\psi}_i. \quad (4)$$

Definindo uma matriz $\boldsymbol{\Psi} \in \mathbb{R}^{n \times n}$ cujas colunas são os vetores de \mathcal{B} , podemos reescrever (4) como

$$\mathbf{x} = \boldsymbol{\Psi}^T \mathbf{s}, \quad (5)$$

em que $\mathbf{s} = [s_1 \ s_2 \ \dots \ s_n]^T$.

Uma vez que estamos supondo que \mathcal{B} é uma base ortonormal de \mathbb{R}^n , temos que a matriz $\boldsymbol{\Psi}$ é ortogonal (ou seja, $\boldsymbol{\Psi}^T \boldsymbol{\Psi} = \boldsymbol{\Psi} \boldsymbol{\Psi}^T = \mathbf{I}$, em que \mathbf{I} é a matriz identidade de ordem n). Logo, por (5), obtemos

$$\mathbf{s} = \boldsymbol{\Psi} \mathbf{x}. \quad (6)$$

Dizemos que um sinal $\mathbf{x} \in \mathbb{R}^n$ é *k-esparso* se \mathbf{x} for expresso como combinação linear de apenas no máximo k dos n vetores da base ortonormal \mathcal{B} de \mathbb{R}^n , ou seja, quando no máximo k das n componentes do vetor \mathbf{s} são não nulas. Idealmente, na prática, é desejável que k seja muito menor do que n .

O processo de aquisição de um sinal, que está ilustrado na Figura 1, consiste em reunir amostras a partir do produto interno do sinal \mathbf{x} com m vetores de teste denotados por $\phi_1, \dots, \phi_m \in \mathbb{R}^n$, em que $m \ll n$, de modo a obter m medições dadas por

$$\mathbf{y}_i = \langle \phi_i, \mathbf{x} \rangle = \phi_i^T \mathbf{x}, \quad i = 1, \dots, m. \quad (7)$$

Definindo a matriz $\Phi \in \mathbb{R}^{m \times n}$ cuja i -ésima linha é dada por ϕ_i^T , podemos escrever (7) como

$$\mathbf{y} = \Phi \mathbf{x}. \quad (8)$$

Substituindo (5) em (8), encontramos

$$\mathbf{y} = \Phi \mathbf{x} = \Phi \Psi^T \mathbf{s} = \Theta \mathbf{s}, \quad (9)$$

em que $\Theta = \Phi \Psi^T \in \mathbb{R}^{m \times n}$. Observe que (9) representa um sistema linear com m equações e n incógnitas (que são as componentes do vetor \mathbf{s}). Além disso, uma vez que $m < n$, se $\text{posto}(\Theta) = m$, então o sistema linear (9) terá infinitas soluções.

Note, pela Figura 1, que uma imagem pode ser representada por uma matriz, em que cada uma de suas entradas está associada a um *pixel*. As colunas desta matriz são empilhadas para formar o sinal \mathbf{x} , percorrendo-a da esquerda para a direita. Tendo o sinal \mathbf{x} em mãos, obtemos as amostras, que são dadas pelas componentes do vetor \mathbf{y} dado em (9).

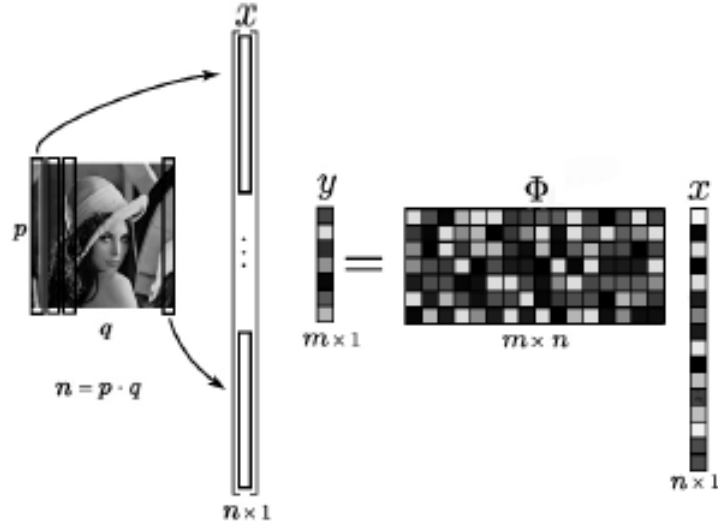


Figura 1: Ilustração do processo de aquisição de um sinal, com poucas amostras (Figura extraída de [7]).

De um modo geral, o problema de recuperar um sinal k -esparso a partir de m amostras, ou seja, obter um sinal \mathbf{s}^* com a maior quantidade possível de elementos nulos e que satisfaça (9), pode ser formulado como um problema de otimização, dado por (veja Baraniuk [2])

$$\begin{aligned} & \text{minimizar} && f(\mathbf{s}) \\ & \text{sujeito a} && \Theta \mathbf{s} = \mathbf{y}, \end{aligned} \quad (10)$$

em que $f : \mathbb{R}^n \rightarrow \mathbb{R}$ é a *função objetivo* que, neste caso, representa uma métrica para a esparsidade do sinal.

Em Baraniuk [2], são apresentadas algumas escolhas para a função objetivo f do problema (10), que serão descritas a seguir. Uma delas é a função que representa o número de componentes não nulas do vetor \mathbf{s} , denotada por $f(\mathbf{s}) = \|\mathbf{s}\|_0$. Mas, infelizmente, ao adotar esta métrica, a resolução do problema (10) exige um processo de busca exaustiva dentre todos os vetores de \mathbb{R}^n com no máximo k componentes diferentes de zero, o que é impraticável. Além disso, esta função não é diferenciável, o que impede o uso de métodos clássicos de otimização que requerem o cálculo de derivadas. Na tentativa de contornar este problema, são utilizadas com frequência outras duas métricas de esparsidade, dadas por

$$f(\mathbf{s}) = \|\mathbf{s}\|_1 = \sum_{i=1}^n |s_i|, \quad (11)$$

que é a *norma-1* de \mathbf{s} , e

$$f(\mathbf{s}) = \frac{1}{2} \|\mathbf{s}\|_2^2 = \frac{1}{2} \sum_{i=1}^n s_i^2, \quad (12)$$

em que $\|\mathbf{s}\|_2$ denota a *norma Euclidiana* de \mathbf{s} .

Se utilizarmos a métrica (11), é possível reescrever o problema (10) como um problema de Otimização Linear (veja Luenberger [8]). Neste trabalho, tais problemas serão resolvidos utilizando o pacote GLPK, disponível na linguagem de programação *Julia* (veja <https://juliapackages.com/p/glpk>).

Quando a função (12) é adotada como métrica de esparsidade, temos a vantagem de que o problema (10) possui uma solução explícita, dada por

$$\mathbf{s}^* = \Theta^T (\Theta \Theta^T)^{-1} \mathbf{y}. \quad (13)$$

O cálculo da inversa de uma matriz (quando ela existe) é muito caro do ponto de vista computacional, ou seja, demanda uma grande quantidade de operações que podem ser evitadas. Desta forma, na prática, para evitar a obtenção da matriz inversa $(\Theta \Theta^T)^{-1}$, reescrevemos (13) como

$$\mathbf{s}^* = \Theta^T \mathbf{b},$$

em que

$$\mathbf{b} = (\Theta \Theta^T)^{-1} \mathbf{y},$$

de modo que \mathbf{b} é encontrado por meio da solução do sistema linear

$$(\Theta \Theta^T) \mathbf{b} = \mathbf{y}. \quad (14)$$

Observamos que, como $m < n$, se $\text{posto}(\Theta) = m$, a matriz $\Theta \Theta^T$ é positiva definida e, portanto, não singular. Neste caso, utilizaremos o *Método dos Gradientes Conjugados* (veja Luenberger [8]) para resolver o sistema linear (14).

O problema de *Compressive Sensing* será resolvido utilizando a formulação dada por (10) com a adoção das métricas (11) e (12), e será investigado o potencial destas duas métricas para gerar sinais esparsos.

5 Forma de Análise dos Resultados

Para validar os resultados obtidos pelas técnicas da Decomposição em Valores Singulares e de *Compressive Sensing*, elas serão implementadas utilizando a linguagem de programação *Julia*. As compressões de imagens frequentemente encontradas na literatura serão testadas por meio dos programas desenvolvidos pelo candidato. Além disso, por meio dos resultados obtidos no testes computacionais, o candidato analisará as possíveis vantagens e desvantagens de cada técnica estudada, e fará uma comparação do desempenho entre elas.

6 Plano de Trabalho e Cronograma de Atividades

As atividades deste projeto serão divididas nas etapas descritas abaixo.

1. Compreensão da linguagem de programação *Julia* e dos fundamentos de compressão de imagens.
2. Estudo da técnica de Decomposição em Valores Singulares.
3. Implementação (em *Julia*) da técnica de Decomposição em Valores Singulares para resolver o problema de compressão de imagens e a respectiva realização de testes computacionais.
4. Estudo da técnica de *Compressive Sensing* e do Método dos Gradientes Conjugados.
5. Implementação (em *Julia*) do Método dos Gradientes Conjugados para o problema de compressão de imagens pela técnica de *Compressive Sensing* e a respectiva realização de testes computacionais.
6. Realização de testes computacionais de compressão de imagens pela técnica de *Compressive Sensing*, utilizando o pacote de otimização GLPK (em *Julia*).
7. Análise e comparação dos resultados obtidos por cada uma das técnicas estudadas.
8. Redação do Relatório Final de Atividades.
9. Participação no Congresso Acadêmico da UNIFESP.
10. Entrega do Relatório Final de Atividades.

O cronograma com a realização de cada etapa está mostrado abaixo.

Etapas	Realização
1, 2 e 3	setembro a dezembro de 2024
4, 5 e 6	janeiro a maio de 2025
7 e 8	junho e julho de 2025
9	(a definir)
10	agosto de 2025

Referências

- [1] Ahmadi, S. B. B.; Zhang, G.; Wei, S.; Boukela, L.; *An intelligent and blind image watermarking scheme based on hybrid SVD transforms using human visual system characteristics*. The Visual Computer, 37, 2021, pp. 385-409, <https://doi.org/10.1007/s00371-020-01808-6>
- [2] Baraniuk, R. G.; *Compressive Sensing [Lecture Notes]*. IEEE Signal Processing Magazine, 24(4), 2007, pp. 118-121, <https://doi.org/10.1109/MSP.2007.4286571>
- [3] Donoho, D. L.; *Compressed Sensing*. IEEE Transactions on Information Theory, 52(4), 2006, pp. 1289-1306, <https://doi.org/10.1109/TIT.2006.871582>
- [4] Golub, G. H.; Van Loan, C. F.; *Matrix Computations*, Fourth Edition, Philadelphia: The Johns Hopkins University Press, 2013, ISBN 978-1-4214-0794-4
- [5] Hashimoto, F.; Ote, K.; Oida, T.; Teramoto, A.; Ouchi, Y.; *Compressed-Sensing Magnetic Resonance Image Reconstruction Using an Iterative Convolutional Neural Network Approach*. Applied Sciences, 10(6), 2020, <https://doi.org/10.3390/app10061902>
- [6] Herguedas-Alonso, A. E.; García-Suárez, V. M.; Fernández-Martínez, J. L.; *Compressed Sensing Techniques Applied to Medical Images Obtained with Magnetic Resonance*. Mathematics, 11, 2023, <https://doi.org/10.3390/math11163573>
- [7] Llave, B. C.; *Aplicação do Método do Gradiente Espectral Projetado ao Problema de Compressive Sensing*. Dissertação de Mestrado defendida no IME-USP, 2012, <https://teses.usp.br/teses/disponiveis/45/45134/tde-02102012-162650/publico/bc1lave.pdf>
- [8] Luenberger, D. G.; Ye, Y.; *Linear and Nonlinear Programming*. Third Edition, New York: Springer, 2008, ISBN 978-0-387-74502-2

- [9] Mardani, M.; Gong, E.; Cheng, J. Y.; Vasanawala, S. S.; Zaharchuk, G.; Xing, L.; Pauly, J. M.; *Deep Generative Adversarial Neural Networks for Compressive Sensing MRI*. IEEE Transactions on Medical Imaging, 38(1), 2019, pp. 167-179, <https://doi.org/10.1109/TMI.2018.2858752>
- [10] Monika, R.; Dhanalakshmi, S.; *An efficient medical compression technique for telemedicine systems*. Biomedical Signal Processing and Control, 80(2), 2023, <https://doi.org/10.1016/j.bspc.2022.104404>
- [11] Meyer, C. D.; *Matrix Analysis and Applied Linear Algebra*. Second Edition, Philadelphia: SIAM, 2023, ISBN 978-1-61197-743-1
- [12] Pradhan, T.; Routray, A.; Kabi, B.; *Comparative Evaluation of Symmetric SVD Algorithms for Real-Time Face and Eye Tracking*. In: Nielsen, F., Bhatia, R. (eds) Matrix Information Geometry. Springer, Berlin, Heidelberg, 2013, https://doi.org/10.1007/978-3-642-30232-9_13
- [13] Tripathi, P.; Garg, R. D.; *Comparative Analysis of Singular Value Decomposition and Eigen Value Decomposition Based Principal Component Analysis for Earth and Lunar Hyperspectral Image*. 11th Workshop on Hyperspectral Imaging and Signal Processing: Evolution in Remote Sensing (WHISPERS), Amsterdam, Netherlands, 2021, pp. 1-5, <https://doi.org/10.1109/WHISPERS52202.2021.9483978>.
- [14] Watkins, D. S.; *Fundamentals of Matrix Computations*. Third Edition, Hoboken: John Wiley & Sons, 2010, ISBN 0470528338
- [15] Zermi, N.; Khaldi, A.; Kafi, M. R.; Kahlessenane, F.; Euschi, S.; *Robust SVD-based schemes for medical image watermarking*. Microprocessors and Microsystems, 84, 104134, 2021, <https://doi.org/10.1016/j.micpro.2021.104134>