



A comparative study of Singular Value Decomposition (SVD) and Discrete Cosine Transform (DCT) techniques for image compression applications.

Ruan Medina Carvalho, Gisele Goulart Tavares

Abstract. *The demand for methods capable of extracting knowledge from data and organizing large amounts of information has been growing in recent decades. Present in several applications in biomechanics and biomedicine, images sometimes require the use of computational methods to reduce the process complexity without compromising quality. This paper proposes the application of computational methods for image compression and pattern recognition, considering data storage and transfer issues. In particular, we present a comparative study between the Singular Value Decomposition (SVD) and Discrete Cosine Transform (DCT) techniques for image compression. The DCT method is one of the steps of the JPEG file compression process. Its application makes it possible to detect spatial redundancy in the image, identifying the similarity between neighboring pixels. The SVD consists of the factorization of a real or complex rectangular matrix, capable of dividing the system into linearly independent parts. The SVD factorization allows the identification of the most representative components of a matrix, which is explored in image compression. During the analysis, the perceptible quality of the compressed image is of great importance, so we propose an automatic way of obtaining some metrics for compression without excessive loss of information. The computational experiments in this work consisted of fixing error metrics and evaluating the performances of DCT and SVD. We also analyzed the coupling of these techniques in matrix compressions, proving a distinguished approach in comparison with the literature. This approach allowed us to analyze the reconstructions of the images and the process of pattern recognition of the SVD and DCT. The results demonstrate that DCT presents, in most tests, better performance in reducing memory usage in compression compared to SVD, despite resulting in lower image quality. Since the SVD factorization demands higher storage than DCT, image compression applications generally apply the DCT method. However, in medical applications, for example, where precision in the smallest details is essential for a correct diagnosis, the use of SVD combined with DCT can be more effective, generating compressed images with higher quality.*

Keywords: *Image Compression, Singular Value Decomposition (SVD), Discrete Cosine Transform (DCT)*

1 INTRODUÇÃO

A demanda por métodos que consigam extrair conhecimento a partir de dados e organizar grandes quantidades de informação vem crescendo nas últimas décadas. Neste sentido, o desenvolvimento e aplicação de métodos da álgebra linear computacional alimentam áreas como a Ciência de Dados, provendo o ferramental para organizar a quantidade de informação presente nos dados coletados (Sarkar, 2018). As imagens, presentes em diversas aplicações das áreas de biomecânica e biomedicina, demandam, por vezes, a aplicação de métodos computacionais para diminuir a complexidade de processamento sem comprometer a qualidade.

A compressão de imagens é um problema diretamente ligado a questões de armazenamento e transferência de dados, porém, os métodos aqui estudados podem ser estendidos para compressão de dados em geral, processamento de sinais ou aplicações em estatísticas em dados. Aplicar o estudo e entendimento dos métodos em imagens tem como vantagem obter a visualização direta e intuitiva dos resultados dos dados tratados e do padrão de identificação de informação de cada um deles, porém suas aplicações não se limitam a esse contexto.

Este artigo propõe a aplicação de métodos computacionais para compressão de imagens e reconhecimento de padrões, considerando questões de armazenamento e transferência de dados. Em particular, apresenta-se um estudo comparativo entre as técnicas de Decomposição em Valores Singulares (SVD) e Transformada Discreta de Cossenos (DCT) neste contexto.

Existem diferentes formas para compressão de imagens, portanto, estudos envolvendo comparações entre as diferentes abordagens são comuns na literatura. O trabalho de Majumder and Hussain (2008) (Majumder and Hussain, 2008), em específico, apresenta um estudo comparativo completo entre métodos que envolve testes com DCT e SVD. Mais recentemente, trabalhos como os de Sadek (2012) (Sadek, 2012) apresentaram as evoluções e modificações consideradas para o estado da arte do SVD, e trabalhos como o de Banday and Shah (2013) (Banday and Shah, 2013) desenvolve novas abordagens para a utilização do DCT para compressão de imagens, inclusive estudando o acoplamento da estratégias.

Neste contexto, justificam-se a execução de novas análises comparativas entre os métodos tratados, a fim de investigar suas performances como compressores em bases de dados mais atuais e investigar a possibilidade de acoplamento de estratégias.

2 Material e Métodos

Nesta seção apresenta-se a base teórica para a aplicação dos métodos na compressão de imagens, as implementações computacionais utilizadas e os *benchmarks* para os experimentos.

2.1 Decomposição em Valores Singulares (SVD)

A Decomposição em Valores Singulares, (Stewart, 1993) e (Chen, 2000), consiste em uma fatoração de uma matriz \mathbf{M} , $m \times n$, real ou complexa em três matrizes como descrito em (1), onde \mathbf{U} é uma matriz $m \times m$ ortogonal se real (ou unitária se complexa), Σ é uma matriz retangular diagonal $m \times n$ com números reais não-negativos na diagonal, e \mathbf{V}^* , matriz adjunta ou conjugada transposta de \mathbf{V} , é uma matriz $n \times n$ ortogonal se real (ou unitária se complexa).

$$\mathbf{M}_{m \times n} = \mathbf{U}_{m \times m} \Sigma_{m \times n} \mathbf{V}^*_{n \times n} \quad (1)$$

São denominados valores singulares de \mathbf{M} os valores na diagonal de Σ . As m colunas de \mathbf{U} e as n colunas de \mathbf{V} são os chamados vetores singulares à esquerda e vetores singulares à direita

de \mathbf{M} , respectivamente. Nessa construção, os valores singulares são ordenados do maior para o menor valor na diagonal principal de Σ , ou seja, $\sigma_1 > \sigma_2 > \dots > \sigma_t ; t = \min \{n, m\}$.

A utilização da decomposição SVD no problema da compressão de imagem surge da possibilidade de aproximar a matriz \mathbf{M} por uma matriz \mathbf{M}' formada pela operação matricial de multiplicação entre \mathbf{U}' , Σ' e $\mathbf{V}^{* \prime}$ com postos menores que os das matrizes que fatoraram a matriz original. Além disso, geralmente poucos dos valores singulares da decomposição tem valores relativamente altos. Por isso, mesmo que as imagens originais tenham *rank* completo, muitas dimensões que compõem esse *rank* não agregam muita informação da imagem (Sadek, 2012). Com isso, em uma compressão, representa-se a matriz \mathbf{M} como uma combinação linear de matrizes de *rank* unitário e constrói-se uma matriz \mathbf{M}' , aproximação de \mathbf{M} , com r termos da combinação linear descrita, desconsiderando as posições de dimensões menos representativas, isto é, $\mathbf{M}' = \sigma_1 u_1 v_1^* + \sigma_2 u_2 v_2^* + \dots + \sigma_r u_r v_r^* ; r < \min \{n, m\}$

2.2 Transformada Discreta de Cossenos (DCT)

A Transformada de Fourier é um mecanismo utilizado para representação de funções contínuas no domínio da frequência. Ela é formada por uma Transformada de Cossenos e uma de Senos definidas, respectivamente, pela parte real e imaginária da transformada (Easton, 2010).

A Transformada Discreta de Cossenos (DCT) é a versão discreta da parte real da Transformada de Fourier. A transformada utilizada nesse trabalho é a versão bidimensional do DCT que pode ser observada em (2) para uma matriz \mathbf{M} , $n \times n$. A matriz dessa transformada é composta de vetores ortonormais, sendo por isso uma matriz ortogonal (de rotação). Assim, essa transformada pode ser considerada como uma rotação, ou, mais especificamente, duas rotações consecutivas, uma em cada dimensão (Watson, 1994).

$$M'_{i,j} = \frac{1}{\sqrt{2n}} C(i) C(j) \sum_{x=0}^{n-1} \sum_{y=0}^{n-1} M_{x,y} \cos \left(\frac{(2y+1) j \pi}{2n} \right) \cos \left(\frac{(2x+1) i \pi}{2n} \right);$$

$$0 \leq i, j \leq n-1 \quad \text{Onde : } C(f) = \begin{cases} \frac{1}{\sqrt{2}} & ; se \quad f = 0 \\ 1 & ; se \quad f > 0 \end{cases} \quad (2)$$

Esse processo é vastamente aplicado em compressões de dados, pois posiciona os coeficientes mais significativos no início da matriz transformada dos dados. Como os demais valores são muitos pequenos, carregam pouca informação da imagem e podem passar por um processo de eliminação (Sayood, 2017). Assim, os itens mantidos após a eliminação são referentes ao bloco quadrado superior esquerdo de valores da matriz transformada.

2.3 Descrição das Imagens

Para a realização dos testes, foram utilizadas duas (2) fontes de imagens.

The USC-SIPI Image Database

O banco de imagens USC-SIPI (<http://sipi.usc.edu/database/>) é um conjunto digital de imagens para apoio em pesquisas de processamento e análise de imagens, e visão computacional. A primeira versão do banco data de 1977, e apesar de diversas atualizações com o passar dos anos,

manteve sua característica de apresentação de um conjunto de imagens de caráter básico. Todas as imagens utilizadas foram consideradas em escala de cinza com 8 bits para representação de cores. As imagens utilizadas nos testes desse trabalho são provindas do Volume 3: *Miscellaneous*, consistindo de 44 imagens, 16 previamente coloridas e transformadas para escala de cinza e 28 já apresentadas monocromáticas. O tamanhos das imagens variam, sendo 14 imagens com 256x256 pixels, 26 imagens com 512x512 pixels, e 4 imagens com 1024x1024 pixels. Além disso, duas (2) imagens mosaico do Volume 1: *Textures* foram adicionadas à base por questões de testes. A Figura 1 apresenta quatro (4) exemplos de imagens da base.

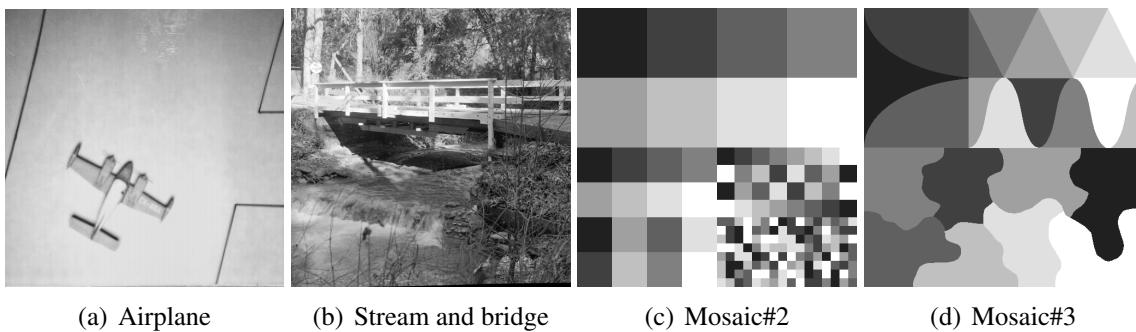


Figura 1: Exemplos de Imagens Utilizadas da Base USC-SIPI.

Rawzor Image Compression Benchmark

O banco de imagens proposto por Rawzor e colaboradores (<http://imagecompression.info/>) se propõe como uma alternativa às imagens clássicas utilizadas em pesquisas de compressão ao disponibilizarem imagens com resoluções superiores e escolhidas de forma técnica para enfatizar diferentes aspectos dos algoritmos modernos de compressão. A página do banco de imagens ainda oferece alguns resultados sobre perda de informação por compressão que podem ser utilizados para comparações durante a avaliação de algoritmos. As imagens do banco estão disponíveis em variações lineares de 8, 16 e 16 bits, RGB e cinza. Para esse trabalho, foram consideradas apenas as versões das imagens em tons de cinza. A Figura 2 apresenta alguns exemplos das imagens utilizadas.



Figura 2: Exemplos de Imagens Utilizadas da Base de Rawzor et. al..

2.4 Descrição dos Experimentos Computacionais

As implementações e os resultados foram obtidos por meio de um *script* desenvolvido em *Python3*. A fim de diminuir a complexidade do algoritmo e otimização das execuções, as seguintes bibliotecas/pacotes disponíveis na linguagem foram utilizadas: (i) *PIL* para a leitura

da matriz dos pixels da imagem; (ii) *fftpack* da biblioteca *Scipy* para a aplicação da transformada DCT; *Numpy* (para realização dos cálculos com matrizes e decomposição do SVD).

As comparações são realizadas tendo em vista os resultados das performances dos métodos para as imagens considerando métricas de erro médio, como em (Easton, 2010) e (Putalapattu, 2017). Considere o erro associado a um pixel i dado por $\epsilon_i = |f_i - c_i|$, onde f_i é o valor real do pixel e c_i é o valor após a compressão. Define-se como RMSE (Raiz do Erro Médio Quadrático) a raiz da média do somatório dos erros quadráticos em todos os pixels da imagem, dado pela Equação 3, onde N é o número de pixels na imagem tratada.

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N \epsilon_i^2} \quad (3)$$

Com relação às compressões, define-se como Taxa de Compressão (Cr) a razão entre o número de posições de memória utilizados pela matriz original em comparação com a matriz após a redução. A Equação 4 define tais medidas, onde $m \times n$ é a dimensão de origem e r é relativo à dimensão da compressão. Repare que a medida é aplicada sobre a representação computacional da imagem, que são diferentes após a fatoração SVD ou da transformação DCT. Essa medida não é aplicada diretamente sobre os pixels da imagem. Contudo, a redução do posto da matriz de representação das imagens implicará na redução dos detalhes na apresentação da imagem. Isso reflete no erro ϵ .

$$\text{Cr}_{\text{SVD}} = \frac{mn}{r(m+n+1)} \quad \text{Cr}_{\text{DCT}} = \frac{mn}{r^2} \quad (4)$$

Pretende-se ainda que seja possível estimar o número de dimensões necessárias para comprimir cada imagem sem que haja grandes perdas de informação? Neste caso, considerar um posto (k) padrão para a compressão não é um parâmetro justo entre diferentes imagens, uma vez que não leva em consideração a quantidade de informação presente em cada uma delas. Propõe-se estimar tendo como base os valores singulares da decomposição matricial para o SVD. Sendo assim, considere a matriz Σ de uma decomposição SVD. O somatório dos valores singulares (σ) de uma matriz M nos indica o que chama-se Quantidade de Informação ($Qinf$) da imagem, como visto na Equação 5.

$$Qinf_M = \sum_{i=1}^{\min\{n,m\}} \sigma_i \quad (5)$$

Com isso é possível definir um percentual da quantidade de informação que queremos em nossa compressão e encontrar o número de dimensões necessárias para tal representação (Chen, 2000). Visto isso, o Percentual de Quantidade de Informação Representada (%Qinf_M), descrito na Equação 6, é uma grandeza relativa para cada imagem que aproxima o percentual da quantidade de informação da imagem original que está sendo reproduzida na imagem comprimida, onde M_c é a matriz comprimida para dimensão r . O número de valores singulares r necessários para conseguir o percentual de informação desejado na compressão pode ser encontrado facilmente somando gradativamente os valores singulares obtidos na fatoração SVD, e assim podemos estimar o erro da compressão em termos relativos à imagem.

$$\%Qinf_M = \frac{\%Qinf_{M_c}}{\%Qinf_M} = \frac{\sum_{i=1}^r \sigma_i}{\%Qinf_M} \quad (6)$$

3 Resultados e Discussão

4 Conclusão

REFERÊNCIAS

- Banday, M. T. and Shah, T. (2013). A comparative study of image compression techniques.
- Chen, J. (2000). Image compression with svd. ecs 289k scientific computation.
- Easton, J. R. L. (2010). *Fundamentals of Digital Image Processing*. Jr. Roger L. Easton, *online* edition.
- Majumder, S. and Hussain, M. (2008). A comparative study of image compression techniques based on svd, dwt-svd and dwt-dct.
- Putalapattu, R. (2017). Jupyter, python, image compression and svd. an interactive exploration.
- Sadek, R. A. (2012). Svd based image processing applications: State of the art, contributions and research challenges. In (*IJACSA*) *International Journal of Advanced Computer Science and Application*, volume 3, pages 26–34.
- Sarkar, T. (2018). Essential math for data science - ‘why’ and ‘how’.
- Sayood, K. (2017). *Introduction to data compression*. Morgan Kaufmann.
- Stewart, G. (1993). On the early history of the singular value decomposition. In Siam, editor, *Society for Industrial and Applied Mathematics*, volume 35, pages 551–566.
- Watson, A. B. (1994). Image compression using the discrete cosine transform. *Mathematica journal*, 4(1):81.